CCL 2020

The 19th Chinese National Conference on Computational Linguistic

Proceedings of the 19th Chinese National Conference on Computational Linguistics

October 30 - November 1, 2020
Haikou, China

Order copies of this and other CCL proceedings from:

Chinese National Conference on Computational Linguistic (CCL)
Courtyard 4, South Fourth Street, Zhongguancun
, Haidian District, Beijing 100190
China
Tel: + 010-62562916
Fax: + 010-62661046
cips@iscas.ac.cn

Introduction

Welcome to the proceedings of the 19th China National Conference on Computational Linguistics (19th CCL). The conference and symposium were hosted online and co-organized by Hainan University, China. CCL is an annual conference (bi-annual before 2013) that started in 1991. It is the flagship conference of the Chinese Information Processing Society of China (CIPS), which is the largest NLP scholar and expert community in China. CCL is a premier nation-wide forum for disseminating new scholarly and technological work in computational linguistics, with a major emphasis on computer processing of the languages in China such as Mandarin, Tibetan, Mongolian, and Uyghur.

The Program Committee selected 109 papers (74 Chinese papers and 35 English papers) out of 303 submissions for publication. The acceptance rate is 35.97 – Machine Translation and Multilingual Information Processing (7)
– Fundamental Theory and Methods of Computational Linguistics (10)
– Minority Language Information Processing (9)
– Social Computing and Sentiment Analysis (12)
– Text Generation and Summarization (5)
– Information Retrieval, Dialogue and Question Answering (11)
– Linguistics and Cognitive Science (15)
– Language Resource and Evaluation (15)
– Knowledge Graph and Information Extraction (11)
– NLP Applications (14)

The final program for the 19th CCL was the result of intense work by many dedicated colleagues. We want to thank, first of all, the authors who submitted their papers, contributing to the creation of the high-quality program. We are deeply indebted to all the Program Committee members for providing high-quality and insightful reviews under a tight schedule, and extremely grateful to the sponsors of the conference. Finally, we extend a special word of thanks to all the colleagues of the Organizing Committee and secretariat for their hard work in organizing the conference, and to Springer for their assistance in publishing the proceedings in due time. We thank the Program and Organizing Committees for helping to make the conference successful, and we hope all the participants enjoyed the first online CCL conference.

September 2020

Maosong Sun
Sujian Li
Yue Zhang
Yang Liu

Organizers

19th CCL Program Chair:

Sujian Li, Peking University (China)
Yue Zhang,Westlake University (China)
Yang Liu, Tsinghua University (China)

19h CCL Area Co-Chairs:

Gaoqi Rao, Beijing Language and Culture University (China)
Timothy O'Donnell, McGill University (Canada)
Wanxiang Che, Harbin Institute of Technology (China)
Fei Xia, University of Washington (USA)
Xin Zhao, Renmin University of China (China)
Hongzhi Yin, University of Queensland (Australia)
Xiaojun Wan, Peking University (China)
Jinge Yao, Microsoft Research Asia
Kang Liu, Institute of Automation, CAS (China)
Ruihong Huang, Texas A&M University (USA)
Yang Feng, Institute of Computing Technology, CAS (China)
Haitao Mi, Ant Financial
Hongxu Hou, Mongolian university (China)
Fanglin Wang, Leading Intelligence Corporation
Weiguang Qu, Nanjing Normal University (China)
Nianwen Xue, University of Bratis (USA)
Meishan Zhang, Tianjin University (China)
Jiang Guo, Massachusetts Institute of Technology (USA)
Qi Zhang, Fudan University (China)
Jun Lan, Alibaba

19th CCL Local Arrangement Chairs:

Chunjie Cao, Hainan University (China)
Ting Jin, Hainan University (China)

19th CCL Evaluation Chairs:

Ting Liu, Harbin Institute of Technology (China)
Wei Song, Capital Normal University (China)

19th CCL Publications Chairs:

Shizhu He, Institute of Automation, CAS (China)
Gaoqi Rao, Beijing Language and Culture University (China)

19th CCL Workshop Chairs:

Jun Xu, Renmin University of China (China)
Xipeng Qiu, Fudan University (China)

19th CCL Sponsorship Chairs:

Zhongyu Wei, Fudan University (China)
Binyang Li, University of International Relations (China)

19th CCL Publicity Chairs:

Zhiyuan Liu, Tsinghua University (China)

19th CCL System Demonstration Chairs:

Shujian Huang, Nanjing University (China)
Zhaopeng Tu, Tencent (China)

19th CCL Student Counseling Chairs:

Pengyuan Liu, Institute of Computing Technology, CAS (China)

19th CCL Student Seminar Chairs:

Jinsong Su, Xiamen University (China)
Hongyu Lin , Institute of Computing Technology, CAS (China)

19th CCL Student Finance Chairs:

Yuxing Wang, Tsinghua University (China)

Table of Content

基于规则的双重否定识别——以“不v1 不v2”为例

王昱……1

基于语料库的武侠与仙侠网络小说文体、词汇及主题对比分析

张三乐，刘鹏远，张虎……10

基于计量的百年中国人名用字性别特征研究

杜冰洁，刘鹏远，田永胜……20

伟大的男人和倔强的女人：基于语料库的形容词性别偏度历时研究

朱述承，刘鹏远……31

用计量风格学方法考察《水浒传》的作者争议问题——以罗贯中《平妖传》为参照

宋丽，刘颖……43

多轮对话的篇章级抽象语义表示标注体系研究

黄彤，李斌，闫培艺，计婷婷，曲维光……54

发音属性优化建模及其在偏误检测的应用

郭铭昊，解焱陆……66

基于抽象语义表示的汉语疑问句的标注与分析

闫培艺，李斌，黄彤，霍凯蕊，陈瑾，曲维光……77

语用视角下复述句生成方式的类型考察

马天欢……88

面向汉语作为第二语言学习的个性化语法纠错

张生盛，庞桂娜，杨麟儿，王辰成，杜永萍，杨尔弘，黄雅平……97

中文问句的形式分类和资源建设

黎江涛，饶高琦……107

基于组块分析的汉语块依存语法

钱青青，王诚文……117

新支话题的句法成分和语义角色研究

卢达威……128

眼动记录与主旨结构标注的关联性分析研究

单昊聪，周强……141

汉语竞争类多人游戏语言中疑问句的形式与功能

张文贤，苏琪……153

融合目标端句法的 AMR-to-Text 生成

朱杰，李军辉……162

基于神经网络的连动句识别

孙超，曲维光，魏庭新，顾彦慧，李斌，周俊生……172

融合全局和局部信息的汉语宏观篇章结构识别

范亚鑫，蒋峰，褚晓敏，李培峰，朱巧明……183

基于图神经网络的汉语依存分析和语义组合计算联合模型

汪凯，刘明童，陈圆梦，张玉洁，徐金安，陈钰枫……195

基于强负采样的词嵌入优化算法

王雨晨，林淼哲，詹杰凡……207

联合依存分析的汉语语义组合模型

陈圆梦，张玉洁，徐金安，陈钰枫……215

基于对话约束的回复生成研究

管梦雨，王中卿，李寿山，周国栋……225

多模块联合的阅读理解候选句抽取

吉宇，王笑月，李茹，郭少茹，关勇……236

基于层次化语义框架的知识库属性映射方法

李豫，周光有……246

面向垂直领域的阅读理解数据增强方法

吕政伟，杨雷，石智中，梁霄，雷涛，刘多星……256

融入对话上文整体信息的层次匹配回应选择

司博文，孔芳……266

一种结合话语伪标签注意力的人机对话意图分类方法

丁健德，黄沛杰，许嘉宝，彭佑铭……277

基于 BERTCA 的新闻实体与正文语义相关度计算模型

向军毅，胡慧君，毛瑞彬，刘茂福……288

基于多任务学习的生成式阅读理解

钱锦，黄荣涛，邹博伟，洪宇……301

基于多头注意力和 BiLSTM 改进 DAM 模型的中文问答匹配方法

秦汉忠，于重重，姜伟杰，赵霞……313

基于 Graph Transformer 的知识库问题生成

胡月，周光有……324

基于 BERT 与柱搜索的中文释义生成

范齐楠，孔存良，杨麟儿，杨尔弘……336

基于深度学习的实体关系抽取研究综述

夏振涛，曲维光，顾彦慧，周俊生，李斌……349

小样本关系分类研究综述

胡晗，刘鹏远……363

基于阅读理解框架的中文事件论元抽取

陈敏，吴凡，王中卿，李培峰，朱巧明……376

基于 BERT 的端到端中文篇章事件抽取

张洪宽，宋晖，王舒怡，徐波……390

面向微博文本的融合字词信息的轻量级命名实体识别

陈淳，李明扬，孔芳……402

引入源端信息的机器译文自动评价方法研究

罗琪，李茂西……414

细粒度英汉机器翻译错误分析语料库"的构建与思考

裘白莲，王明文，李茂西，陈聪，徐凡……424

层次化结构全局上下文增强的篇章级神经机器翻译

陈林卿，李军辉，贡正仙……434

基于多语言联合训练的汉-英-缅神经机器翻译方法

满志博，毛存礼，余正涛，李训宇，高盛祥，朱俊国……446

基于跨语言双语预训练及 Bi-LSTM 的汉-越平行句对抽取方法

刘畅，高盛祥，余正涛，黄于欣，尤丛丛……457

基于子词级别词向量和指针网络的朝鲜语句子排序

闫晓东，解晓庆……467

基于统一模型的藏文新闻摘要

闫晓东，解晓庆，邹煜，李维……479

蒙古文拼写形式多样化现象研究

白双成，呼斯勒……491

面向司法领域的高质量开源藏汉平行语料库构建

沙九，周鹭琴，冯冲，李洪政,张天夫,慧慧……499

一种基于相似度的藏文词同现网络构建及特征分析

加羊东周，才智杰，才让卓玛，三毛措……509

《动词句法语义信息词典》知识内容说明书

袁毓林，曹宏……518

面向中文 AMR 标注体系的兼语语料库构建及识别研究

侯文惠，曲维光，魏庭新，李斌，顾彦慧，周俊生……528

面向人工智能伦理计算的中文道德词典构建方法研究

王弘睿，刘畅，于东……539

汉语否定焦点识别研究：数据集与基线系统

盛佳璇，邹博伟，沈龙骧，叶静，洪宇……550

面向医学文本处理的医学实体标注规范

张欢，宗源，常宝宝，穗志方，昝红英，张坤丽……561

汉语块依存语法与树库构建

钱青青，王诚文……572

汉语学习者依存句法树库构建

师佳璐，罗昕宇，杨麟儿，肖丹，胡正声，王一君，袁佳欣，余婧思，杨尔弘……581

CDCPP：跨领域中文标点符号预测

刘鹏远，王伟康，邱立坤，杜冰洁……593

多目标情感分类中文数据集构建及分析研究

刘鹏远，田永胜，杜成玉，邱立坤……604

基于 Self-Attention 的句法感知汉语框架语义角色标注

王晓晖，李茹，王智强，柴清华，韩孝奇……616

基于词语聚类的汉语口语教材自动推送素材研究

杨冰冰，赵慧周，王治敏……624

基于半监督学习的中文社交文本事件聚类方法

郭恒睿，王中卿，李培峰，朱巧明……634

基于多粒度语义交互理解网络的幽默等级识别

张瑾晖，张绍武，樊小超，杨亮，林鸿飞……645

文本情感分析中的重叠现象研究

娜仁图雅，徐晓音……656

基于 BiLSTM-CRF 的社会突发事件研判方法

胡慧君，王聪，代建华，刘茂福……667

结合金融领域情感词典和注意力机制的细粒度情感分析

祝清麟，梁斌，刘宇瀚，陈奕，徐睿峰，毛瑞彬……676

基于层次注意力机制和门机制的属性级别情感分析

冯超，黎海辉，赵洪雅，薛云，唐靖尧……688

基于循环交互注意力网络的问答立场分析

骆旺达，刘宇瀚，梁斌，徐睿峰……698

新型冠状病毒肺炎相关的推特主题与情感研究

梁帅龙，黄辉，张岳……707

融入多尺度特征注意力的胶囊神经网络及其在文本分类中的应用

王超凡，琚生根，孙界平，陈润……719

结合深度学习和语言难度特征的句子可读性计算方法

唐玉玲，于东……731

基于预训练语言模型的案件要素识别方法

刘海顺，王雷，陈彦光，张书晨，孙媛媛，林鸿飞……743

基于拼音约束联合学习的汉语语音识别

梁仁凤，余正涛，高盛祥，黄于欣,郭军军,许树理……754

基于数据增强和多任务特征学习的中文语法错误检测方法

谢海华，陈志优，程静，吕肖庆，汤帜……761

基于有向异构图的发票明细税收分类方法

赵珮瑶，郑庆华，董博，阮建飞，罗敏楠……771

半监督跨领域语义依存分析技术研究

毛达展，李华勇，邵艳秋……783

汉英篇章衔接对齐语料构建研究

李艳翠，冯继克，来纯晓，冯洪玉……795

Cross-Lingual Dependency Parsing via Self-Training

Meishan Zhang and Yue Zhang……807

A Joint Model for Graph-based Chinese Dependency Parsing

Xingchen Li, Mingtong Liu, Yujie Zhang, Jinan Xu and Yufeng Chen……820

Semantic-aware Chinese Zero Pronoun Resolution with Pre-trained Semantic Dependency Parser

Lanqiu Zhang, Zizhuo Shen and Yanqiu Shao……831

Improving Sentence Classification by Multilingual Data Augmentation and Consensus Learning

Yanfei Wang, Yangdong Chen and Yuejie Zhang……842

Attention-Based Graph Neural Network with Global Context Awareness for Document Understanding

Yuan Hua, Zheng Huang, Jie Guo and Weidong Qiu……853

Combining Impression Feature Representation for Multi-turn Conversational Question Answering
Shaoling Jing, Shibo Hong, Dongyan Zhao, Haihua Xie and Zhi Tang……863

Chinese Long and Short Form Choice Exploiting Neural Network Language Modeling Approaches
Lin Li, Kees van Deemter and Denis Paperno……874

Refining Data for Text Generation
Qianying Liu, Tianyi Li, Wenyu Guan and Sujian Li……881

Plan-CVAE: A Planning-based Conditional Variational Autoencoder for Story Generation
Lin Wang, Juntao Li,Dongyan Zhao and Rui Yan……892

Towards Causal Explanation Detection with Pyramid Salient-Aware Network
Xinyu Zuo, Yubo Chen, Kang Liu and Jun Zhao……903

Named Entity Recognition with Context-Aware Dictionary Knowledge
Chuhan Wu, Fangzhao Wu, Tao Qi and Yongfeng Huang……915

Chinese Named Entity Recognition via Adaptive Multi-pass Memory Network with Hierarchical Tagging Mechanism
Pengfei Cao, Yubo Chen, Kang Liu and Jun Zhao……927

A Practice of Tourism Knowledge Graph Construction based on Heterogeneous Information
Dinghe Xiao, Nannan Wang, Jiangang Yu,Chunhong Zhang and Jiaqi Wu……939

A Novel Joint Framework for Multiple Chinese Events Extraction
Nuo Xu, Haihua Xie and Dongyan Zhao……950

Entity Relative Position Representation based Multi-head Selection for Joint Entity and Relation Extraction
Tianyang Zhao, Zhao Yan, Yunbo Cao and Zhoujun Li……962

A Mixed Learning Objective for Neural Machine Translation
Wenjie Lu, Leiying Zhou, Gongshen Liu and Quanhai Zhang……974

Multi-Reward based Reinforcement Learning for Neural Machine Translation
Shuo Sun, Hongxu Hou, Nier Wu, Ziyue Guo and Chaowei Zhang……984

Low-Resource Text Classification via Cross-lingual Language Model Fine-tuning
Xiuhong Li, Zhe Li, Jiabao Sheng and Wushour Slamu……994

Constructing Uyghur Name Entity Recognition System using Neural Machine Translation Tag Projection
Azmat Anwar, Xiao Li, Yating Yang, Rui Dong and Turghun Osman……1006

Recognition Method of Important Words in Korean Text based on Reinforcement Learning
Feiyang Yang, Yahui Zhao and Rongyi Cui……1017

Mongolian Questions Classification Based on Mulit-Head Attention

Guangyi Wang, Feilong Bao and Weihua Wang……1026

The Annotation Scheme of English-Chinese Clause Alignment Corpus

Shili Ge, Xiaopin Lin and Rou Song……1035

Categorizing Offensive Language in Social Networks: A Chinese Corpus, Systems and an Explainable Tool

Xiangru Tang, Xianjun Shen , Yujie Wang, Yujuan Yang……1045

LiveQA: A Question Answering Dataset over Sports Live

Qianying Liu, Sicong Jiang, Yizhong Wang and Sujian Li……1057

Chinese and English Elementary Discourse Units Segmentation based on Bi-LSTM-CRF Model

Yancui Li, Chunxiao Lai, Jike Feng, Hongyu Feng……1068

Better Queries for Aspect-Category Sentiment Classification

Yuncong Li, Cunxiang Yin, Sheng-hua Zhong, Huiqiang Zhong, Jinchang Luo, Siqi Xu and Xiaohui Wu……1079

Multimodal Sentiment Analysis with Multi-perspective Fusion Network Focusing on Sense Attentive Language

Xia Li and Minping Chen……1089

CAN-GRU: a Hierarchical Model for Emotion Recognition in Dialogue

Ting Jiang, Bing Xu, Tiejun Zhao and Sheng Li……1101

A Joint Model for Aspect-Category Sentiment Analysis with Shared Sentiment Prediction Layer

Yuncong Li, Zhe Yang, Cunxiang Yin, Xu Pan, Lunan Cui , Qiang Huang and Ting Wei……1112

Compress Polyphone Pronunciation Prediction Model with Shared Labels

Pengfei Chen, Lina Wang, Hui Di, Kazushige Ouchi and Lvhong Wang……1122

Multi-task Legal Judgement Prediction Combining a Subtask of Seriousness of Charge

Zhuopeng Xu, Xia Li, Yinlin Li, Zihan Wang, Yujie Fanxu and Xiaoyan Lai……1132

Clickbait Detection with Style-aware Title Modeling and Co-attention

Chuhan Wu, Fangzhao Wu, Tao Qi and Yongfeng Huang……1143

Konwledge-Enabled Diagnosis Assistant Based on Obstetric EMRs and Knowledge Graph

Kunli Zhang, Xu Zhao, Xiabing Zhou, Qi Xie and Hongying Zan……1155

Reusable Phrase Extraction Based on Syntactic Parsing

Xuemin Duan, Hongying Zan , Xiaojing Bai and Christoph Zahner……1166

WAE_RN: Integrating Wasserstein Autoencoder and Relational Network for Text Sequence

Xinxin Zhang, Xiaoming Liu, Guan Yang and Fangfang Li……1172

基于规则的双重否定识别——以“不v1不v2”为例

王昱
北京大学中文系/ 北京

摘要

“不v1不v2”[1]是汉语中典型的双重否定结构形式之一，它包括“不+助动词+不+v2”（不得不去）、“不+是+不v2”（不是不好）、述宾结构“不v1...不v2”（不认为他不去）等多种双重否定结构，情况复杂。本文以“不v1不v2”为例，结合“元语否定”、“动词叙实性”、“否定焦点”等概念，对“不v1不v2”进行了全面的考察，制定了“不v1不v2”双重否定结构的识别策略。根据识别策略，设计了双重否定自动识别程序，并在此过程中补充了助动词表、非叙实动词表等词库。最终，对28033句语料进行了识别，识别正确率为97.87%，召回率约为93.10%。

关键词： 双重否定；非叙实动词；语义识别；否定焦点

Double Negative Recognition Based on Rules——Taking “不v1不v2” as an Example

Wang Yu
Department of Chinese Language and Literature,
Peking University / Peking

Abstract

“不v1不v2” is one of the typical double negation structures in Chinese.It includes “不+助动词+不+v2” (不得不去), “不+是+不v2” (不是不好), and “不v1...不v2”. Many kinds of double negative structures, such as “不v1...不v2” (不认为他不去),are very complex. Taking “不v1不v2” as an example, with the theories of ”non-truth-functional negation”, ”factuality of verbs” and ”negative focus”, this paper made a comprehensive investigation of “不v1不v2” . With the investigation result, this paper formulated a strategy for automatically recognizing the double negative structure of “不v1不v2”. According to the strategy, the automatic double negation recognition program is designed. During the designing process, this paper listed out the auxiliary verb list and the non factual verb List. Finally, the program was tested with 28033 sentences. The recognition accuracy is 97.87%, and the recall rate is about 93.10%.

Keywords: double negation , non-factural verb , semantic recognition , negative focus

[1]v1与v2指动词，不包括动词短语。

1 引言

1.1 研究背景

在语义识别领域，否定对语义有着重要的影响，否定的有无影响整个句子的真值与情感。例如，“我喜欢这件衣服”，这句话的真值为真，情感为积极。但是如果在动词前加上否定词“不”（即“不喜欢”），真值便为假，情感便为消极。因此，句子中的否定成分是自然语言语义识别处理时必须考虑的内容。而在否定用法中，有一种特殊的用法——双重否定。丁声树先生在《现代汉语语法讲话》中将其概括为：“一句话先后用两个否定词，如‘不能不去’，‘没有人不去’，‘非去不可’之类，都是双重否定的句法。双重否定意思上是肯定的，不过跟单纯肯定不全一样”[1] 。例如，“我不得不喜欢他”指“我得喜欢他”，“我不能不同意这个观点”指“我得同意这个观点”。虽然用的是否定的格式，表达的却是肯定的语义。这种连用否定表达肯定的特殊结构即为双重否定。对于自然语言语义识别来说，双重否定是处理语料时必须考虑的内容。如果计算机无法识别双重否定，则可能会导致整个句子的语义真值和情感极性的判断错误。例如，“我不知道你不喜欢他”与“我不认为你不喜欢他”，前者表示“你不喜欢他”，后者表示“你喜欢他”，两者具有相反的语义，又如“不是不容易，是很难”与“不是不容易，是超简单”的语义也不同，若计算机无法正确识别双重否定，便可能无法判断两者语义真值上的差别。因此，由于双重否定本身情况十分复杂且对语义有着重要影响，我们有必要对双重否定进行更全面深入的研究与探索。双重否定的自动识别将有利于句子语义与情感的识别，有助于聊天机器人、文本分析、问答系统等人工智能应用的进一步发展。

1.2 文献综述

目前国内对于双重否定的研究主要集中在双重否定的定义、范围、语义和语用等理论问题方面。双重否定的定义与范围方面，学界一直存在争议，如吕叔湘先生(1956) 、王力先生（1943）等认为含有否定词的反诘句是双重否定，而符达维先生（1986）则认为反诘句不是双重否定。双重否定格式方面，已有研究都是从分类与举例的角度进行讨论，尚未有一篇论文从形式的角度对双重否定格式进行具体详细的遍历分析。目前对双重否定格式概括最为全面的是芜崧（1987）所划分的八大类型，25个格式，但是其主要涵盖的是“构式”类的双重否定，对于非构式类的双重否定尚未进行全面的考察。语义语用方面，叶文曦（2013）、方绪军（2017）、何爱晶（2019）等引入了Ladusaw（1997）的形式语义学，Horn（1985）的元语否定等理论，对一些典型的结构进行了分析，得出了具有解释力的成果。然而，由于双重否定的范围、格式还未确定，目前学者只集中分析了几个典型的结构，概括面十分有限，无法直接应用于实践。

自然语言理解领域，关于双重否定识别的研究非常少。目前只有王勇（2014）在其极性词典的构建中，简单地搜集了一些典型的双重否定结构，构建了一个双重否定词典。具体如下：

双重否定词典	绝非不、并非不、不是不、不能不、不会不、不可不、不要不、不得不、没有不、无不、不无

Figure 1: 王勇（2014）[2]双重否定词典

然而该词典所归纳的结构数量有限，大量双重否定结构未被收入，如“不应不”、“不准不”、“不该不”、“非...不可”、“无一不”、“没有...不...”，等等。

1.3 本文的选题及意义

综上所述，目前国内多从分类与举例的角度对双重否定进行讨论，尚未有从形式的角度对双重否定格式进行遍历分析的成果。关于双重否定，我们只知道一些具体的实例，并没有归纳其完整的形式格式。另一方面，双重否定对于语义的真值与情感极值有着重大的影响，如“我非去不可”与“我不去”真值完全相反，然而目前国内并未有学者对双重否定的识别予以关注。基于

[1]丁声树等.现代汉语语法讲话[M].北京：商务印书馆，2004.200-202.

此，本研究选取了“不v1不v2”为研究对象，尝试对含“不”的双重否定结构实现自动识别功能。本文将对“不v1不v2”做全面的考察，判断属于双重否定结构的“不v1不v2”的具体特征，并根据归纳出的特征建立自动识别程序。最后，建立的自动识别程序应能对含有多个“不”的语料进行识别，区分属于双重否定与不属于双重否定的句子，并返回其内部的双重否定结构。本研究可以作为预实验，为后续的全面的双重否定自动识别程序做基础。

2 “不v1不v2”双重否定的类型及识别规则

目前关于双重否定的定义和标准，学界尚未有定论。鉴于语义真值识别和情感极值判断是计算机对否定结构进行语义识别时所面临的主要问题，本文采用了形式语义学上对双重否定的定义：只要两次否定与肯定在语义真值上相同，“¬ ¬ P==P”，即为双重否定。符合双重否定条件的结构即为双重否定结构。

传统语言学上“不v1不v2”通常指“不得不，不能不，不要不，不会不，不是不”等v1为助动词或“是”的结构。这种“不v1不v2”结构是目前学界公认的典型的双重否定结构。然而从形式上来说“不v1不v2”还可以指“不走不跳”、“不买不卖”等并列结构，“不吃饭不睡觉”、“不买票不进场”等紧缩条件复句结构，以及“不觉得不尊重”、“不想不去”等述宾结构。这些结构也存在含有双重否定结构的可能。因此，为了得到完整的双重否定结构格式，本文结合前贤研究，采用遍历的方法，梳理了所有的“不v1不v2”[3]的语法形式并对其进行考察，试图找出所有可能的双重否定结构。具体的遍历方法如下：

1.在CCL语料库中搜寻含有“不$Num不”的语料（Num取值为“1-20”）；其中Num表示“不”与“不”之间相隔的字数。如“不得不”的Num为1，而“不觉得不”的Num为2；

2. 根据所得语料，结合词性限制，在“不$Num不”中提取所有可能的“不v不v”，并进行数据统计；

3. 根据统计结果与具体语料，对结果进行概括，抽象出具体的所有可能的“不v1不v2”结构。

具体结果如下表所示。

不 v1 不 v2	例句
不+助动词+不+v2	不得不去；不能不喜欢
不+是+不+v2/不+是…+不+v2	不是不喜欢/不是他不知道
不 v1+不+v2/不 v1…+不+v2 （紧缩、并列、主谓、述宾等结构）	不去不行；不唱不跳；不吃饭不好； 不觉得不尊重；不认为他不喜欢你

Table 1: “不v1不v2”格式表

2.1 不+助动词+不+v2

“不+助动词+不+v2”是目前公认的双重否定的结构，其具体格式为“不+表示可能或必要的助动词+不+v2”，如“不得不去”、“不会不来”、“不可不说”等等。例句如下：

1a. 一些农村学校的校长为了保证学习的正常运转，不得不四处筹钱。

1b. 一些农村学校的校长为了保证学习的正常运转，（必须）得四处筹钱。

2a. 这样重大的事件新闻界不会不给予特别重视。

2b. 这样重大的事件新闻界会给予特别重视。

1-2的a、b例句，虽然它们的语气强度存在区别，如1a中的“不得不”比1b中的“得”的肯定语气更强，然而这种语气变化并不影响语义的真值。1a、2a仍与1b、2b的语义真值一致，符合“¬ ¬ P==P”双重否定的标准，属于双重否定结构。

对于“不+助动词+不+v”这种在长期使用中已经逐渐变为接近于构式的固化结构，本文采取简单的字符串匹配的方法，便可实现对其的识别。

[3]v1若不为助动词，则前面还可带助动词。构成“不+助动词+v1+不+v2”结构，如“不会认为不好”，这种结构也是识别的对象。

2.2 不+是+不+v2/不+是...+不+v2

2.2.1 不+是+不+v2

“不+是+不+v2”与“不+助动词+不+v2”结构类似，也是最为常见的双重否定结构之一，但是“不+是+不+v”的情况更为复杂。因为“不+是+不+v”中的“不是”除了可以表示描述性真值否定（descriptive truth-functional negation）外，还可以表示元语否定（non-truth-functional negation）。“所谓元语否定，就是用元语言对对象语言所描述的非真值语义的否定，...是一种非真值意义否定；与之相对应的是真值否定，否定的是句子的真值条件（truth condition）”[4] 这种元语否定常常是引述性否定，是对之前对话中已出现的内容的否定。例句如下：

3a. “可现在杀他不容易啊。”有人说。不是不容易，是根本不可能。
3b. * “可现在杀他不容易啊。”有人说。是容易，是根本不可能。
4a. “可我不想跟她结婚。”刘东北进一步道，“不是不想跟她结婚，是不想结婚。”
4b. * “可我不想跟她结婚。”刘东北进一步道，“是想跟她结婚，是不想结婚。”
5a. 不是不要读书，而是要读得更好。
5b. 是要读书，（而）是要读得更好。

上述例句，3-4中的“不是”为元语否定，它是对之前内容的语用否定，而不是对本句内容的语义否定。3a与3b，4a与4b语义不一致，且3b、4b内部也有语义矛盾，无法成立。从语义来说，3a、4a中的“不是不”只包含一重语义否定，其逻辑式为“¬ P”，并不符合“¬ ¬ P==P”双重否定的标准，因此，这种“不是不”不属于双重否定结构。而5a例句中的“不是”是对语义的否定，其“不是不”符合“¬ ¬ P==P”双重否定的标准，属于双重否定结构。

综上所述，对于不+是+不+v”，本文需要使计算机先区分其中的“不是”是元语否定还是描述性真值否定，然后才能判断其是否为双重否定结构。在单纯的文本里，绝大多数表示元语否定的“不是”都只出现在“不是+不+x，（而）是+y”的结构中。因此，下面我们将以“不是+不+x，（而）是+y”为对象作进一步的讨论。

由于元语否定“是对命题适宜性进行的修订性否定”，其修正内容并不是对真值进行修正，因此修正内容（y）的真值应与“错误内容（即“不x”）”的真值一致。而对于描述性真值否定来说，其修订的是命题真值，因此，其修正内容（y）的真值应与“错误内容（即“不x”）”的真值相反。根据这一观察，我们提出了判断“不是+不+x，（而）是+y”是否为双重否定结构的具体方法：提取“不是+不+x”中的x,并匹配y与“不x”的真值[5]。若“不是+不+x，（而）是+y”中“y”的真值与“不x”一致，则句中的“不是不”不为双重否定结构；否则“不是不”为双重否定结构。

结合这一方法，本文对3、5例句分析如下：

3a. “可现在杀他不容易啊。”有人说。不是不容易，是根本不可能。
（y真值：-1，不x真值：-1，不为双重否定）
5a. 不是不要读书，而是要读得更好。
（y真值：1，不x真值：-1，为双重否定）

根据上述方法，本文可以判断出上述四个例句中的“不是不”，是否是双重否定结构。[6] 因此，本文采用字符串匹配与计算真值的方法，实现计算机对“不是不”双重否定结构的识别。

2.2.2 不+是...+不+v2

不+是...+不+v2除了需要满足“‘不是’表示语义否定”的条件外，还需要满足“不是”的否定焦点落在“不v2”上的条件。如“不是我故意不来”并不等于“是我故意来”。因为在该例中，“不是”的否定焦点是“故意”而不是“不来”。具体否定焦点的限制条件本文将会在2.4.2节中进行讨论。

2.3 “不v1不v2”/“不v1...不v2”(非述宾结构)

“不v1不v2”指“v1”不是助动词、不是“是”的结构，它的结构类型有并列、紧缩（主谓）、述宾等。“不v1...不v2”则指“不v1”与“不v2”之间不紧邻的“不v1+其他内容+不v2”结构。下面

[4]何爱晶.反叙的非真值义否定和真值义肯定[J].外语研究,2019,36(04):第25页.
[5]判断真值的方法为，计算其谓语真值。一次否定为-1，两次否定为1，以此类推。判断真值时，也包括内含否定的谓词，如“讨厌”、“丑”等。
[6]这种策略，理论上仍有可能存在例外，但是目前尚未在自然语言语料中发现例外。

本文将对以上各个结构类型一一进行讨论。

紧缩结构[7]的“不v1（...）不v2”包括“不x不行”、“不x不成”等固定形式的短语结构，也包括“不买票就不让进”“不给钱不办事”这一类表达。其语义为“如果不v1，那么不v2”。关于紧缩条件类的结构是否为双重否定未有定论。在这里本文从形式语义学的角度对其进行讨论。以“不x不行”为例。“x行”语义为“如果x，那么行”。P命题可以分解为q1“x”，q2“y”，逻辑式为蕴含式q1→q2。而“不x不行”语义为“如果不x，那么不行”，逻辑式应为蕴含式¬q1→¬q2。从下列真值表本文可以看出，q1→q2与¬q1→¬q2的语义真值不一致，不符合“¬ ¬ P==P”的标准，因此从形式语义学来看，紧缩语义结构不是双重否定结构。[8]

q1	q2	¬q1	¬q2	q1→q2	¬q1→¬q2
T	T	F	F	**T**	**T**
T	F	F	T	**F**	**T**
F	T	T	F	**T**	**F**
F	F	T	T	**T**	**T**

Table 2: 紧缩结构语义真值表

并列结构的“不v1（...）不v2”，指“不哭不闹”、“不高不低”这一类表达。袁毓林（1999）提出并列结构“通常不能通过直接在这种谓词性并列结构的前面加上“不、没有”等否定词来构成否定式,而是要在这种并列结构的各个直接成分之前分别加上“不、没有”等否定词。”[9] 因此，“不v1不v2”只是“v1v2”并列结构的一重否定结构，不属于双重否定结构。如“不哭不闹”不等于“哭闹”。

2.4 “不v1不v2”/“不v1...不v2”（述宾结构）

述宾结构的“不v1不v2”（不v1与不v2紧邻）、“不v1...不v2”（不v1与不v2非紧邻）指“不觉得不好”、“不知道你不来”等v1为述语，“不v2”为宾语的结构。该结构中存在一部分结构，其“不v1”对“不v2”有语义指向，属于双重否定结构，如例7;同时还存在一部分结构“不v1”与“不v2”之间没有语义指向，不是双重否定结构，如例6。

6a. 我不知道你不来。（述宾）　　6b. *我知道你来

7a. 平时也是人来人往，我不相信你不难受。（述宾）

7b. 平时也是人来人往，我相信你难受。（述宾）

因此，对于述宾结构的“不v1（...）不v2”，本文的主要任务是找出其中“不v1”对“不v2”存在语义指向、属于双重否定的结构，并提取其特征，制定识别规则。下面本文将以“不v1...不v2”为对象进行讨论。

2.4.1 第一个条件：v1为非叙实动词

首先我们需要确定在什么情况下“不v1”对“不v2”具有语义指向与管辖功能。

8a.我不知道他不来　　8b.*我知道他来

9a.我不幻想他不来　　9b.*我幻想他来

10a.我不认为他不来　　10b.我认为他来

通过例句，可以发现，当v1为“知道”、“幻想”时“不v1...不v2”不能转换成“v1...v2”，而当v1为“认为”时，却可以转换。“知道”、“幻想”、“认为”同样是动词，却存在着区别。本文认为，“不v1”对“不v2”是否有语义指向或管辖的作用与v1的语义有关，具体来说与v1的叙实性有关。李新良（2015）将叙实性定义为“叙实性是动词的一种语义功能，即动词预设其宾语小句真值的能力。具体来说，肯定式和否定式都预设其宾语小句为真的动词是叙实动词，叙实动词具有的预设其宾语小句为真的能力叫叙实功能；肯定式和否定式都不预设其宾语小句为真，也不预设其宾语小句为假的动词是非叙实动词，非叙实动词具有的不预设其宾语小句为真，也不

[7]我们认为主谓结构的不v1（...）不v2与紧缩结构相似。由于篇幅限制，便不再讨论。

[8]虽然紧缩语义结构不是双重否定结构，但是比起普通的并列结构，紧缩语义结构仍然具有其独特的表达功能，应该与其他结构区分开，未来我们希望能对其展开进一步的研究。

[9]袁毓林.并列结构的否定表达[J].语言文字应用,1999(03):第42页.

预设其宾语小句为假的能力叫非叙实功能；肯定式和否定式都预设其宾语小句为假的动词是反叙实动词，反叙实动词具有的预设其宾语小句为假的能力叫反叙实功能"[10] 。因此，对于叙实动词和反叙实动词来说，无论它自身是肯定式还是否定式，它的宾语小句的真值都不会改变。如"我知道他不来"与"我不知道他不来"中宾语小句的语义都是"他不来"。所以，对于叙实动词与反叙实动词，由于其宾语小句真值已定，否定式不能管辖后一宾语小句，无法满足"¬ ¬ P==P"的条件，不为双重否定。而当v1是非叙实动词（如：认为）时，由于非叙实动词的宾语小句并没有预设，在述宾结构中，v1对宾语具有约束管辖关系，能够影响宾语的真值，具有可以产生"¬ ¬ P==P"的条件，存在属于双重否定的可能。因此，本文得出了"不v1...不v2"述宾结构为双重否定的第一个条件：v1为非叙实动词。

2.4.2 第二个条件："不v1"的否定焦点包含v2

该条件只对分开的"不v1...不v2"有约束，对于紧连的"不v1不v2"并无影响。当句子v1确定为非叙实动词时，该句子并不一定为双重否定句。示例如下：

11a.我不认为他不来。　　11b.我认为他来。

12a.我不认为他故意不来。　　12b. *我认为他故意来。

13a.我不相信他不喜欢我。　　13b.我相信他喜欢我。

14a.我不相信他不喜欢我到了看见我就恶心的地步。

14b.*我相信他喜欢我到了看见我就恶心的地步。

在例句中，11a、13a可以转换为11b、13b，而12a、14a却不能转换为12b、14b。为何？本文认为这主要与否定的焦点有关。袁毓林(2000)指出"有的成分表达的是句子的预设意义，属于旧信息，事实上它们的意义在否定的情况下仍然得以保持;有的成分表达的是句子的焦点意义，属于新信息，它们是真正被否定的。" 当v2不是否定焦点时，"不v1"并不会对v2进行否定，不会形成"¬ ¬ P"，因此不满足"¬ ¬ P==P"的条件，不是双重否定结构。如"我不认为他故意不来。"中的"不v1"否定的是"故意"而不是"不来"，其中"不来"是预设成分，属于旧信息，"不v1"并不会影响到"不来"的真值。如：

15a.我不认为他故意不来。　　预设：他不来。

15b.我认为他故意不来。　　预设：他不来。

因此，为了满足双重否定"¬ ¬ P==P"的条件，本文需要确保"不v1"的否定焦点是落在v2上的。结合袁毓林（2000）关于否定词焦点与辖域的观点，本文将"不v1...不v2"中"不v1"的否定焦点的情况归结如下：

（1）若v2存在谓语状语或者谓语补语[11]，则谓语状语或者谓语补语是否定焦点；反之，则v2是否定焦点；

（2）若v2的状语、补语、宾语中含有全称量词或者"一+量"时，量词为否定焦点。

16a.我不认为他每一天都不来。　　16b.*我认为他每一天都来。

17a.我不认为他不喜欢所有人。　　17b.*我认为他喜欢所有人。

（3）若v2与其他谓语结构构成紧缩复句结构，如"不v2就..."、"不v2不..."，v2不是否定焦点，否定焦点是整个紧缩复句，无法构成双重否定结构；若v2与其他谓语结构构成并列结构"不v2不v3"时，v2不是否定焦点，否定焦点是整个并列结构，但是仍然可以构成双重否定结构，不过转换成肯定式时需要同时去掉并列结构中所有的"不"。

18a.我不认为他不吃不喝。　　18b. 我认为他吃喝。[12]

19a.我不认为不去就不行。　　19b.*我认为去就行。

由此本文得出了述宾结构的"不v1...不v2"为双重否定的第二个条件：不v1的否定焦点包含v2。这一条件同样适用于"不是...不v2"。具体参照2.2.2。

[10]李新良,王明华.汉语动词的叙实性研究的应用前景[J].对外汉语研究,2015(02):第122页.

[11]当否定句中存在"连、就"等标记时，双重否定转换时，语义会发生一定的改变。但是不v1的否定焦点仍然包含了"V2"。

[12]这里否定到肯定的转化是一个常规的理解。本文不否认存在一定的可能性，"我不认为他不吃不喝"表达的是"我认为他吃但是不喝"，或者"我认为他不吃但是喝"。但是这种可能性不符合人们平时交际的习惯，因此本文不对此进行进一步的讨论。

3 双重否定自动识别程序的建立

3.1 词库的建立

为了使计算机能够识别助动词、非叙实动词，本文对助动词与非叙实动词进行了梳理，在常用的基础词表中补充了助动词词表与非叙实动词词表。助动词方面，本文以郑贵友（1989）[13] 整理的助动词范围为基本，结合鲁晓琨（2004）[14]等前人的研究以及现代汉语的使用情况，选取了26个助动词，构成常用助动词词表。具体如下：

能、能够、可能、会、可以、应该、应、应当、要、得、愿意、愿、甘愿、肯、可、情愿、想、要、敢、该、配、当、准、许、得、容

非叙实动词方面，结合袁毓林、李新良等人对非叙实动词的研究，本文认为非叙实动词多为心理动词。因此本文对心理动词进行了考察。若一个心理动词的宾语的真值无法确定，则该心理动词为非叙实动词。以此为标准，本文对心理动词进行了考察，筛选出了24个非叙实动词[15]。整理如下：

认为、觉得、想（料想、猜想）、感到、情愿、相信、乐意、愿意、盼望、希望、猜、猜测、揣测、揣摩、推测、估计、估摸、猜想、考虑、打算、说（认为）、同意、赞同、允许

3.2 双重否定自动识别程序的流程

以第二章所讨论的语法规则为核心，本文设计了双重否定自动识别程序。程序的输入为含有两个副词词性“不”的语料txt文件，输出为一个txt文件。输出的txt文件分为两个部分，一是所有识别出的含有双重否定结构的句子及其双重否定结构类型；二是语料总句数、双重否定句子句数、各双重否定结构的句子句数等统计信息。示例如下：

【文件名:\当代\报刊\人民日报\1995年人民日报\6月份.txt 文章标题:作者:】99937:...该所旅馆的老板娘川邑太太，看到有些年轻夫妇因为带着孩子去旅行不方便而不得不取消旅行计划，灵机一动抓住了“亮点”，办起了托儿服务旅馆，... 【文件名:\当代\报刊\人民日报\1995年人民日报\6月份.txt 文章标题:作者:】99938:...一次，队里统一出工。

**********此句话为双重否定句，双重否定结构为不+助动词+不**********
因此，它们的主张和行动不可能不对整个世界经济与政治产生重大影响。

**********此句话为双重否定句，双重否定结构为不+助动词+不**********
一共有28033句不+助+不的结构有628句不+助动词+非叙实动词+不 的结构有45句不+非叙实动词+不的结构有27句不+助动词+非叙实动词+...不的结构有2句不+是+...不的结构有199句

Figure 2: 实验结果示例图

具体程序的识别步骤如下：

（1）通过python程序使用哈工大LTP对语料进行分句、分词、词性标注；对每一个句子进行以下（2）-（6）操作；

（2）检测其是否含有“不+助动词+不+v”结构，若含有，则为双重否定句，其双重否定结构为“不+助动词+不+v”，将句子写入文件并归入数据统计，跳过后续步骤。若不含有，则进行下一步；

（3）检测其是否含有“不+助动词+非叙实动词+不+v”结构，若含有，则为双重否定句，其双重否定结构为“不+助动词+非叙实动词+不+v”，将句子写入文件并归入数据统计，跳过后续步骤。若不含有，则进行下一步；

（4）对句子进行句法分析；

（5）检测其是否含有述宾关系的“不+非叙实动词（+...）+不+v”结构，若含有，则为双重否定句，其双重否定结构为“不+非叙实动词（+...）+不+v”，将句子写入文件并归入数据统计，跳过后续步骤。若不含有，则进行下一步；

[13]郑贵友.汉语“助动词”的研究刍议[J].汉语学习,1989(06):23-27.
[14]鲁晓琨. 现代汉语基本助动词语义研究[M].北京: 中国社会科学出版社,2004.
[15]非叙实动词的界限并不是完全清晰的。非叙实动词与叙实动词、反叙实动词之间还存在一定的渗透性。由于其情况较为复杂，且对本文研究的影响较小，故暂不讨论。

（6）检测其是否含有述宾关系且“不是”为语义真值否定（通过否定值判断）的“不+是（+...）+不+v”结构，若含有，则为双重否定句，其双重否定结构为“不+是（+...）+不+v”，将句子写入文件并归入数据统计，跳过后续步骤。若不含有，则该句不含有双重否定结构，进行数据统计。

4 双重否定自动识别实验

4.1 实验语料来源

本文在ccl语料库中，提取了100000条含有“不”的语料，并通过程序从中抽取含有两个副词词性“不”的句子，共计28033句，以此为“不”的基础语料，测试双重否定识别程序的正确率。同时，本文从28033句语料中选取了1000条语料作为人工检测语料，以测试双重否定识别程序的召回率。

4.2 实验语料来源

本文使用双重否定识别程序对28033句语料进行识别，获得了894句含有双重否定结构的句子。经人工的检校，发现以上894个句子中，含有双重否定结构的句子数为875，该程序识别正确率为97.87%。具体如下表[16]：

结构	识别句子数	错误句子数	正确句子数	正确率
不+助+不+v	628	0	628	100%
不+助动词+非叙实动词+不+v	45	2	43	95.56%
不+非叙实动词+不+v	27	3	24	88.89%
不（+助动词）+非叙实动词+...+不+v	2	2	2	100%
不+是（+…）+不+v	192	11	181	94.27%
总计	894	19	875	97.87%

Table 3: 实验结果统计表

为了测试该程序的召回率，本文人工对1000句语料进行检校，筛选出了29句含有双重否定结构的句子。本文将这1000句语料输入到双重否定识别程序中，程序识别出了27句含有双重否定结构的句子。因此，该程序的召回率约为93.10%。根据F1值公式与上述数据，该程序的F1值为95.43%。

4.3 实验分析

无论是正确率还是召回率，实验的准确率都是在百分之九十多，未达到百分之百。我们对以上三个未达到百分百正确率的双重否定结构的语料进行分析。结果显示，程序识别与召回错误主要与句子的句法分析错误有关。由于分词与句法分析等基础自然语言处理工具的问题，程序对一些句子的句法判断错误，导致一些原本应被判为并列关系、因果关系的成分，被误判为述宾关系，从而使整个双重否定结构的判断错误。例句如下：

20. 如果用适用各种土质的几十台钻机同时作业，电力条件不允许，地下管网不安全。

21. ...每到拔棉柴时，农民心里直犯愁，用手拔吧，费工费时不说，拔不了多少，便满手血泡。有的急于求成把将来要办的事情，拿到今天来办，由于条件不允许迟迟开展不了。

以上例句中，v1与v2都并非述宾关系，不符合程序所制定的双重否定的规则，然而程序并未检索出这一点，导致识别错误。这个问题主要与句法分析的处理工具有关，本文暂时无法对其进行进一步的改进。

除此之外，还有一些句子的识别错误，是由于结构格式总结疏漏所造成的。例句如下：

22.因为他们不是不懂足球，便是偏爱市井蜚闻的帮闲之辈。

上述例句，其结构为“不是...便是...”，整体为选择并列的关系。其中“不是不懂足球”中的“不是”并不表示否定的功能，而是选择的功能。可以理解为“要么是不懂足球”，因此不属于双重否定结构。然而由于我们归纳结构格式时，没有考虑到类似的特殊结构，将其简单地归入了“不是不”结构，从而导致了错误。这一问题，可以通过更多次的实验与探究来进行改进。

[16]由于文章讨论的范围限制，删去了“岂不是”这类不+反问词的句子。

4.4 实验讨论

上述实验结果表明，本文设计的双重否定识别程序具有从语料中识别出含有双重否定结构的句子的能力，程序识别的正确率与召回率都较高，证明了基于规则进行双重否定自动识别的可能性，具有实践价值。然而，本次实验的实验对象单一（只有“不v1不v2”），许多常见的双重否定结构，如“不”与其他否定词构成的双重结构（“不...没有”、“没有...不”、莫不、无不等），“不”与反问词构成的双重否定结构（“难道不”、“岂不”等），都未被纳入监测范围。因此，为了进一步完善双重否定识别程序，我们有必要对双重否定结构进行更为全面的考察。只有这样，双重否定识别程序才能真正地实现计算机对自然语言中双重否定结构的自动识别，具有实际的应用价值。

5 总结

本文讨论了“不”的双重否定结构，采用遍历的方式，对所有可能的“不v1不v2”的结构进行了考察，整理出了完整的“不v1不v2”结构，对现有的双重否定结构形式进行了补充。本文首次采用“非叙实动词”作为述宾式“不v1...不v2”双重否定结构成立的条件，解释了“我不知道你不来”与“我不认为你不来”之间的区别；引入了否定焦点的概念，解释了“我不认为你不来”与“我不认为你故意不来”之间的区别。在归纳所有的双重否定结构后，本文从语义管辖的角度对双重否定形式上的形成进行了解释。实践方面，本文针对所有的“不v1不v2”提出了相应的自动识别策略。基于规则，设计了“不”的双重否定识别程序，并用该程序对28033句语料进行了识别，识别正确率为97.87%，召回率约为93.10%，F1值约为95.43%。在程序构建过程中，本文还补充了助动词词表与非叙实动词词表。该程序证明了基于规则进行双重否定识别的可能性，是一次成功的尝试。然而，这只是对“不v1不v2”这一个格式进行的分析。未来，本文将引入更多的双重否定结构，全面地对双重否定进行考察，以期真正地实现计算机对双重否定的自动识别，将其应用于文本分析、问答系统等自然语言处理应用当中。

参考文献

Horn,L. 1989. *A Nature History of Negation.* University of Chicago Press, Chicago, US, 311-312.

Ladusaw, W. A. 1997. *Negation and polarity items. In S. Lappin (ed.),The Handbook of Contemporary Semantic Theory.* Blackwell Publishing Ltd, Oxford, UK, 321-341.

丁声树等. 2004. 现代汉语语法讲话. 商务印书馆, 北京, 200-202.

方绪军. 2017. *“不是不X”*、*“不是没(有)X”*和*“没(有)不X”*. 语言科学, 16(05):511-521.

符达维. 1986. 对双重否定的几点探讨. 福建论坛*(*文史哲版*)*,(6):78-81.

何爱晶. 2019. 反叙的非真值义否定和真值义肯定. 外语研究, 36(04):24-29.

郎桂青. 1989. 双重否定句表示肯定的条件. 语文研究,（1）:28.

李新良, 王明华. 2015. 汉语动词的叙实性研究的应用前景. 对外汉语研究, (02):120-129.

鲁晓琨. 2004. 现代汉语基本助动词语义研究. 中国社会科学出版社, 北京.

吕叔湘. 1956. 中国文法要略. 商务印书馆, 北京.

王力. 1943. 中国现代语法. 商务印书馆, 北京.

王勇, 吕学强, 姬连春. 2014. 基于极性词典的中文微博客情感分类. 计算机应用与软件,(01):40-43+132.

芜崧. 1987. 双重否定句的种类与功能. 荆州师专学报（哲社版）,(3)：52-57.

叶文曦. 2013. 否定和双重否定的多维度研究. 语言学研究,(2):20-31.

袁毓林. 1999. 并列结构的否定表达. 语言文字应用,(03):42-46.

袁毓林. 2000. 论否定句的焦点、预设和辖域歧义. 中国语文,(02):99-108+189.

郑贵友. 1989. 汉语“助动词”的研究刍议. 汉语学习,(06):23-27.

基于语料库的武侠与仙侠网络小说文体、词汇及主题对比分析

张三乐　　刘鹏远*　　张虎
北京语言大学　信息科学学院
国家语言资源监测与研究平面媒体中心
北京市海淀区学院路15号, 100083
sanle0409@163.com　liupengyuan@pku.edu.cn　1170226830@qq.com

摘要

网络文学在我国发展迅猛，其数量和影响力呈现逐年上升的趋势，但目前尚无公开的较大规模网络文学作品语料库，鲜见基于语料库对网络文学具体类别作品的定量研究。本文初步建立了一个网络文学语料库，其中包括武侠和仙侠网络小说，使用文本计量、词频统计以及主题挖掘的方法对两类小说的文体风格、具体词汇使用和小说主题进行对比分析。通过比较，我们发现两类小说的文体风格大致相同，它们在词汇的使用和主题上既有共性又各具特色。从微观到宏观，从表面到内容，将定量统计和定性分析相结合，多角度、多层次的对武侠和仙侠网络小说进行比较。

关键词： 网络文学；武侠小说；仙侠小说；文体风格；词汇使用；主题

A Corpus-based Contrastive Analysis of Style, Vocabulary and Theme of Wuxia and Xianxia Internet Novels

Sanle Zhang　　Pengyuan Liu*　　Hu Zhang
Beijing Language and Culture University, School of Information Science
Language Resources Monitoring and Reserch Center
15 Xueyuan Road, Haidian District, Beijing, 100083, China
sanle0409@163.com　liupengyuan@pku.edu.cn　1170226830@qq.com

Abstract

Internet literature is developing rapidly in our country, and its number and influence are increasing year by year. However, there is no publicly large-scale online literary corpus, and there are few quantitative researches on specific types of online literature based on corpus. This article has initially established a corpus of online literature, including Wuxia and Xianxia online novels, using text measurement, word frequency statistics and topic mining methods to compare the stylistic style, specific vocabulary use and novel themes of the two types of novels. Through comparison, we find that the styles of the two types of novels are roughly the same, and they share commonalities and distinctive features in terms of vocabulary use and themes. From the micro to the macro, from the surface to the content, it combines quantitative statistics and qualitative analysis to compare Wuxia and Xianxia online novels from multiple angles and levels.

Keywords: Internet literature , Wuxia Novels , Xianxia Novels , Stylistic style , Vocabulary using , Theme

* 通讯作者　Corresponding Author

1 引言

中国通俗文学发展至今，武侠小说始终是其中一个重要类别。金庸、古龙、梁羽生等老一辈的武侠大家所创作的作品曾引起社会上广泛的武侠小说阅读热潮，然而在这些武侠大家退隐之后，武侠作品的创作就陷入低潮期(张珍珍, 2017)。随着网络时代的来临，网络文学在此背景下逐步发展起来，至今已有20余年，其内容和影响力呈现逐年上升的趋势。网络的普及以及数字阅读平台的构建给网络小说提供了创作依托，大批网络写手利用网络平台发表自己的作品，受到广大读者的喜爱与追捧，涌现出大批网络原创小说。新时代网络作者接过金庸、古龙的接力棒，将武侠作品融入新时代元素，使其再度成为一大热潮，同时，由于IP改编影视剧的影响，仙侠小说逐渐走进大众视野，不仅在国内有一大批读者，在国外也很受欢迎。

从文学发展的角度来看，仙侠小说直接脱胎于武侠小说，是在武侠小说的在基础上发展起来的一种新型小说类型，在各大网络文学网站上的点击阅读量居高不下。虽然仙侠小说在网络小说中数量庞大，广受读者欢迎，但是关于它的研究仍处于网络文学研究的边缘地带，与它的发展不匹配(段晓云, 2018)。时至今日，网络文学的研究已取得了较大成就，涌现出一批代表性学者，如黄鸣奋、欧阳友权等，对网络文学的研究做出了巨大的贡献。目前,国内对于网络文学的研究一般以西方的理论研究为基本背景，研究着眼点放在网络文学的个别文本(崔宰溶, 2011)，且研究角度集中在文艺批评、文学特色和文化产业等方面，从定性的角度研究网络文学的特点，多把网络文学当做整体进行研究探讨，尚无公开的较大规模的网络文学作品语料库，鲜见学者基于大规模语料对网络文学具体类别的作品进行定量方面研究。

本文初步建立了一个网络文学语料库，语料来源于国内最大、最有影响力的网络文学网站———起点中文网[0]。该语料库目前包含网络武侠小说和仙侠小说，每种语料大小各100M byte，分别约2380万和2440万词次。基于这个语料库，本文对两类小说文本进行文本计量、词频统计分析以及主题挖掘，试图回答以下问题：

1）在宏观上，网络武侠与网络仙侠两类小说在文体风格上是否相同？为什么？
2）在微观上，两类小说在具体词汇的使用上有哪些异同？各有何种特色？
3）在内容上，两类小说在小说主题上有哪些异同？

2 相关工作

计量风格学产生于1851年英国数学家和逻辑学家Augustus De Morgan的猜想，他认为不同作家的作品风格可以通过隐形的数据特征进行辨别(Herdan, 1964)。近年来，计量风格学更广泛的应用于现当代文学研究领域。

在国外，Chaski (2001)从句法、标点符号、句子复杂度、文本易读性等方面对四位同龄女作者的部分作品进行了分析和比较；Argamon and Levitan (2005)等人认为功能词最容易反映作者的语言风格，并提出了675个能够反映作者风格的功能词；Grieve (2007)则以词首、词尾中字母的频率和包含各个字母的单词频率作为特征对作品进行分析等。在国内，刘颖and 肖天久 (2014)运用文本聚类和N元文法对词长分布、词类等语言特征进行考察，发现《红楼梦》前八十回和后四十回存在较大差异，得出其非一人所作的结论；金迪 (2018)采用数理统计学中定量分析的手段和方法，以计量风格学的视角，从频率统计和假设检验两大角度探究余华和格非小说在词汇和句子层面上的差异，从而分析二者的语言风格。

众多计量风格学的研究是在语料库的基础上对所选择的语言特征项进行分析，语料库在研究中起到了重要的作用。20世纪80年代开始，将语料库运用到文学作品中的研究逐渐升温，为文学研究提供了一个全新的视角，基于语料库的文学研究这一具有鲜明实证研究特征的文学研究领域应运而生(胡开宝and 杨枫, 2019)。

在国外，Stubbs (2005)以康德拉小说《黑暗之心》为语料库进行研究，发现其主题词为不确定的实词、虚词、以及抽象名词和带有否定前缀形容词的名词词组；Mahlberg (2007)以狄更斯的23部作品为语料库，分析其中的高频词簇，发现和身体部位相关的词簇常常可以推动故事情节发展。在国内，刘宇凡et al. (2011)等人将唐代以来的文学作品按不同时期分类建立语料库并对其进行字频分析，发现唐代以来人们使用汉字的习惯处于不断变化之中，时期越相近，汉字的使用习惯越一致; 陈建生and 王岩 (2016)将厄普代克所著“兔子系列”小说语料库与厄普代

[0]起点中文网：www.qidan.com。

克其他类型小说语料库中的关键词进行比较分析，发现第三人称代词高频出现，且多与心理动词搭配。涂梦纯and 刘颖 (2019)以余华和莫言的各5部小说为语料库，对二者的用词特征进行详尽的分析，讨论了量词、拟声词等词，发现了莫言用词丰富、情感充沛以及文言化、乡土化的特征，而余华与之相比白话、冷静、讽刺的风格。

由上可知，随着时代的进步和技术的发展，计量风格学和语料库语言学越来越多的应用于文学作品研究领域，为文学作品研究提供了新的视角和方法，但研究多集中在传统经典的文学作品，对新兴的网络文学领域却极少涉及。

在网络文学研究方面，欧美的网络文学研究成果已经相当可观，他们的网络文学研究趋势集中在文学实验的理论性研究。Bolter (1991)创造性的用解构主义理论观照“超文本”得叙事特点，指出超链接使“超文本”具有多重阅读路径，彻底颠覆了传统的线性阅读的叙事模式，从而体现了解构主义取消中心、无限变更的开放活动的观念；Aarseth (1997)提出“制动文本理论”，为广大网络文学研究者提供了新的视角，掀起了网络文学研究的热潮；近年来，德、美合作出版了网络文学研究系列论文集，至今已出版四集，从不同的视角对网络文学进行全面的探讨，代表着当今西方网络文学研究的前沿(王艳, 2016)，如Beyond the Screen。在国内，网络文学研究也在迅速发展。黄鸣奋 (2002)阐释了网络文学贵在鲜活、追求互动的网络根本特质；欧阳友权 (2004)在哲学意义上探讨网络文学，以“本体论”和“现象学”讨论网络文学的终极意义。后来，越来越多的学者从新颖的观点和角度对网络文学进行研究，如王黎 (2010)从女性主义的角度去分析网络文学；张珍珍 (2017)描述了网络武侠小说兴起的文化背景，讨论了网络武侠小说和传统武侠小说的内在关联，分析了网络武侠小说在类型创造、传播方式上的转变；段晓云 (2018)从文学空间切入，探讨网络仙侠小说中文学空间的描写。

综上所述，网络文学作为一种新兴的文学形式，广泛的受到人们的喜爱，也吸引了众多学者的目光。目前，网络文学定性研究在我国已取得不小的成就，这些研究多集中在文艺批评、文学特色和文化产业、基础学理研究等方面。然而，很少有学者从计量、统计等定量的角度对网络文学作品的特点进行探究，也尚无公开的较大规模的网络文学语料库。本文初步建立了一个网络文学语料库，包括网络原创武侠小说和仙侠小说，使用计量、词频统计和主题挖掘的方法，多层次、多角度的对这两类网络文学文本的特点进行探究。

3 数据

本文初步构建了一个网络文学语料库，语料库构建步骤大致如下：

- 数据获取。编写爬虫软件，随机抓取了起点中文网中上架于2015年至2017年、作品分类为“武侠”和“仙侠”的两类小说。
- 数据清洗。对每一类别抽取了100M字节的文本，对其中存在的网站标语、乱码等进行处理，对语料进行整理，去掉其中的重复行，将标点符号由半角转为全角。
- 分词和词性标注。整理完语料之后，采用分词和词性标注工具jieba[1]对文本进行分词和词性标注。由于网络小说中的词汇和句法较为随意，会有一些未登陆词，在后续实验中不予考虑。

最终，本文的研究数据基于上述语料，共约7000万字，语料库数据基本情况如表1所示。

Table 1: 网络文学语料库数据分布

文本类别	总字数	总词数	标点数	句子数	段落数
武侠小说	35825816	24405279	30041194	1043739	369989
仙侠小说	35034702	23845448	29511768	1027756	364526

4 文本风格比较

本文从四种最常见的计量指标以及词类的分布对网络武侠小说和仙侠小说进行了风格计量，并使用统计检验的方法检验二者之间是否存在显著性差异，从而比较二者的文体风格。

[1]https://github.com/fxsjy/jieba

4.1 基本计量指标比较

本文从词汇丰富度、文本可读性、句子离散度、句子破碎度和词类分布这些角度进行计量比较：

- 词汇丰富度

  词汇丰富度是指作者在文本中使用词汇的丰富程度。本文选择词频广度和型例比作为考察词汇丰富度的参数特征。词频广度即计算高频词之外的词语的比例，本文设定高频词为覆盖率90%的词语。型例比，是指文本中不同的词语在所有词语中所占的比例，公式如下：

$$TTR = Types/Tokens$$

- 文本可读性

  文本可读性最初由教育学家Dale提出。文本可读性高可理解为文本简单，文本复杂度低。文本可读性可以从平均词长、平均句长来初步考察。平均词长是所用词汇的平均字数，平均句长是所用句子包含的平均字数。

- 句子离散度

  句子离散度即文本中句子的句长偏离平均句长的长度，可以提现出文本节奏，计算公式如下：

$$D = \sqrt{\frac{1}{n}\Sigma\left(L_i - L_0\right)^2}$$

  公式中D表示句子离散度,L_i表示每句句长，L_0表示平均句长，n表示句子的总数。

- 句子破碎度

  破碎度即在句子中停顿的次数，可以侧面体现文本的语体色彩。计算公式为：句子破碎度=句子停顿次数[2]/句子总数。

我们在对网络武侠小说和仙侠小说进行基本计量之后，将两类小说各平均分成十份小语料，每份约10M，使用spss软件对每一类计量指标结果进行独立样本T检验，两类语料计量和统计检验结果见表2。

Table 2: 网络文学文本计量数据及统计检验结果

		武侠小说	仙侠小说	P值
词汇丰富度	型例比	0.151	0.186	0.541
	词频广度	0.875	0.962	0.259
文本可读性	平均词长	1.468	1.469	0.076
	平均句长	34.324	34.089	0.831
句子离散度		37.500	24.909	0.042
句子破碎度		2.424	2.328	0.225

通过表格可以看出，网络武侠小说和仙侠小说的型例比、词频广度、平均词长、平均句长和句子破碎度的P值都大于0.05，没有达到显著水平，即两类小说的词汇丰富度、文本可读性、句子破碎度没有显著差异。

在此之外，两类小说句子离散度的P值为0.042，说明两类小说的句子离散度存在显著差异。通过比较，发现武侠小说的句子离散度比仙侠小说更高，说明武侠小说中句子长短不一，跳跃性大，使文本富于节奏变化，使读者阅读起来长短不一、错落有致，有着抑扬顿挫、跌宕起伏的感受，阅读体验感强；而仙侠小说的句子离散度较低，说明其文章节奏较为缓和，读者阅读时有较为严肃、平缓的体验感。

[2]黄柏荣and 廖序东 (2002)在《现代汉语》中指出，“点号主要用来表示句子中的各种停顿”，并将点号分为句中点号（逗号、顿号、分号和冒号）以及句末点号（句号、感叹号、问号）。本文在计算句子破碎度中采用了这一标准。

4.2 词类分布比较

词类是词的语法分类，是词在语法结构中表现出来的类别。不同的词在文本中起着不同的作用，在文本风格分析中词类的使用频率是构成文本风格的重要特征之一[3] 。我们使用编程对网络武侠小说和仙侠小说的部分词类进行统计，然后同样将两类小说各平均分成十份，使用spss软件对其进行独立样本T检验，从而比较两类小说的异同。

Table 3: 网络文学文本的词汇分布和统计检验结果

实词	武侠小说	仙侠小说	P值	虚词	武侠小说	仙侠小说	P值
名词	0.1255	0.1242	0.165	拟声词	0.0005	0.0007	0.068
动词	0.1188	0.1193	0.464	连词	0.0170	0.0171	0.064
形容词	0.0176	0.0189	0.046	介词	0.0167	0.0170	0.771
数词	0.0240	0.0253	0.011	助词	0.0460	0.0510	0.175

由上可知，经过统计检验，网络武侠小说和仙侠小说的名词、动词、拟声词、连词、介词和助词的P值都大于0.05，说明两类小说在这些词类的使用频率上没有显著差异；形容词和数词的P值小于0.05，说明两类小说在形容词和数词的使用频率上差异较为显著。

形容词在汉语中充当修饰成分，在小说中的作用主要是对人物的刻画、对环境的渲染以及对场景的描写。数词使用时常常与量词搭配作定语，也起到修饰的作用，在小说本文中使用数词，说明其更加注重细节描写，给人以真实感。由表可知，仙侠小说的形容词和数词使用频率较高，如：

例：一缕琴声随风传来，缓如溪水流泉，脆如珠落玉盘，叮叮咚咚空灵有质。随着琴声渐清，一丝歌声却是悱恻辗转，酥人心扉.....（选自仙侠小说）

说明仙侠小说中人物刻画和环境描写更加丰富、细致，使文章内容生动形象，提高了读者的阅读体验，更能吸引读者眼球，一定程度上说明了仙侠小说越来越受读者喜爱的原因。

总之，通过对网络武侠小说和仙侠小说进行计量和统计检验，发现除了句子离散度、形容词和数词使用频率有差异之外，其他指标都没有显著差异，说明两类小说的文体风格基本相同。这可能是由于仙侠小说是在武侠小说的基础上发展起来的，且二者都属于网络文学文本，其用词、用句、行文结构等都大致相似，因而文体风格上没有显著差异。

5 具体词汇使用比较

本文从词频统计的角度考察网络武侠小说和仙侠小说的词汇使用情况，探究两类小说的用词风格。由于实词在文本中主要承担表达意义的作用，有具体的词汇意义，所以我们选择使用频率较高且有丰富词义的名词、动词、形容词进行对比分析。根据语料库的词性标注结果，我们通过编程抽取了两类小说的名词、动词和形容词，去除停用词、人名、地名以及“说、有”之类的常用词，统计两类小说按照频率排序的共用词和独用词的使用情况。

5.1 不同词类下的共用词

按照频率列出前500个高频词中名词、动词和形容词的共用词，各取前20词。统计结果如下：

通过对不同词类的共用词进行比较，可以发现网络武侠小说和网络仙侠小说在具体词汇使用上的共性。我们将这些高频的共用名词、动词按照语义进行粗略分类，名词可以分为武功武器词、身体部件词、人物关系词以及其他，动词可以分类为使令动词、动作动词以及心理动词。另外，按照郭伊迪 (2012)中的分类将统计出的共用形容词进行分类，可以分为度量形容词、情绪形容词和色彩形容词。

在网络武侠小说和网络仙侠小说的共用名词中，人物关系词占比最高，符合小说注重刻画人物关系的特点。在人物关系词中，如“弟子、师父、大哥”等，发现描写对象多为男性，而且关系多为师徒、父子、师兄弟关系。这说明男性往往是小说中的主要角色，且对师徒、父子和兄弟情义描写较多，突出了两类小说中侠肝义胆的人物情感和侠义色彩。在武功武器词中，“剑”的使用频率最高，其他词也多与“剑”相关，如“剑法、剑气、长剑”，另外还有“刀、气

[3]道格拉斯·比伯、苏珊·康拉德、兰迪·瑞潘 (2012)《语料库语言学》，刘颖、胡海涛译，北京：清华大学出版社，2012年，第43页。

Table 4: 武侠和仙侠网络小说按照频率排序的共用词

名词	武功武器词	剑、刀、剑法、长剑、剑气、气息
	身体部件词	手、眼睛、脸、
	人物关系词	弟子、师父、大哥、前辈、父亲、师弟、长老、掌门
	其他	风、山、门派
动词	使令动词	派、令
	动作动词	死、杀、跑、出手、救、抓住、拉、打、修炼、盯、追、愣、跳、躲
	心理动词	怕、担心、放心、喜欢
形容词	度量形容词	深、快、大
	情绪形容词	好、平静、激动、紧张、难、简单、轻松、诡异、干净、厉害、尴尬、逍遥、犹豫、强大
	色彩形容词	白色、红色、黑色

息”，说明两类小说中最常使用的武器是“剑”，“刀”次之，而且都注重内功。在身体部件词中，如“手、眼睛、脸”，说明其细节描写丰富。在其他词中，“门派”一词符合两类小说的特点，有共同信仰和武功继承的人同处于一个派系，门派和门派之间形成敌友关系，构成小说的故事网络。另外，其他词中还包括“风、山”这样的自然风景词，表明小说环境烘托较为丰富，提高了作品的阅读性。

在两类小说的共用动词中，动作动词占比最高，符合小说注重动作刻画的特点。两类小说使用了大量相同的动作动词，说明二者在动作情节的描写上有一定的相似性，例如“死、杀、出手、救”，说明二者都有着极为丰富的打斗、征战、死伤的情节；又如“打、跑、抓住、拉、追”等，用不同的动词细致的表示不同的动作形态，丰富了小说内容；对人物表情也有一定描写，如“盯、愣”；值得注意的是“修炼”一词，一般在文中搭配“武功、法力”，体现出二者对武功描写着墨较多。在心理动词中，如“怕、担心、放心”等，说明其注重心理描写，有着较为丰富的情绪表达。另外还有一些使令动词，从中可以看出“派遣”、“命令”的行为较多，如“派、令”，也表明这两类小说中普遍存在着地位等级关系。

在两类小说的共用形容词中，情绪形容词的占比最高，其中有一些对人物情绪的刻画，如“平静、激动、紧张”等，还有一些对事物的主观判断，如“好、简单、难”等。度量形容词的高频使用说明小说中有一些对事物性质的叙述，如“深、快、大”等。另外，两类小说中的还有一些色彩描写，“如白色、黑色、红色”，丰富了小说色彩，增强了小说的趣味性和形象性。

总之，通过比较发现，网络武侠小说和网络仙侠小说都注重人物刻画和动作描写。两类小说的男性角色较为突出，普遍存在着地位等级关系；在动作描写上有一定的相似性，常用“剑”作为武器，有着丰富的武打情节，注重细节刻画和环境烘托。

5.2 不同词类下的独用词

按照频率列出前500个高频词中名词、动词和形容词的独用词，各取前20词。统计结果如下：

通过对不同词类的独用词进行比较，可以看出网络武侠小说和网络仙侠小说在具体词汇使用上各具特色。同样，我们将这些高频的名词、动词和形容词的独用词按照语义进行粗略分类，名词可以分为武功武器词、门派派系词、人物动物词以及其他，动词全部划分为动作动词，形容词划分为颜色形容词和情绪形容词。

在网络武侠小说和网络仙侠小说各自的独用名词中，我们可以发现网络武侠小说中多使用冷兵器和内功，例如“刀法、枪、内力、轻功”等；而网络仙侠小说则多使用法力、法术，如“灵力、法术、法力”等，体现出武侠小说传统、写实和仙侠小说创新、虚幻的特点。由上共用词可知“门派”一词的在两类小说中得到高频使用，在独用词中我们就可以看出两类小说在门派派系词在具体使用上的不同。可以看出武侠小说中的门派多使用“盟、教”构词，如“教主、红衣教、金兰盟”等，而仙侠小说则多使用“宗、族”，如“宗门、人族、魔族”，具有神化色彩，另外，武侠小说的门派词使用要多于仙侠小说，说明武侠小说中的派系较多。在人物动物词中，武侠小说中使用了“公主、陛下、夫人”等词，体现出其真实性和历史性；而仙侠小说中人造、虚构的人物较多，例如“仙人、凡人、女娲”等，除了人物词，仙侠小说中的动物词使用要多与武侠小

Table 5: 武侠和仙侠网络小说按照频率排序的独用词

		武侠小说	仙侠小说
名词	武功武器词	暗器、刀法、穴道、枪、兵刃、内力、修为、轻功	灵力、法术、法力、炼气、法器
	门派派系词	教主、红衣教、丐帮、盟主、金兰盟	宗门、人族、魔族
	人物动物词	公主、女儿、老头、夫人、陛下、猴	仙人、凡人、女娲、妖兽、鹤、妖怪、僵尸、野兽、
	其他	镖局	灵魂、宝物、血影、血光
动词	动作动词	争宠、娶、围攻、切磋、厮杀、打擂、拔剑、逃走、下马、联姻、拿下、联手、扫视、飞扬、失踪、打探、追赶、出鞘、刺杀、埋伏	修行、飞行、叫魂、飞升、尸变、提升、炼丹、炼制、吸收、吞噬、穿越、渡劫、传送、喷出、斩杀、砸断、燃烧、弥漫、毁灭、下狱
形容词	色彩形容词	深红、暗橙	金色、紫色、蔚蓝、血色、丹色、青色
	情绪形容词	慈悲、仗义、娇羞、柔情、冷峻、胆怯、敏捷、刚猛、焦躁、羞涩、精明、妩媚、孤傲、肃然、稀奇、雄厚、精壮、悦耳	混沌、硕大、古朴、清凉、浓郁、浩瀚、清冷、稚嫩、强横、充沛、炽热、轻易、坚挺、猖狂

说，如“妖兽、鹤、野兽”等，在现实的基础上进行虚构，体现出其创造性和奇幻、志怪色彩，提高了小说的趣味性。在其他词中，武侠小说中有“镖局”一词，同样体现其写实的特点；仙侠小说中有“灵魂、宝物、血影、血光”，这些字眼更具有暗黑色彩，营造出恐怖神秘的场景。

在两类小说各自的独用动词中，可以发现，武侠小说更加注重对打斗方式的描写，如“围攻、刺杀、拔剑、追赶”等，更关注传统武打和江湖世界；在武侠小说中构造的故事更加贴近生活，如“争宠、娶、打擂、联姻”，体现了当时的社会特色，重现了当时的社会场景，具有社会性和真实性，拉近了读者与故事的距离，产生亲切感。仙侠小说则不同，仙侠小说中使用了一些动能较高的动词，如“斩杀、砸断、吞噬”，给人以冲击感；在武功方面突破传统武功技法，如“叫魂、飞行、飞升”，还使用了如“尸变、穿越、渡劫”等词，突破现实世界，具有虚构色彩，给人以新奇之感。

在两类小说各自的独用形容词中，我们可以发现，武侠小说中的情绪形容词中多是对人物情感、性格的描写，如“慈悲、仗义、娇羞”等，而仙侠小说则多为对事物的主观判断，如“混沌、硕大、古朴”。另外，两类小说含有一些色彩形容词，如武侠小说中有“深红、暗橙”，仙侠小说中则更加丰富，如“金色、紫色、蔚蓝”等颜色词，颜色词层次更加丰富，种类较多，为读者构建了色彩斑斓的世界，增强了小说的趣味性和形象性，提高了读者的阅读体验。

总之，通过比较，可以看出两类小说在具体使用词汇上的不同。网络武侠小说更加关注传统武打和现实世界，多使用冷兵器和内功，门派多为“盟、教”，且派系相对较多，具有真实性、社会性和历史性，还将笔墨更多的用于人物刻画，让读者从字里行间体会人物情绪；网络仙侠小说则突破传统武功技法和现实世界，多使用法力、法术，门派多为“宗、族”，具有创造性、虚构性和灵异、志怪、暗黑色彩，常给小说营造出恐怖神秘的场景，注重人对事物的描写和对周边世界的评价，还使用了较为丰富的色彩形容词，更具趣味性。

6 主题比较

LDA(latent dirichlet allocation,隐狄利克雷分配)，是一种采用词袋模型的文档主题生成模型，能够把文本中的主题自动汇集。它的基本思想是假设所有的文档存在K个隐藏主题，一篇文档的每个词都是以一定概率选择了某个主题，并从这个主题中以一定概率选择了某个词语，不断的抽取隐含主题及其特征词，直到遍历完文档中的全部单词。

本文使用LDA模型对网络武侠小说和仙侠小说两类文档集合进行主题建模，挖掘文本隐含的主题信息，对两类小说的主题进行比较。

6.1 语料预处理

在原有语料分词和词性标注的基础上，我们去除了语料中的停用字，过滤了标点符号以及无意义的词，如数词、量词、虚词等等，以避免对模型最终结果的影响，降低噪音。然后，根据小说文本的特点，把小说按照不同的章节分隔开来，作为独立的文档。

6.2 主题比较分析

我们使用LDA模型对两类小说进行主题建模，指定主题数为150个，设置每个主题打印出最能描述该主题的前20个词。由于篇幅限制，将其中部分主题进行总结并制成词云进行比较，图1是武侠小说的两类主题，图2是仙侠小说的两类主题。

Figure 1: “受害”和“敛财”主题

Figure 2: “修炼”和“降妖”主题

根据主题聚类的结果，我们将武侠小说中的主题总结为：受害、敛财、出征、联姻、门派等，将仙侠小说中的主题总结为：修炼、降妖、打斗、魔界、天界等（见附录）。

可以发现，网络武侠小说和仙侠小说在主题上有一定的相关性，比如两类小说中都出现了有关“武打”的主题，如“受害”和“打斗”主题。在这类主题中出现了许多关于细节描写的主题词，如“转身、胸骨、瞳孔、脚尖”以及“眼珠、汗毛、双手”等等，说明两类小说在描写打斗场面时会细化到人物的面部表情和身体变化；又比如两类小说中都出现了有关“派别”的主题，网络武侠小说中有“门派”主题，而仙侠小说中则有“魔界”和“魔界”主题，也体现出一定的相关性和对应性。

同时，两类小说的主题也有不同。网络仙侠小说中的主题多与“仙、魔”相关，主题词中也多以“仙、魔”构词，如“成仙、魔头、魔神”等等，说明仙侠小说多以神、人、魔三界为背景讲述故事，着重“修仙练功”和“降妖伏魔”等情节，更加新颖、新奇，具有神化色彩；而网络武侠小说的主题涉及范围较广，除了武功、打斗之外，还涉及到钱财、婚姻、君臣等社会的各个方面，说明武侠小说多以人民生活为背景讲述故事，各个场景都与人民生活息息相关。如“敛财”主题，其中有“打耳光、上交、商人”等主题词，更加传统、熟悉，具有现实色彩。

总之，通过对两类小说的主题进行挖掘，我们发现两类小说在主题既有一定的对应性，也有一些不同之处，从二者的主题词中也可看出两类小说的异同。两类小说都出现了一些相似的

主题，但是网络武侠小说的主题涉及范围更广，而网络仙侠小说的主题则更加集中。另外，两类小说都注重细节描写，但是网络武侠小说的主题词更加传统、熟悉，具有现实色彩，而仙侠小说的主题词更加新颖、新奇，具有神化色彩。

7 结论

网络文学的发展至今已有20余年，随着网络的普及以及数字阅读平台的构建，网络文学以其方便、经济的特点越来越受到人们的关注和喜爱，其内容和影响力呈现逐年上升的趋势。

本文建立了一个网络文学语料库，包括武侠和仙侠网络小说。通过对两类小说文本进行计量、词频统计以及主题挖掘，从宏观到微观，多层次、多角度的比较了武侠和仙侠网络小说的异同，回答了我们提出的三个问题。

从宏观上，我们对网络武侠小说和网络仙侠小说进行计量风格分析，发现两类小说的文体风格基本相同，这可能是由于仙侠小说是在武侠小说的基础上逐步发展起来的，用词、用句、行文结构都大致相似，因而文体风格没有显著差异。

从微观上，我们对两类小说的名词、动词和形容词进行统计分析，发现两类小说在具体词汇使用上既有共性又各具特色。从两类小说使用的词汇上可以看出，两类小说都注重人物刻画和动作描写，小说中男性角色较为突出，且普遍存在地位等级关系；有丰富的武打情节，在动作描写上有一定的相似性，常用“剑”作为武器，注重细节刻画和环境烘托，具有集体性、等级性和侠义色彩。网络武侠小说更加写实、传统，具有真实性、社会性和历史性，还将笔墨更多的用于人物刻画，让读者从字里行间体会人物情绪；网络仙侠小说则突破传统武功技法和现实世界，关注异世界，多使用法力、法术，具有创造性、虚构性和灵异、志怪、暗黑色彩，常给小说营造出恐怖神秘的场景，注重人对事物的描写和对周边世界的评价，还使用了较为丰富的色彩形容词，更具趣味性。

从内容上，我们使用LDA主题模型对两类小说进行主题挖掘，发现两类小说主题的异同。两类小说的主题具有一定的相关性，也有一些不同之处。两类小说出现了一些类似的主题，然而武侠小说的主题更加广泛，主题词更加传统、写实，具有现实色彩；而仙侠小说的主题则更加集中，主题词更加新颖、新奇，具有神化色彩。同时，也表明LDA模型可以应用于大规模小说语料的主题挖掘。

在后续工作中，我们将进一步扩充网络文学语料库，多角度、多层次，使用多种方法对各类网络文学进行研究。

致谢

感谢各位匿名评审老师和论文辅导老师的帮助。本论文受教育部人文社会科学研究规划基金资助项目（18YJA740030）和北京语言大学研究生创新基金项目（20YCX153）资助。

参考文献

Espen J Aarseth. 1997. *Cybertext: Perspectives on ergodic literature*. JHU Press.

Shlomo Argamon and Shlomo Levitan. 2005. Measuring the usefulness of function words for authorship attribution. In *Proceedings of the 2005 ACH/ALLC Conference*, pages 4–7.

Jay D Bolter. 1991. *Writing space*. Erlbaum.

Carole E Chaski. 2001. Empirical evaluations of language-based author identification techniques. *Forensic Linguistics*, 8:1–65.

Jack Grieve. 2007. Quantitative authorship attribution: An evaluation of techniques. *Literary and linguistic computing*, 22(3):251–270.

Gustav Herdan. 1964. Quantitative linguistics.

Michaela Mahlberg. 2007. Clusters, key clusters and local textual functions in dickens. *Corpora*, 2(1):1–31.

Michael Stubbs. 2005. Conrad in the computer: examples of quantitative stylistic methods. *Language and Literature*, 14(1):5–24.

刘宇凡, 郭金忠, and 陈清华. 2011. 唐代以来汉语文学作品中的字频演变. 中文信息学报, 25(3):93–98.

刘颖and 肖天久. 2014. 《红楼梦》计量风格学研究. 红楼梦学刊, (4):25.

崔宰溶. 2011. 中国网络文学研究的困境与突破. Ph.D. thesis, 北京大学博士学位论文.

张珍珍. 2017. 网络武侠小说的发展及其特色. Master's thesis, 青海师范大学.

欧阳友权. 2004. 网络文学本体论纲. 文学评论, 6:69–74.

段晓云. 2018. 网络仙侠小说文学空间研究. Master's thesis, 兰州大学.

涂梦纯and 刘颖. 2019. 余华与莫言长篇小说的计量统计和分析. 中文信息学报, 33(2):131–142.

王艳. 2016. 西方网络文学研究综述. 创新与探索*:* 外语教学科研文集.

王黎. 2010. 女性网络文学作者的创作倾向. Master's thesis, 山东大学.

胡开宝and 杨枫. 2019. 基于语料库的文学研究: 内涵与意义. 浙江大学学报*(*人文社会科学版*)*, 5(5):130.

道格拉斯·比伯、苏珊·康拉德、兰迪·瑞潘. 2012. 语料库语言学. 清华大学出版社.

郭伊迪. 2012. 基于语义角度的形容词分类研究. Master's thesis, 黑龙江大学.

金迪. 2018. 基于语料库的格非, 余华小说计量风格学研究. Master's thesis, 南京师范大学.

陈建生and 王岩. 2016. 厄普代克"兔子系列" 小说特点的语料库文体学研究. 牡丹江大学学报, 25(9):22–24.

黄柏荣and 廖序东. 2002. 现代汉语.

黄鸣奋. 2002. 网络文学之我见. 社会科学战线, 4:15–16.

A 附录：武侠和仙侠小说的部分主题

	主题	武侠小说	主题	仙侠小说
1	受害	禀报、太子、刺杀、拦截、转身、陷阱、杀害、山贼、喷出、弥漫、瞳孔、纷飞、胸骨、脚尖、主持、凄厉、血水、功力、威胁、披风	修炼	师父、修炼、法宝、成仙、道友、真人、大师、教主、禅师、飞剑、光明、只能、施法、化作、祭炼、身体、飞升、化成、斗法、仙剑
2	敛财	防守、来路、打耳光、资金、抢夺、惯例、集散、上交、商人、主动权、缓和、余味、交给、鬼迷心窍、金子、引流、交入、小贩、呛着、紧跟	降妖	犹如、人族、抓住、危险、屋顶、金牌、村民、目光、击中、铃声、城墙、天师、妖兽、脸颊、飞刀、瞳孔、肩膀、菩萨、刀鞘、收服
3	出征	说道、安排、来到、皇上、格格、车马、师父、姑娘、战场、皇帝、回来、宝刀、说完、看着、离开、父亲、义军、吩咐、将军、总舵主	打斗	鲜血、眼珠、带头、弟弟、苍穹、汗毛、丧命、阻拦、直扑、双手、刺杀、飞剑、了结、流血、大片、穿过、带队、指使、飞掠、嚎叫
4	联姻	兵马、身份、去路、迎战、退身、女人、摆手、使臣、解除婚约、公主、嫁入、男人、马车、收买、帮忙、皇上、征讨、将领、边境、协议	魔界	弟子、光明、魔头、法力、神魔、炼成、放出、魔法、魔教、出手、佛光、灰尘、魔神、敌人、宝物、石像、神光、正宗、血影、尊者
5	门派	方志、师父、弟子、想到、不知、全真教、见到、功夫、不由、修习、神功、掌门、少林、想着、实在、丐帮、担心、徒弟、教主、内功	天界	祖师、修行、猴子、不由、佛祖、真人、修为、不知、看着、天地、人类、老道、古灵精怪、菩萨、天庭、微笑、混沌、师父、点头、雀儿

基于计量的百年中国人名用字性别特征研究

杜冰洁　　刘鹏远*　　田永胜
北京语言大学　信息科学学院
国家语言资源监测与研究平面媒体中心
北京市海淀区学院路15号, 100083
blcudbj@gmail.com　liupengyuan@pku.edu.cn　blcutys@gmail.com

摘要

本文构建了一个包含11万以上条目规模的中国名人人名数据库，每条数据含有人名、性别、出生地等社会文化标签，同时含有拼音、笔画、偏旁等文字信息标签，这是目前已知最大的可用于研究的汉语真人人名数据库。基于该数据库，本文从中选择1919年至今的人名，用定性与定量结合的方法探究人名中汉字的特征和其性别差异以及历时变化。从人名长度来看，男性人名比女性人名长；从人名用字的难易度来看，女性用字比男性更复杂；从用字丰富度来看，人名用字越来越单一和集中化，男性人名的用字丰富度大于女性人名。计算人名用字的性别偏度后发现女性人名的专用自更多。两性用字意象有明显的不同，用字的意象随着时间发生改变，但改变最明显的时间节点是改革开放前后，其中女性的变化比男性显著。除此之外，我们还得出人名中的性别极性字表、各个阶段的高频字表、用字变化趋势表等。

关键词： 中国人名数据库；汉字；性别差异；人名历时变化；定量分析

A Quantified Research on Gender Characteristics of Chinese Names in A Century

Bingjie Du　　Pengyuan Liu*　　**Yongsheng Tian**
Beijing Language and Culture University, School of Information Science
Language Resources Monitoring and Reserch Center
15 Xueyuan Road, Haidian District, Beijing, 100083, China
blcudbj@gmail.com　liupengyuan@pku.edu.cn　blcutys@gmail.com

Abstract

In this paper, a database of Chinese celebrities' names with a scale of more than 110,000 entries is constructed. Each data contains social and cultural labels such as names, gender, and birthplace, as well as Chinese character information labels such as Pinyin, strokes and character components, which is the largest known database of Chinese real people's names that can be used for research. Based on this database, this paper selects names from 1919 to the present and uses a combination of qualitative and quantitative methods to explore Chinese names in character characteristics, gender differences, and diachronic changes. Through research, it is found that the average number of Chinese characters in women's names is higher than that of men. The Chinese characters in female names are more complicated but lower richness than that of males. The use of personal names has become more monotonous and centralized with time. The imagery of Chinese characters used in the names of the two sexes is obviously different. The imagery of the characters changes over time, but the most obvious time point for the

* 通讯作者　Corresponding Author

change is around the reform and opening up, in which the change of women is more significant than that of men. Besides, we also obtained the gender polarity characters table of the names, high-frequency characters table of each stage, characters change trend table, etc.

Keywords: Chinese name database , Chinese Characters , Gender differences , Diachronic change of names , Quantitative analysis

1 引言

人名是不同个体为区分彼此而创造出的指称符号。人名既特殊又普遍，其特殊性表现在，人名属于词汇系统中专有名词的一种，具有指称的唯一性和确定性；其普遍性表现在人名在社会生活中的出现频率极高，在社会系统的正常运作中扮演着十分重要的角色，我们需要“说”名字，也需要“写”名字。与字母文字不同的是汉字具有表意的功能。因此中国人名不仅具有读音上的特殊性，在字形、字义上也具有特殊性，对于人名用字的研究也就显得十分重要。

本文建立了一个大规模中国名人人名数据库，从汉字本体的角度做了跨度长达百年的人名用字分析，同时从性别的角度展开，探究人名中汉字的性别差异。本文发现两性人名在长度、难易度、丰富度、变化趋势等方面都存在显著差异。本文贡献在于：1）建立了目前已知最大规模的真人人名数据库；2）分别从汉字本体及计量语言学两种研究视角进行了人名用字研究，这些研究方法应用到人名中被证明具有一定价值；3）得到了百年两性人名用字特征的差异与演变规律。

2 相关工作

对人名的语言学研究侧重语音、语义等。部分文献对于人名中的汉字，有所体提及。一些文献中(苏培成, 2001; 吴继章, 2001; 邱莉芹and 鞠泓, 2002; 张书岩, 1999; 张书岩, 2004)探讨了人名中出现的的生僻字、多音字、异体字等问题，提倡入名汉字应该规范化。赵越 (2006)、何晓明 (2001)提出中国人取名对合体字独体字等不同字形的讲究，遗憾的是作者并没有针对这一问题进行深入阐述。谢玉娥 (2000)、Jia and Zhao (2019)认为人名中的部分汉字具有性别偏向，但其讨论的主要是汉字的意义上的偏向问题，且没有做详细的定量统计和分析，也未从汉字本体的角度进行考察。关于汉字与性别之间的关系，韩燕et al. (2008)采用事件相关电位(ERP) 技术证明汉语人名具有性别刻板印象。王玉新 (2000)和潘世松 (2004)从汉字的偏旁结构和发展规律论述了汉字结构本身的性别歧视现象。

以上关于人名的研究多是基于几百上千条人名，样本数量较小。从研究方法来看，多是共时研究，或者两个时间段的对比研究。关于人名的历时研究时间跨度较小，难以宏观反映人名在一段时间范围内的变化。人名中的汉字研究较为单一，多是从汉字意象的角度进行解释，人名中汉字本身的特征研究几乎没有。而有关认知科学的实验又证明人名是具有一定刻板印象存在的，因此关于人名汉字的性别倾向研究可进一步探讨汉字性别歧视的现象。同时，目前缺乏一个公开的、可支持人名共时历时研究的中国人名数据库。

3 数据

3.1 人名数据库构建

本文基于知识图谱中的信息，来建立中国名人人名数据库。选择名人来建设人名数据库的原因有二：1.可最大程度上保证人名及相关信息的真实性；2.本文假设是否能成为名人与其姓名不相关，因此该语料库也可以支持对中国人名的其他研究。构建过程主要如下：

1）抽取。从百科人物知识图谱[0] 抽取了名人相关信息（这样可以较好的保证人名信息的真实准确性），抽取的条目具体为：姓，名，性别，出生日期，出生地；

2）筛选。原百科人物知识图谱中的每个名人信息分布混杂，并且有大量国外名人信息。为了最大化获得人名数据并且保证数据包含所需的几个维度，在抽取过程中我们主要通过姓名长

[0]http://openbase.openkg.cn/

度、出生地等筛选中国名人姓名。中国人姓名的主流格式是两个字的姓名，即姓+单名、三个字的姓名，即姓+双名，因此我们将姓名长度限定为两个字和三个字。在出生地方面，我们将关键词限定为中国所有省市，并将添加了“中国“村”等不同粒度的关键词。考虑到有些名人信息不包含出生地信息，我们又添加了“民族”这一判断规则；

3）信息补充。为丰富对中国人名的研究，我们又为每个名人条目补充了拼音、笔画、偏旁等信息。该信息来源于一个开源中华新华字典数据库[1]。对于部分不在字典数据库的汉字，我们利用人工的方法补充字典信息再进行匹配。

最终建成的中国名人人名数据库[2]共有111564个条目，每个条目包含姓、名、性别、出生地以及人名用字的拼音、笔画、偏旁等信息，其中有男性人名条目83706条，女性人名条目27858条。在这些条目中有54264条包含出生日期，时间跨度从古代至今，主要以近现代人名为主。该语料库可为中国人名多维度研究提供数据支持。

3.2 研究对象

从上述的人名数据库中抽取1919年至今等性别比例的人名作为研究对象[3]，从汉字本体的角度探究人名的长度、人名汉字难易度、人名用字变化在两性中的差异及其历时发展变化规律，从侧面了解近一百年中国社会政治、经济、文化的发展变化。

所选时间段中的人名中共出现了2342个字种，男性人名中的字种有1800个，女性人名中的字种有1807个。之所以选择这一时段是因为本文希望对近现代近100年的人名从汉字的角度做定量分析，而1919年作为近代史开端，自然成为本次研究的时间起点。本文对数据做两种划分，详见表1：

按自然年份划分	1919-1938	1939-1958	1959-1978	1979-1998	1999至今
男名	1168	2637	5275	3742	106
女名	1202	2670	5208	3624	224
按重要历史事件划分	1919-1949	1950-1965	1966-1978	1979至今	-
男名	2279	3689	3112	3848	-
女名	2279	3689	3112	3848	-

Table 1: 研究对象的历时划分

1）按照自然年份划分。本文希望能以时间均匀的角度观察人名变化的规律。每二十年一个阶段划分，共五个阶段；

2）按照重要历史事件为时间点进行划分。本文假设重大政治经济事件对人们起名的影响较大。这个方式主要用于对自然年份划分的对比和说明。

基于中国百年人名数据库及表1的划分，文本希望能揭开百年来中国人名变化趋势的一角，并试图回答以下问题：1）人名在长度上是否有性别差异？百年来人名长度变化情况与原因？2）在用字难易度上是否有性别差异？百年来人名用字难度变化情况与原因？3）在用字丰富度上是否有性别差异？百年来人名用字丰富度的变化情况及原因？4）具体用字是否在时间维度上有显著差异？百年来人名具体用字变化情况及原因？

4 人名用字性别差异及历时分析

4.1 名字长度

文字同语言一样是一种信息交流的工具，人名中的汉字是记录人名内涵的书写符号。很多汉字都能独立表达一定含义，人名的内涵可以通过每个汉字的排序、人名汉字的多少（即名字长度）等来传达。本文将只有一个字的名字称为单名（如：杜甫），两个字的名字称为双名（如，周树人）。

[1]https://github.com/HelloDreamen/chinese-xinhua。共收录了16142个汉字的相关信息。

[2]https://github.com/NLPBLCU/Chinese-Celebrities-Names

[3]本文仅考虑常规两个字和三个字的姓名。对于传统的复姓，因样本相对较少，故不在研究范围内。近些年出现的类似“王张”的双姓，本文当做单姓处理。

我们计算了数据中两性人名的平均长度，其中女性人名长度均值2.88，男性人名长度均值2.91。随后做了皮尔逊卡方检验[4]，结果表明，男女人名中的单名和双名分布存在统计学意义上的差异。具体统计结果见表2:

	值	自由度	渐进显著性（双侧）
皮尔逊卡方	75.287[a]	1	.000
连续性修正[b]	74.937	1	.000
似然比	75.565	1	.000
有效个案数	25856	-	-
a. 0 个单元的期望计数小于5,最小期望计数为1374.00;b. 仅针对2x2 表进行计算			

Table 2: 性别与人名长度卡方检验

为了解各个阶段单双名的具体分布情况，下图从自然年份的划分观察近百年来中国人名长度的变化:

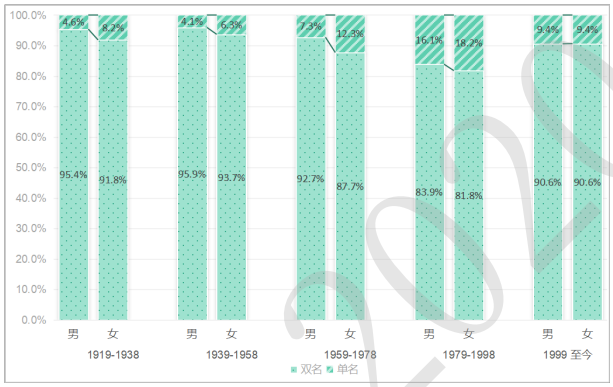


Figure 1: 自然年份中两性人名单双名的变化趋势

由图1可知：1）女性人名中的单名比例一直高于男性，但是两性人名中的单双名差异随着时间的发展不断缩小；2）总体而言，人名中单名双名的比例呈现一定波动，但是双名占据绝对优势；3）1979-1998区间单名比例是一个峰值，但是进入21世纪，单名比例又有所降低，几乎与1919-1938区间单名比例持平。

双名是魏晋门阀制度盛行，强调宗族家谱以后才逐渐占据主流的。因为族谱的存在，在取名时中间一个字需要固定，因此中国人名大多数是双名，而家谱和宗法制度有着密切的联系，宗法制度强调与家族中男性长辈的血缘亲疏，其主要精神为“嫡长继承制”。这就会对家族中男性晚辈身份的约束。所以男性的双名比例始终高于女性，反之女性单名比例高于男性。即使除去按字辈取名的习俗，双名有两个汉字承载的信息量也大于单名，增加名字内涵的同时避免了重名的概率，所以双名一直占据主流地位。但是随着近现代中国各种思想解放运动展开，一定程度上打破了传统的宗法制度，按照字辈取名的习俗也逐渐减少。原本按照字辈取名的双名，实际上只添加了名字末尾的一个新信息，如今中间的字没有了，依然只需要添加一个新信息，所以单名的比例就呈现增高的趋势。在改革开放初期，随着思想观念的进一步解放以及追求个性的心理特征，单名比例不断增长，在1979-1998年到达高峰。但是进入21世纪，随着人口的增多、姓名规范意识的增强，单名比例又开始下降，双名逐渐增多。

4.2 用字难易度

本文的难易度仅指书写难易程度或认读的难易程度。我们采用汉字常用等级这一指标衡量人名汉字的难易度，并以国家语言文字工作委员会1988年1月发布《现代汉语3500常用字表》作为标准。该字表中的字主要满足基础教育和文化普及的基本用字需要，因此人名中的汉字若是来自于该字表，则用字在认读或书写上相对简单。该表中的汉字有一级常用字和二级常用字之分，对于不在字表中的字统称为用非常用字，即用字相对复杂。

[4]本文所有统计检验均采用SPSS Statistics 25.0.0软件计算得出。

按照性别分组，采用卡方检验验证用字难度是否存在统计学意义上的差异，结果表明，男女人名用字的难度具有统计学意义上的差异。具体统计结果见表3:

	值	自由度	渐进显著性（双侧）
皮尔逊卡方	899.676[a]	2	.000
似然比	908.557	2	.000
有效个案数	48964	-	-
a. 0 个单元格的期望计数小于5;最小期望计数为2204.47。			

Table 3: 性别与用字难度卡方检验

由于单名在编码长度上比双名短一位，大大增加了重名的可能。取名人为了降低重名的概率，会在选字入名时有意选择一些复杂的、不常用的字。由于女性的单名比例高于男性，为排除名字长度的影响，本文将单名和双名分离，分别研究两性人名用字的难易程度。下图2是按自然年份划分的单名用字难易程度的分布。

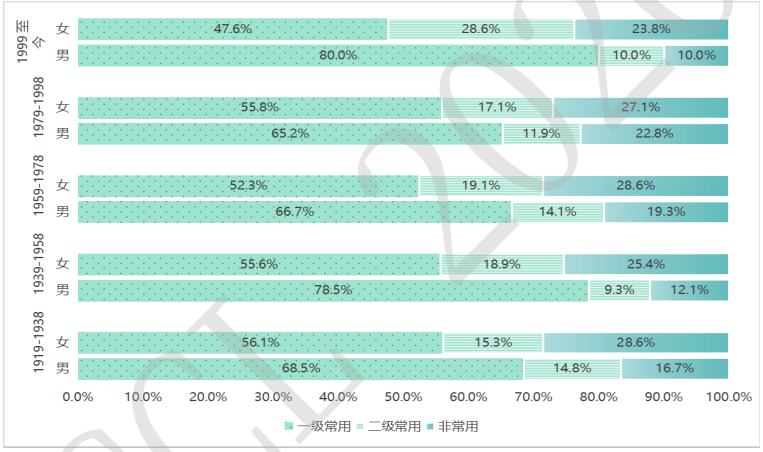


Figure 2: 自然年份中用字难易程度的性别差异（单名）

由图2可知：1）单名用字以一级常用字为主，其次是非常用字，最后才是二级常用字。二级常用字比较稳定，非常用字最不稳定；2）女性单名用字较之男性更复杂，但是两性人名的非常用字总体呈现出下降的趋势。男性用字难易度波幅较大，较女性更加不稳定；3）对比图3后发现，人名的非常用字出现过两次低谷阶段，第一次低谷时期是1939-1958年附近，第二次则是1999年至今。在1966-1979年附近，非常用字出现过一次激增阶段。

人名主要的功能是区分彼此并且满足社会成员之间的互动，所以名字要便于辨认，因此常用字总体占比会更多。但同时人名也是个人身份的标签，为了体现个性或者表达某些特殊含意，也会出现特殊的字，因此非常用字也会占据一定比例。而正是出于这些特殊的情感表达，会在某些特定时间段内出现较大变化，于是就会出现上图2,3中所看到的较大起伏。值得注意的是，男性人名的非常用字波幅大于女性，男性非常用字的激增阶段对应的是1966-1978这一时期，其名字更容易与特定事件和时段挂钩。人名非常用字的两次低谷对应的原因可能有所不同。第一次非常用字的低谷伴随的是一级常用字的增长，而第二次则伴随的是二级常用字的增长。我们的猜想是第一次低谷相应时期的新生儿父母多经历了多年战争，受教育机会少，文化程度低，在取名的时候受到文化程度等因素的限制，用字相对其他时期更加简单。而第二次低谷则更可能是新时期语言文字工作者和有关政府对于姓名规范的呼吁。

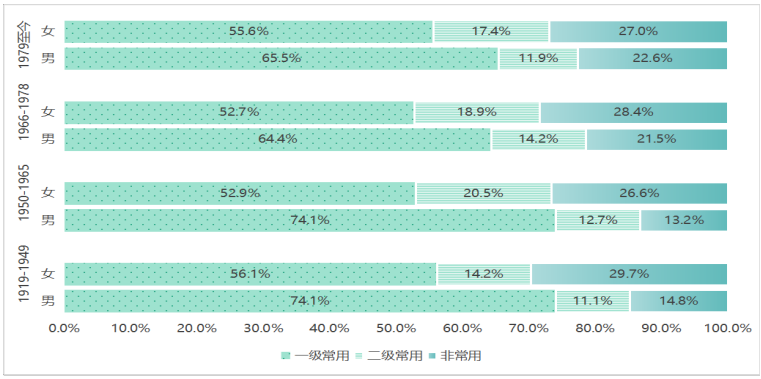


Figure 3: 重大历史事件中用字难易程度的性别差异(单名)

图4是按自然年份划分的双名用字难易程度的分布。

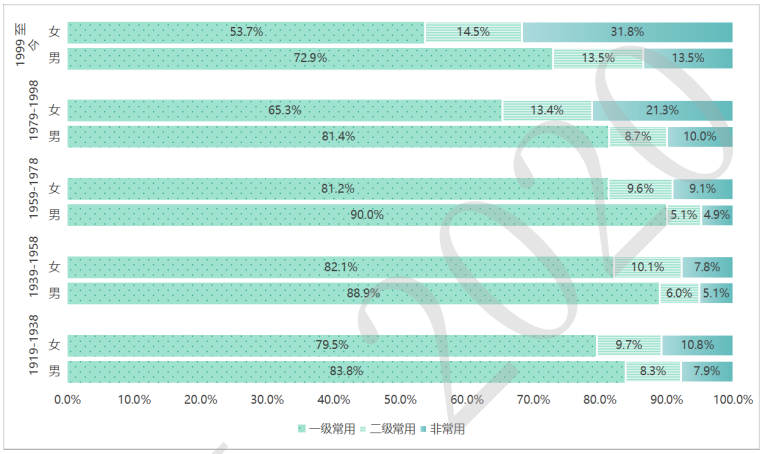


Figure 4: 自然年份中用字难易程度的性别差异（双名）

由图4可以知:

1）与单名类似，双名的一级常用字占据绝对优势，二级常用字最稳定；2）女性双名的整体用字仍然难于男性，两性人名用字在1978年后都发生了显著变化，一级常用字减少，非常用字呈现上升趋势；3）改革开放后人名用字与前三个阶段有明显区别，而这期间又分为两个时期，21世纪后的两性用字的难易程度差异变大，女性非常用字增幅比男性更显著。

虽然双名难易度的三个等级分布于单名类似，但是整体上双名用字比单名简单，这印证了本节开头的假设即用字的难易度与名字难易度有一定关联。与单名逐渐变得更简单所不同的是，双名逐渐变得复杂，这与二者的基数有关。

4.3 人名用字丰富度

在计量语言学中测量词汇丰富度常用的指标是型例比（TTR，即type-token ratio），该指标计算的是文本中不同的词语在所有词语中所占的比例，TTR值越大,说明文本使用的词汇越丰富。但是TTR的值受到语料库大小的影响，因此我们在本文中使用的“吉罗指数”(Teich, 2012; Yu, 2010)，它是TTR的变体，可减少文本大小对于丰富度的影响，其公式为:

$$Index \quad of \quad Guiraud = Types/\sqrt{Tokens}$$

其中，types是型符，tokens是类符。我们将词汇丰富度这一指标应用到人名用字中，计算人名用字的丰富度，计算得到历时百年的人名用字总体丰富度为11.45。以施建刚and 邵斌 (2016)计算的“兰卡斯特现代汉语语料库”传记和散文子库中的吉罗指数参照，该语料库的吉罗指数为43.93，远高于人名用字的吉罗指数，这说明相对于普通汉语书面语，人名用字的丰富度较低，用字比较集中。

为探究不同年代用字丰富度的变化以及两性人名用字丰富度的差异，我们分别按照自然年代的划分计算了男性和女性人名用字的吉罗指数：

	1919-1938	1939-1958	1959-1978	1979-1998	1999至今
男	15.03	12.98	11.81	14.58	10.34
女	14.95	12.82	11.73	13.60	12.00

Table 4: 两性人名用字丰富度的吉罗指数

从表4中可以看到，从1919年至今，人名用字的丰富度总体呈现出降低的趋势，也就是人名用字逐渐单一化，进入人名的汉字越来越集中。人名作为专有名词的一种，其专有性和独特性越来越突出。同时对比两性人名用字的吉罗指数发现，1998年以前，男性人名用字比女性人名用字丰富。

4.4 百年人名高频字及用字性别偏度

人名与所处的时代、相关政治历时事件、社会经济等有密切关系，不同年代的人取名有一定的特点。我们对近百年来两性人名中出现的汉字进行统计，取高频字的前15个字。两性人名用字的总体差异，见表5。在这些高频字中只有“华”字是重叠的，其在现代汉语词典 (2016)中的解释主要有“中国、繁华、精英、美丽”等，这些代表了中国人名最常包含的意义。

男高频字	明、文、国、华、建、志、军、伟、德、平、海、林、东、成、永
女高频字	丽、英、红、晓、华、芳、玲、玉、兰、梅、小、文、秀、萍、慧

Table 5: 百年来两性人名高频字

在我们感性认知中，男女人名在用字上应该有所不同，有些字在男性名字中常用，有些则在女性名字中常用。但是哪些汉字具有明显的倾向，以及汉字本身具有怎样的特征呢？本文设计了人名用字的性别偏度这一指标，从定量上对人名与性别的关联进行评估考察，其公式为：

$$V = \frac{P_{male} - P_{female}}{P_{total}}$$

对于V大于0的汉字我们认为其偏向于男性用字，小于0的汉字我们认为其偏向于女性用字，最终得到偏男汉字652个，偏女汉字432个。当V=0时，代表该汉字在男女中的分布均衡，我们称之为中性字。当V=±1.00时，则代表该汉字仅出现在一种性别之中，我们称之为极男/女字。取总字频为前1000的所有汉字中的性别偏度，最终得出百年人名性别极性字表(表5)。

极男（29）	栋、彪、乾、腾、庚、敦、涌、朋、营、干、创、钊、挺、甲、关、典、财、录、仰、猛、巨、罡、谋、专、炯、熠、纲、炯、熠
极女（61）	琴、婉、蕾、娣、妹、茜、婧、瑛、娅、筠、妙、莺、婕、苹、媚、蕙、妤、姿、姗、姝、荷、婵、女、黛、玛、翎、嫣、苑、涓、妃、甜、蔓、笛、珈、鸽、菱、璧、颐、舞、瑰、函、姐、鹃、纳、蜜、蜀、箐、霭、瑗、玟、拉、涟、漪、欧、嫦、绣、飘、俏、菡、蘅、薰
中性（28）	乐、庭、桐、淼、又、祯、阿、李、陆、汶、地、季、闻、贻、翼、铮、尘、薪、至、层、呈、古、隽、临、珑、施、晏、尹

Table 6: 百年高频人名用字性别极性字表

可以发现在高频字中，极女字较极男字更多，也就是在取名系统中女性人名的专用字较多。我们对表5中比较集中的几个意象作了简要归纳，得到表6：

类别	字义/意象	字例
偏男	具有较强动作性 凶猛、巨大等意	腾、涌、营、干、创、挺、仰 彪、猛、巨
偏女	描写女性姿态气质 女性称谓 含有娇小、美丽等意象	婉、媚、姿 娣、妹、姐 蕾、莺、茜、婧
中性	姓氏用字	李、陆、季、古、施、尹

Table 7: 两性用字意象归纳

4.5 人名用字历时变化趋势

我们分性别将所有年份的用字进行统计，得出不同年份的高频字表，表8、9:

	男高频字	女高频字
1919-1938	德、华、良、文、明、振、民、元 成、家、国、昌、一、兴、荣	英、华、兰、玉、珍、芳、芬、淑 梅、琴、文、丽、凤、秀、桂
1939-1958	国、明、文、德、生、林、华、光 建、正、平、学、新、海、祥	英、兰、华、玉、玲、淑、芳、丽 秀、小、美、萍、敏、凤、珍
1959-1978	明、文、军、国、华、建、平、志 东、永、伟、海、林、春、新	红、丽、英、晓、梅、玲、华、萍 芳、秀、霞、玉、慧、燕、兰
1978-1998	龙、俊、伟、文、子、鹏、明、杰 晓、东、小、宇、志、海、天	佳、婷、晓、丽、子、雨、文、娜 小、思、琳、雅、嘉、梦、一
1999至今	嘉、俊、宇、泽、子、博、涵、浩 杰、天、轩、艺、柏、晨、海	佳、子、儿、思、怡、雅、一、涵 琪、倩、诗、彤、雯、馨、钰

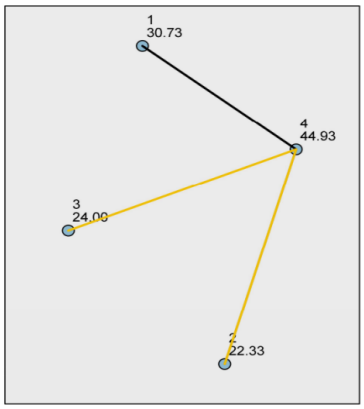
Table 8: 自然年份中两性人名高频字分布

	男高频字	女高频字
1919-1949	德、文、华、国、明、昌、成、林 良、正、家、元、生、民、学	英、华、兰、玉、芳、淑、珍、丽 秀、梅、芬、凤、桂、美、文
1950-1965	明、国、建、华、文、平、林、德 生、志、光、军、新、荣、伟	英、丽、玲、华、萍、兰、秀、晓 芳、梅、小、红、玉、霞、凤
1966-1978	文、军、明、国、东、华、志、海 建、伟、平、峰、春、永、成	红、丽、晓、英、梅、玲、芳、慧 霞、华、玉、艳、燕、小、秀
1979至今	俊、龙、伟、文、子、鹏、明、杰 宇、晓、小、东、海、天、志	佳、婷、子、晓、丽、雨、文、思 娜、小、雅、琳、一、嘉、怡

Table 9: 重大历史事件中两性人名高频字分布

按照总体高频字表中每个字对应的排序，对历时层面的前15个高频字进行编码。前人的研究认为，人名用字与历史事件有关，因此在划分人名阶段的时候采用了重大历史事件作为节点。所以我们首先选择表9的内容，采用独立样本Kruskal-Wallis检验。检验结果表明男性用字的分布在不同阶段上不具有统计学意义上的差别，而女性用字具有统计学意义上的差异。具体统计检验结果见图5。

因为女性用字分布存在差异，所以进一步对女性用字做了成对比较，探究不同阶段之间的差异。如图6所示，图中不同时段用节点表示（每个节点显示样本的平均秩），彼此之前的关系



样本1-样本2	检验统计	标准错误	标准 检验统计	显著性	调整显著性
2-3	-1.667	6.373	-.262	.794	1.000
2-1	8.400	6.373	1.318	.187	1.000
2-4	-22.600	6.373	-3.546	.000	.002
3-1	6.733	6.373	1.057	.291	1.000
3-4	-20.933	6.373	-3.285	.001	.006
1-4	-14.200	6.373	-2.228	.026	.155

每行会检验零假设：样本 1 和样本 2 分布相同。
显示渐进显著性（双侧检验）。显著性水平为 .05。
Bonferroni 校正已针对多个检验调整显著性值。

Figure 6: 女性用字分布与年代的成对比较

用实线连接，其中黄色的实线代表具有显著差异性。可以看到，女性用字差异主要来源于第四个阶段，即改革开放以后。这说明女性用字在改革开放后发生了显著变化。

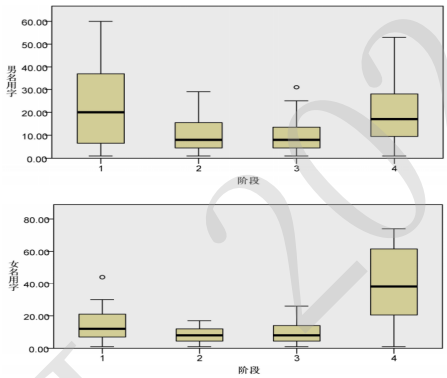


Figure 5: 人名用字的Kruskal-Wallis检验，检验总计量为60。男性人名均值6.704，双侧渐进显著性为0.082，女性人名均值15.627，双侧渐进显著性为0.001。

鉴于差异主要分为两个大的阶段，即1979年前后（改革开放前后），因此我们对这一时期前后两性人名高频字变化的主要特点做出归纳，见表10:

性别	时期	字义/意象	例字
男	1979年以前	品德与志向；	德、正、伟、荣、志；
		代表时政；	国、军、红、建；
	1979年以后	表外貌；	俊；
		取动物中宏大高远的有关意象；	鹏、龙；
女	1979年以前	品德与美貌；	丽、美、淑；
		取植物、美玉中美好的有关意象；	兰、玉、梅；
	1979年以后	译名常用字；	娜、琳、菲；
		表梦幻、优雅等意象。	雨、诗、怡、雅；

Table 10: 改革开放前后高频字主要变化

从表中可以看到，1979年以前男性人名部分与时政有关联，两性人名对于政治历史事件的敏感度是不一样的，相比之下男性人名与政治历史事件联系更加紧密，从不同阶段的男性高频字可以看出时代特点。在新中国成立前强调传统“修身治国平天下”的理想，男性高频用字跟多

反映个人品德修养。表8中可以看到在第一阶段“德”“良”等字都在前15个高频字中。在第二阶段男性人名所反映出的伟大志向、建设祖国的心愿，如“建”“志”等。第三阶段则反映出革命与建设的潮流，如“军”“伟”。男性人名用字分布虽然总体没有显著差异，但每个阶段都随着时代特征而体现出不同的侧重，高频字的排序随之发生变化。

不过变化是缓慢且滞后的，可以通过两种时间划分方式的重叠部分来推测不同时间段内部的变化，以“德”“建”二字为例，从“德”字在表8,9中的排序变化可以看出该字在新中国成立前使用非常频繁，新中国成立后起使用频率并没有迅速降低，甚至在早期使用依然较多，随着时间的发展逐渐变得不那么常用。类似地，1950-1958年段中“建”的使用不如1959-1965年多。与其他历时阶段比，“建”字在建国初期呈现小高峰，反映了特定的时代特征。而在这一阶段内部再比较，该字的使用又呈现出了逐渐上升的趋势。因此人名的变化虽然受政治事件的影响但是其变化是具有渐进性和滞后性。

为了更清晰地显示人名高频用字的变化，我们分性别总结了各个阶段排名均在50的字，取前10个，归稳定且常用；取第四个阶段在前50，并且呈现上升趋势的前20个字归为上升快且当前常用(Max(Rank1,Rank2,Rank3)-Rank4)；取前三个阶段的任意阶段曾在前50，但在第四个阶段下降的前20个字归为下降快且曾经常(Rank4-Min(Rank1,Rank2,Rank3))。最终得到表11。

	男	女
稳定且常用	成、德、国、海、华、林、明、平文、祥、永、志	芳、慧、娟、君、丽、玲、美、敏文、小、晓、雪、燕、玉
上升快且当前常用	磊、博、锋、洋、豪、佳、君、军超、嘉、波、勇、宇、强、飞、小阳、涛、峰、江	儿、涵、雨、婷、菲、诗、妍、艺萱、宇、梦、妮、倩、心、欣、怡嘉、丹、颖、佳
下降快且曾经常用	武、贵、孝、全、万、山、学、民克、刚、昌、绍、仁、荣、宗、胜景、鸿、良、世	生、荣、桂、素、菊、建、德、莲平、芬、珍、瑞、淑、芝、琼、秀世、利、珠、爱

Table 11: 用字变化趋势表

5 结语

本文构建了一个中国名人人名数据库，条目共11万+，每个条目含有人名、性别、出生地等社会文化标签，同时含有拼音、笔画、偏旁等文字信息标签。该语料库可以支持对人名的地域、历时、性别等多个维度的研究。

在人名数据库的支持下，本文选取1919至今的人名作为研究对象，从人名长度、用字难易度、丰富度等角度进行探究。研究发现男性人名比女性长，但两性人名长度的差异随着时间而不断缩小。建国以来单名比例不断增加，但是进入21世纪又逐渐减小。单名用字比双名难，女性人名用字比男性难，男性用字难易程度波动较大。人名中的二级常用字最为稳定，其次是一级常用字。在用字的丰富度上，随着时间的发展人名用字越来越体现出其专有性的特征，丰富度逐渐降低。男性人名的用字总体上比女性用字丰富。通过计算人名的性别偏度指标后发现女性人名专用字更多。改革开放对人名用字格局产生了重要影响，女性用字的变化显著。男性人名与时政联系更加紧密，用字的变化虽然受时政的影响，但其变化具有渐进性和滞后性。

这些发现可以帮助我们进一步了解人名的发展变化规律，探究汉字中的的性别差异。当然本文还存在一些缺陷，例如在自然年份中1919年至今这一时段的人名较少，对数据的分析产生一定影响。下一步，我们将持续补充新增名人人名数据及相关信息，并从偏旁、地域等维度进行深入研究。

致谢

感谢论文辅导老师对本论文的帮助，感谢各位匿名评审老师的修改建议。本论文受教育部人文社会科学研究规划基金资助项目（18YJA740030）和北京语言大学研究生创新基金项目（20YCX155）资助。

参考文献

Jizheng Jia and Qiyang Zhao. 2019. Gender prediction based on chinese name. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 676–683. Springer.

Elke Teich. 2012. *Cross-linguistic variation in system and text: A methodology for the investigation of translations and comparable texts*, volume 5. Walter de Gruyter.

Guoxing Yu. 2010. Lexical diversity in writing and speaking task performances. *Applied linguistics*, 31(2):236–259.

何晓明. 2001. 姓名与中国文化/中国文化新论丛书. 人民出版社.

吴继章. 2001. 也谈人名中的异体字. 语文建设, 8.

张书岩. 1999. 从人名看50 年的变迁. 语文建设, 4.

张书岩. 2004. 姓名· 汉字· 规范. 北京广播学院出版社.

施建刚and 邵斌. 2016. 基于语料库的汉语译文翻译共性研究——以《苏东坡传》汉译本为例. 外国语言文学, 33(2):97r104.

潘世松. 2004. 汉字结构的性别歧视倾向论析. 求索, 12:212–214.

王玉新. 2000. 汉字认知究. 山东大学出版社.

现代汉语词典. 2016. 第七版. 中国社会科学院语言研究所词典编辑室编. 北京: 商务印书馆.

苏培成. 2001. 谈人名中的异体字. 语文建设, 5.

谢玉娥. 2000. 人名、性别、文化——对男人名,女人名文化现象的考察. 中国文化研究, (1):103–108.

赵越. 2006. 汉人韵律情结, 命名文化与aba (a') 命名方式. 语文学刊, (13):26.

邱莉芹and 鞠泓. 2002. 人名用字中使用生僻字情况的调查与分析. Ph.D. thesis.

韩燕, 邱江, and 张庆林. 2008. 性别刻板化人名推测判断中的冲突效应. Ph.D. thesis.

伟大的男人和倔强的女人：基于语料库的形容词性别偏度历时研究

朱述承　　刘鹏远*
北京语言大学　信息科学学院
国家语言资源监测与研究平面媒体中心
北京市海淀区学院路15号, 100083
zhu_shucheng@126.com　liupengyuan@pku.edu.cn

摘要

性别偏见现象是社会语言学和计算语言学学者均关注的研究热点，但目前大多数研究都是基于英语的，鲜有对汉语中性别偏见现象，特别是基于形容词的研究——而形容词是衡量社会对男性和女性角色规约的有力抓手。本文首先利用调查问卷的方法，构建了一个含有466个形容词的数据集，定义性别偏度为特定形容词词义和男性或女性群体相匹配的程度，并计算了数据集中每个形容词的性别偏度。然后基于DCC语料库，研究了《人民日报》的形容词性别偏度的历时总体变化，并考察了和姓名搭配的形容词的历时变化。发现《人民日报》所使用的形容词随时间的推移整体呈现中性化趋势，但在"文化大革命"期间呈现非常男性化的特征，和男性姓名搭配的形容词整体呈现中性化趋势。

关键词： 语料库；形容词；性别偏度；历时研究

Great Males and Stubborn Females: A Diachronic Study of Corpus-Based Gendered Skewness in Chinese Adjectives

ZHU Shucheng　　LIU Pengyuan*
School of Information Science,
Language Resources Monitoring and Reserch Center Print Media Language Branch,
Beijing Language and Culture University,
15th Xueyuan Road, Haidian District, Beijing, 100083, China
zhu_shucheng@126.com　liupengyuan@pku.edu.cn

Abstract

Gender bias is a hot topic in both sociolinguistics and computational linguistics. However, most of the studies are based on English, and there are few studies on gender bias in Chinese language, especially on adjective, which is a powerful tool to measure social conventions on male and female roles. This article firstly used a questionnaire to construct a data set containing 466 adjectives, and calculated the gendered skewness of each adjective. Here the definition of "Gendered Skewness" is the matching degree between the semantics of particular adjectives and males or females. Then, based on the DCC corpus, the diachronic change of the gendered skewness of the adjectives in "People's Daily" was studied. The diachronic change of adjectives collocated with the names was investigated as well. It is found that the adjectives used in "People's Daily" show a trend of neutralization over time. However, during the "Cultural Revolution", it shows a very masculine skewness, and the adjectives collocated with male names show a trend of neutralization over time.

Keywords: Corpus , Adjective , Gendered skewness , Diachronic study

* 通讯作者　Corresponding Author

1 引言

《诗经·正月》有云：“赫赫宗周，褒姒灭之。”古往今来，女性作为祸国殃民的红颜祸水的说法从未停歇。人们对女性的正面形容要么是说其外貌美丽漂亮，要么是形容其性格品行端庄贤淑，无一不透露着男权社会对女性外表和内在的规约与束缚。而对女性的负面形容则强调了女性之“祸”——妖艳魅惑。因此，从形容词中，我们不但可以知道人或事物的性质、状态和属性，还可以一窥我们所处的社会对人的“定义”，当然，这种“定义”是随时间变化的。

在语法上，汉语的形容词与西班牙语、俄语等屈折语不同，并没有阴性、阳性或中性的语法范畴形式。但在词汇语义上，和所有语言一样，我们赋予了汉语形容词丰富的语义，而形容词的语义又可以分为概念义和附属义。概念义，指的是形容词一般属性或本质属性在人的意识上的概括反映；而附属义指的是人们附加在概念义以外的特定感受 (黄伯荣和廖序东, 2017)。这种特定感受往往偏向一个特定的概念和群体。例如，当我们使用“帅气”、“阳刚”、“威武”等形容词时，我们总是会觉得这些形容词更偏向于一个男性形象；而当我们使用“贤惠”、“漂亮”、“妩媚”等形容词时，我们则总会觉得这些形容词在形容一个女性形象。因此，我们定义形容词的性别偏度（gendered skewness）为特定形容词词义和男性或女性群体相匹配的程度。

形容词的性别偏度是社会对不同性别角色的构建在语言上的内在体现。我们希望形容词的性别偏度是中立的、无偏的，但是基于千百年来男权社会对语言使用的浸染，我们在语言使用过程中会呈现出一种无意识的性别偏见（unconscious gender bias）。随着计算机技术的发展，特别是深度学习和神经网络等算法的出现，计算机往往会放大这种偏见，而造成在下游应用任务的结果中出现一定程度上的性别偏见和刻板印象 (Sun et al., 2019)。在现有的研究中，特别是自然语言处理中，性别偏见现象已经成为学者关注的一大热点，但目前大多数研究都是基于英语的，对于汉语中的性别偏见现象，特别是针对形容词的研究更是接近于一片空白。

本文综合了现在的各种资源，并且利用调查问卷，构建了一个含有466个形容词及其性别偏度、情感信息的数据集AGSS[0]（Adjectives list with Gendered Skewness and Sentiment），不仅仅填补了汉语自然语言处理性别偏见研究中数据集资源缺乏的空白，并且可以应用到未来汉语性别消偏算法的应用中。在构建数据集后，本文利用DCC语料库中的历时《人民日报》语料，采用语料库的方法，结合形容词数据集，考察了不同时代的平面媒体中使用的形容词所反映出的性别偏度，以及和不同性别姓名搭配的形容词的差异。本文的研究问题主要有以下四个：

（1）不同性别和年龄人群的形容词性别偏度是怎样的？是否具有差异？具有怎样的差异？

（2）具有明显男性和女性性别偏度的形容词有哪些？和情感是否有关系？

（3）平面媒体中所使用的形容词体现出怎样的性别偏度历时变化？这是否和政治事件、社会经济发展有所联系？

（4）形容男性姓名和女性姓名的形容词随时间有怎样的性别偏度变化？

2 相关工作

2.1 社会中的性别角色

语言是社会交际的产物，不仅表达和反映了人们的思想，而且塑造了人们的世界观 (Sapir, 1929; Whorf and Carroll, 1956)。性别角色是社会构建出来的，而语言则是社会的无形触手，如同女娲造人一般将个体塑造和区分成男性和女性。因此，生活在社会中的每个个体才会对男性特质和女性特质有一个潜在的感知与印象。这一点在婚恋交友中体现的最为淋漓尽致。

利用婚恋交友网站和平台，我们可以管中窥豹我国当今社会对不同性别配偶的要求，从而反映出社会对男性和女性特质的形容和约束。在自我形容上，男性多形容自己为真诚幽默，而女性则多形容自己温柔大方；在对另一半的要求上，女性更看重男性的身高和年龄且对学历没有过多要求，希望男性可以有一番自己的事业，而男性更看重女性的样貌和身材，希望女性能照顾家庭；而随着时间的推移，男性对女性“贞洁”的要求在逐渐减少，女性从对男性的经济能力有所要求到强调自己的独立性 (金灿灿和邹泓, 2009; 靖元, 2007; 聂晶和郭明珠, 2010; 孟秀文, 2008)。由此我们可以看出，我国社会将女性塑造为一个外表相对重要的持家形象，而将男性

[0] https://github.com/NLPBLCU/Adjectives-list-with-Gendered-Skewness-and-Sentiment

塑造为一个支撑家庭的主人形象，但这种塑造不是一成不变的，是随着社会发展和时间而变化的。社会对不同性别角色的塑造不单单体现在婚恋交友上，更是将性别“密码”嵌入在语言中。

2.2 语言中的性别信息

从语法层面来看，既有像芬兰语和土耳其语这样在名词和代词上没有性标记的语言，也有像英语和瑞典语这样大部分名词没有性标记，人称代词是主要表达性的语言，更有像俄语、西班牙语和意大利语这样所有名词、形容词都有性标记，而与其搭配的其他词也有与之对应的性标记的语言 (Michela and Monica, 2017)，虽然语法上的性标记只是语法范畴，不表达性别的语义，但是计算机很可能学习了这种标记而造成性别偏见 (Gonen et al., 2019)。而在我们不常察觉的层面，例如数字中，人们更容易将精确的数字与男性联系起来，而将概数和女性联系起来 (杨晨和陈增祥, 2019)。

而随着计算机技术的不断发展，特别是深度学习和神经网络等一系列算法的出现，人们发现本应毫无偏见的计算机也“习得”了人类社会中的各种偏见，特别是性别偏见。而这主要是因为计算机放大了语言中暗藏的性别信息。正如上文提到的，像意大利语这样名词和形容词有性标记的语言，会使计算机训练出的词向量含有性别信息，从而造成结果中含有性别偏见 (Gonen et al., 2019)。而对于英语这样性标记不发达的语言，在职业词等词汇的词向量中也发现存在着性别偏见 (Tan and Celis, 2019; Bolukbasi et al., 2016)，且词向量中的性别偏见是很难消除的 (Gonen and Goldberg, 2019)，词向量也可以反映社会中性别刻板印象的历时变化 (Garg et al., 2018; Wevers, 2019)。在关系抽取中，不同的关系抽取出的人物性别不同，如与结婚有关的关系抽取出的女性更多 (Gaut et al., 2020)。造成这些现象的原因在于原始语料中的性别偏见。例如，在政治语料库中女性总是与儿童共现 (Karimullah, 2020)，维基百科中女性的人物传记多与婚姻和家庭事务有关 (Graells-Garrido et al., 2015; Wagner et al., 2015)。而在视觉领域，不同性别用户发布的图片内容不同 (Alvarez-Carmona et al., 2018)，导致视觉语义标注中也存在性别偏见，如在厨房中的人物总是被识别为女性 (Zhao et al., 2017)。

2.3 形容词与性别

如果进一步聚焦到形容词与性别的关系上，我们会发现形容词是衡量社会对男性和女性角色规约的一个有力抓手。在国外的研究中，学者们通过不同的手段对形容不同性别人群的形容词进行了刻画，如在美式英语和英式英语中，分别对与男孩（boy）和与女孩（girl）搭配的形容词从社交、外貌和品行等不同角度进行考察 (刘旭阳, 2017; Baker, 2010)，而形容男性和女性的形容词与文体也有很大关联 (Fast et al., 2016)，美剧中男性人物和女性人物使用的形容词存在差异 (张进, 2010)，印度宝莱坞电影中对男性角色和女性角色的刻画也存在性别差异 (Madaan et al., 2018)，且随着时间的推移和重大事件的影响，人们会选择不同的形容词形容男性和女性 (Garg et al., 2018)。

汉语中考察形容词与性别关系的研究较少。目前已有的研究包括对教科书中不同性别人物刻画的形容词的定量分析 (陈莉娜, 2005),男性和女性在日常生活中使用的形容词的差异 (王悦和齐畅, 2011)。此外，还包括：利用心理学词表建立中国人的男性化和女性化特征 (崔红和王登峰, 2005)；通过语料库统计与实验设计的方法，考察汉语二语学习者在汉语形容词习得过程中性别知识的习得及发展过程 (付超, 2018)。由此可见，对中文形容词的性别研究缺少一个全面且深入的刻画。

3 数据集AGSS

3.1 数据集构建

为了尽可能全面的刻画汉语中形容词的性别的情况，我们首先选取了《现代汉语词典（第五版）》 (中国社会科学院语言研究所词典编纂室, 2005)释义中含有“形容人”的形容词，共计100个。然后又选取了《常用形容词分类词典（第三版）》 (傅玉芳, 2010)中“性格品行”、“智慧才能”、“心情感觉”、“目光神情”、“言行举止”、“行为态度”、“容貌体态”、“名声荣誉”、“感情友谊”、“为人处事”、“健康人生”、“处境遭遇”、“贫富俭奢”、“劳动技艺”、“声响音调”、“正常普通”这16个部分共2045个形容词。综合这两个词表合并取交集，共得到形容词1986个。为了进一步观察这些形容词是否能形容人，我们邀请了6位语言学专业的标注人员，分别对这些形容词进行标注，保留了6位标注者均认为的可以形容人（强调人的固有属性而非动

作等）的形容词，共计556个。最后与大连理工大学情感词汇本体库 (陈建美, 2008)中的形容词取交集，共得到466个可以形容人的，且含有情感信息的形容词。

为了得到不同人群的形容词性别偏度，我们使用了问卷调查的研究方法。使用“问卷星”发放了113份网络调查问卷，同时发放了2份纸质版问卷，共发放了115份调查问卷。所有的受调查者都是自愿参与问卷的填写，并且都可以获得10元人民币的酬劳。调查问卷为五级的李克特量表，要求被调查者对466个形容词（如“伟大”、“倔强”、“狡猾”、“善良”）进行评分，1分为该形容词几乎只形容女性，2分为该形容词形容女性的稍多，3分为该形容词形容男性和女性的程度一样，4分为该形容词形容男性的稍多，5分为该形容词几乎只形容男性。同时统计了被调查者的人口学信息，包括被调查者的性别和年龄。在调查问卷中同时设计了两道测试选择题，用以排除被调查者不能认真填写调查问卷的情况，同时排除了作答时间在600秒以下的调查问卷，共得到有效的调查问卷共计108份，有效率为93.91%。回收的有效调查问卷的人口学信息见Figure 1，其中男性48人，女性60人；20岁以下15人，21-30岁41人，31-40岁9人，41-50岁14人，51-60岁24人，61岁以上5人。108份有效调查问卷的克隆巴赫系数为0.949，属于高信度的调查问卷。整体的平均值为3.11，标准差为0.87，说明所选形容词在人们整体认知中稍稍偏向男性。最终我们得到了一个带有性别偏度的形容词词表数据集。

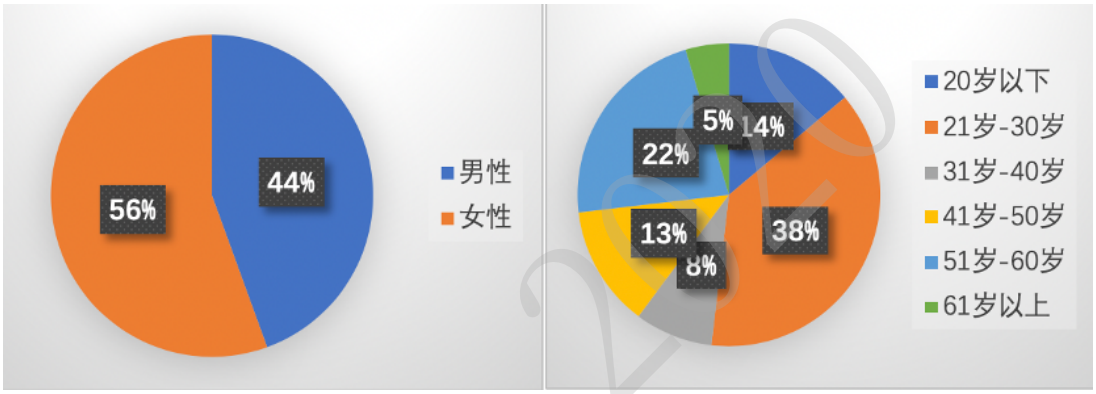


Figure 1: 有效调查问卷的人口学信息

3.2 数据集分析

分别计算出男性群体和女性群体、30岁以下群体和30岁以上群体（以30岁为界是因为人数较为接近）对每个形容词的评分均值，作为该群体的形容词性别偏度GS。该值为一个1至5区间的数值，3为该群体对该形容词的性别偏度为中性，越接近1说明该群体认为该形容词越偏向女性，越接近5说明该群体认为该形容词越偏向男性。然后对男性群体和女性群体，30岁以下群体和30岁以上群体的形容词性别偏度分别进行独立样本t检验，用以刻画不同群体的差异，从而回答研究问题（1）。

	莱文方差等同性检验		平均值等同性t 检验			
	F	显著性	t	自由度	Sig.（双尾）	平均值差值
假定等方差	4.236	0.042	1.122	106	0.264	0.02735
不假定等方差			1.076	78.945	0.285	0.02735

Table 1: 男性群体和女性群体的形容词性别偏度独立样本t检验

男性群体的形容词性别偏度为3.12，标准差为0.87；女性群体的形容词性别偏度为3.11，标准差为0.87。如Table 1所示，男性群体和女性群体对所选择的466个形容词的性别偏度没有统计学意义上的显著性差异。

30岁以下群体的形容词性别偏度为3.08，标准差为0.79；30岁以上群体的形容词性别偏度为3.14，标准差为0.94。如Table 2所示，30岁以下群体和30岁以上群体对所选择的466个形容词的性别偏度存在统计学意义上的显著性差异，即30岁以上群体比30岁以下群体认为这些形容词

	莱文方差等同性检验		平均值等同性t 检验			
	F	显著性	t	自由度	Sig.（双尾）	平均值差值
假定等方差	4.017	0.048	-2.363	106	0.020	-0.05616
不假定等方差			-2.336	92.177	0.022	-0.05616

Table 2: 30岁以下群体和30岁以上群体的形容词性别偏度独立样本t检验

偏向男性的程度更深。由此我们可以认为年龄与时代是影响人们认知形容词性别偏度的一个重要因素。随着社会越来越开放和包容，年轻一代人的形容词性别偏度更加趋向于中性化。

3.3 情感分析

之后，我们计算了调查问卷每个形容词的评分均值，作为该形容词的性别偏度GS，并和大连理工大学情感词汇本体库中相同的形容词的情感极性、情感强度等值进行对比分析，用以回答研究问题（2）。

在情感极性方面，466个形容词中有62个中性词，性别偏度为3.01，标准差为0.42；有244个褒义词，性别偏度为3.07，标准差为0.72；有157个贬义词，性别偏度为3.22，标准差为0.46；有3个兼有褒贬义词，性别偏度为3.16，标准差为0.70。为进一步研究形容词的情感极性和性别偏度之间的关系，对褒义词和贬义词的性别偏度进行了独立样本t检验，如Table 3所示。

	莱文方差等同性检验		平均值等同性t 检验			
	F	显著性	t	自由度	Sig.（双尾）	平均值差值
假定等方差	12.780	0.000	-2.360	399	0.019	-0.1517812
不假定等方差			-2.580	398.978	0.010	-0.1517812

Table 3: 褒义和贬义形容词的性别偏度独立样本t检验

Table 3表明褒义形容词和贬义形容词的性别偏度在统计学上存在显著性差异，即：贬义形容词侧重于形容男性的程度更深，且大众对其认知较为一致；而褒义形容词侧重于形容男性的程度较浅，更侧重于形容中性或女性，且大众对其认知的差异较大。这体现了语言的“社会伪装性”，即虽然我们可能会将女性姓名和消极词汇，男性姓名和积极词汇产生内隐联想（implicit association），并通过词向量进行测试得到了确认 (Caliskan et al., 2017)，但是在正式场合语言总是倾向于对相对弱势的群体，体现在形容词上，则为近年来，女性总是被形容为“美好的” (Baker, 2010)。下面再具体来看一下形容男性和形容女性的典型褒义、贬义和中性的形容词都有哪些。

情感极性	女性（GS小于2）	男性（GS大于4）	中立（GS等于3）
中性	羞羞答答，羞答答，柔弱	高大	懒洋洋，心急
褒义	娇媚，妩媚，柔媚，贤淑，俏丽，水灵灵，娴静，贤惠，丰腴，娇羞，温婉，美丽，妖娆，丰满，端庄，苗条，漂亮，心灵手巧，颖慧，清纯，文静，乖巧，坚贞，灵巧	憨厚，刚毅，斯文，健旺，肥壮，儒雅，清俊，强健，威风，文质彬彬，勇猛，神勇，健壮，英武，帅气，健硕，刚健，精壮，威武，壮实，魁伟，勇武，英俊，雄健，壮硕，魁梧	友善，开朗，谨慎，虚心，无私，认真
贬义	泼辣，纤弱，娇贵，骄矜	窝囊，老谋深算，荒淫，凶暴，下流，流气，猥琐	糊涂

Table 4: 典型的形容男性、女性和中性的不同情感极性的形容词

由Table 4可以看出典型形容女性的褒义词大多强调女性的外貌和神态，对女性内在精神品质的形容则集中在表现女性的贤良淑德，符合社会对女性的认可和规范；而典型形容男性的褒义词则强调男性身体的强壮；典型形容男性和女性的贬义词则强调其性格特征上的缺失。在英语中，积极的形容词总是用来形容女性的身体而非男性的身体 (Hoyle et al., 2019)。人们普遍对于女性的外表更加关注，更加在乎女性给人们带来的审美上的愉悦，说明女性这一形象是被物质化的，人们对女性的评价也多为褒义，充满了对女性的欣赏与喜爱 (刘旭阳, 2017)。这些现象无不表明了女性角色长期是被社会约束的，女性总是被期待成为一个美好的形象，拥有良好的品行，而对男性则没有如此的社会期许 (Suzanne, 2014)。

		GS	情感强度
GS	皮尔逊相关性	1	.123**
	Sig.(双尾)		.008
	个案数	466	466
情感强度	皮尔逊相关性	.123**	1
	Sig.(双尾)	.008	
	个案数	466	466

**. 在0.01 级别（双尾），相关性显著。

Table 5: 形容词性别偏度（GS）和情感强度的相关性

最后我们对每一个形容词的性别偏度和其情感强度进行了相关性分析，如Table 5。结果显示皮尔逊相关系数为0.123，p值小于0.01，表明形容词的性别偏度和情感强度呈显著的正相关。这就表明：倾向于形容男性的形容词，其表达的情感也就相对强烈；而倾向于形容女性的形容词，其表达的情感也就相对柔缓。这同样符合我们对于男性角色和女性角色的认知：男性似乎总与强烈的力量挂钩，而女性则给我们一种柔和中庸的情绪。

4 语料库分析

本文所使用的语料全部来源于国家语言资源监测与研究中心平面媒体中心[1]研制开发的DCC动态流通语料库，为了较为全面的刻画形容词性别偏度的历时变化，我们选取了1946年至2018年全部的《人民日报》语料。然后，利用python的jieba包对文本进行分词、词性标注等预处理。在这里，为了提高对文本中姓名识别的准确率，我们将一个含有767845个姓名的词表[2]加入分词程序的自定义词典中。

4.1 形容词性别偏度与政治经济

在得到了466个形容词的性别偏度GS后，我们统计了这些形容词在分词并进行词性标注后的历时《人民日报》语料中的词频。排除了词频总数为0的形容词，最终剩余256个形容词。然后分别计算每一年《人民日报》语料的总体形容词性别偏度（Gendered Skewness of Each Year，GSEY），计算公式见公式(1)。

$$GSEY = \sum_{i=1}^{256} GS * \frac{N_i}{N} \tag{1}$$

其中，N为该年《人民日报》语料中256个形容词的总词频，N_i为第i个形容词在该年《人民日报》语料的总词频。得到的每年的形容词性别偏度GSEY也为一个1至5之间的数。若该年的总体性别偏度GSEY接近于3，则该年《人民日报》使用语言中的形容词偏向中性；若该年的总体性别偏度GSEY接近于5，则该年《人民日报》使用语言中的形容词偏向男性；若该年的总体性别偏度GSEY接近于1，则该年《人民日报》使用语言中的形容词偏向女性。根据1946年至2018年每年《人民日报》的总体性别偏度GSEY，我们绘制出一条反映《人民日报》所使用形容词性别偏度变化的折线Figure 2。对这条折线和数据加以分析，回答了研究问题（3）。

[1] http://cnlr.blcu.edu.cn
[2] https://github.com/wainshine/Chinese-Names-Corpus

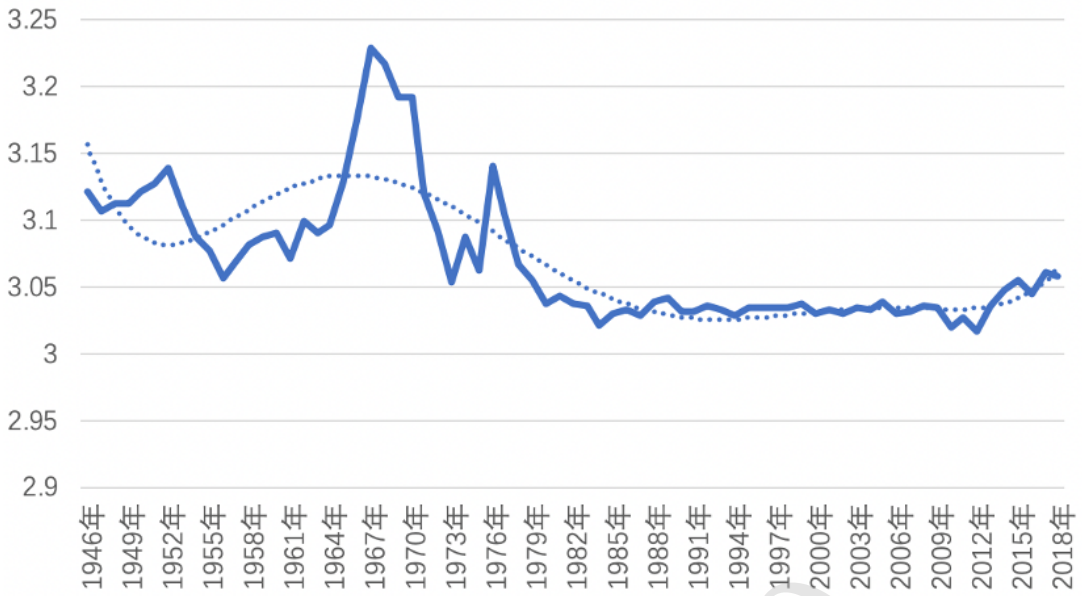


Figure 2: 《人民日报》总体性别偏度GSEY历时变化

		GSEY	GDP总量	人均GDP	人均GDP增速
GSEY	皮尔逊相关性	1	-.304*	-.313*	-.270*
	Sig.(双尾)		.014	.011	.029
	个案数	65	65	65	65
GDP总量	皮尔逊相关性	-.304*	1	1.000**	.093
	Sig.(双尾)	.014		.000	.460
	个案数	65	65	65	65
人均GDP	皮尔逊相关性	-.313*	1.000**	1	.098
	Sig.(双尾)	.011	.000		.437
	个案数	65	65	65	65
人均GDP增速	皮尔逊相关性	-.270*	.093	.098	1
	Sig.(双尾)	.029	.460	.437	
	个案数	65	65	65	65

*. 在0.05 级别（双尾），相关性显著。
**. 在0.01 级别（双尾），相关性显著。

Table 6: 每年《人民日报》总体形容词性别偏度（GSEY）和每年经济发展指标的相关性

Figure 2显示《人民日报》所使用的形容词整体上呈现偏向性别中性化的趋势，但其中没有任何一年的值等于或小于3，说明其使用的形容词还是偏向男性的。值得注意的是，在折线中，自1963年开始，至1979年止，出现了一个波动的峰值，这反映了政治活动对语言性别偏度的影响。1966年至1976年为“文化大革命”时期，从“文革”前期开始，《人民日报》就成为了“文革”的对外宣传窗口，一直受“文革”高层人员的控制，因此折线在1963年至1969年达到了第一个峰值，说明这一时期的语言偏向男性风格的意味十分浓重，例如，“伟大”这一偏向男性的形容词在这一时期达到了词频的顶峰。但在1972年，周恩来等人在《人民日报上》发表社论《争取新的胜利》以及《无政府主义是假马克思主义骗子的反革命工具》等一系列批评“左倾”的文章，《人民日报》开始出现了“反文革”的声音，因此在1972至1974年折线到了一个“鞍部”，这一时期的语言偏向男性的程度有所缓和，“伟大”等偏向男性的形容词词频大幅度下降，而偏向女性的“亲切”、“友好”等形容词的词频则有所上升。然而1976年，周恩来和毛泽东先后去世，《人民日报》再次被“文革”分子掌控，尤其在“天安门事件”中起到了极其不好的影响，1977年《人民日报》仍有社论《学好文件抓住纲》，强调“两个凡是”方针，因此这两年折线又达到了

第二个小高峰，显示语言又重新返回了偏向男性程度较重的风格，“伟大”等偏向男性的形容词词频也重回高峰，但没有达到之前的峰值。1978年，随着十一届三中全会的召开，“文化大革命”时期基本结束，折线也有所回落，趋于平缓，显示男性化的语言风格有所减弱，同时也开始出现“贤惠”、“娇媚”等一系列极偏向女性的形容词。

通过重大政治事件和折线的关系，我们认为激进的政治主义可能会导致使用带有较为强烈的男性性别偏向的语言用法倾向。学者通过研究政治语言中的隐喻，发现激进政客所使用的语言中的隐喻具有男性风格 (Kathleen, 2009)。而与战争有关的政治活动也被视为是“纯粹的”男性主义活动 (Wilson, 1992)。“文化大革命”期间语言使用的一个显性特征就是军事词语（如“胜利”、“斗争”、“敌人”）的大量使用，而粗野词（如“砸个稀巴烂”、“去死”）和政治词语（如“批斗”、“红色”）的使用在《人民日报》的历史上也达到了顶峰；此外，“批斗”、“文革体”等语言风格也在这一时期迅速发展 (刁晏斌, 2006; 周有光, 1995; 郑也夫, 1993; 邱明波, 2008)。总体来看，这些语言特征都与“文化大革命”时期的暴力和激进崇拜有关 (李逊和裴宜理, 1993)，而男性主义往往也是和力量有关的。

语言中的性别偏度除了与政治活动有关外，还与社会经济发展有着千丝万缕的关系。通过Facebook测量的性别差异衡量指标和该国的经济水平、教育、健康等指数呈现相关性 (Garcia et al., 2018)。因此我们分析了1953年至2017年的《人民日报》的总体形容词性别偏度GSEY和国家统计局[3]发布的1953年至2017年的我国GDP总量、人均GDP和人均GDP增速的相关性，如Table 6。结果发现每年的总体形容词性别偏度GSEY和GDP总量（皮尔逊相关系数为-0.304，p值为0.014）、人均GDP（皮尔逊相关系数为-0.313，p值为0.011）和人均GDP增速（皮尔逊相关系数为-0.270，p值为0.029）均呈显著的负相关。即如果使用的形容词越偏向于中性或者女性，那么该时期经济发展水平就越好，这意味着性别越平等时期的经济发展较好。

4.2 形容词性别偏度与姓名搭配

之后，我们利用python程序抽取了“形容词数据集中的形容词+（的）+姓名”这样的搭配组。计算机自动抽取后，经由人工判断每一个搭配中的姓名在原始语料中出现时是否是人名，保留为人名的搭配，并在原始语料中判断该姓名是男性姓名还是女性姓名，进行标注。将最后的搭配进行一系列计算分析，用以回答研究问题（4）。

为了观察与男性姓名搭配的形容词搭配组和与女性姓名搭配的形容词搭配组的丰富程度，我们分别计算了两者的型例比（Type-Token Ratio，TTR），计算公式见(2)，其中Type为搭配的型符数，Token为搭配的例符数。为了消除例符数规模的影响，我们随机抽选出与女性姓名搭配组例符数量相等的男性姓名搭配组，并进行计算。经计算得，女性姓名搭配组的型例比为0.98，而男性姓名搭配组的型例比为0.75，说明形容女性的形容词较形容男性姓名的形容词更加丰富。我们又绘制了不同性别姓名搭配组的词云图（均在形容词后加入“的”以便于理解），如Figure 3所示。

$$TTR = \frac{Type}{Token} \tag{2}$$

Figure 3: 男性姓名搭配组（左）与女性姓名搭配组（右）词云图

[3] http://www.stats.gov.cn

在这里我们只计算了每年与男性姓名搭配的形容词性别偏度（Gendered Skewness collocating with Males' Names，GSMN），计算公式见(3)。

$$GSMN = \sum_{j=1}^{n} GS * \frac{NN_j}{NN} \tag{3}$$

其中，该年与男性姓名的搭配组共有n组，NN为该年的搭配组中男性姓名的总数，NN_j为该年第j个搭配组形容词的总词频。得到的每年与男性姓名搭配的形容词性别偏度GSMN也为一个1至5之间的数。若该年与男性姓名搭配的形容词性别偏度接近于3，则该年形容男性的形容词接近中性；若该年与男性姓名搭配的形容词性别偏度接近于5，则该年形容男性的形容词接近男性；若该年与男性姓名搭配的形容词性别偏度接近于1，则该年形容男性的形容词接近女性。

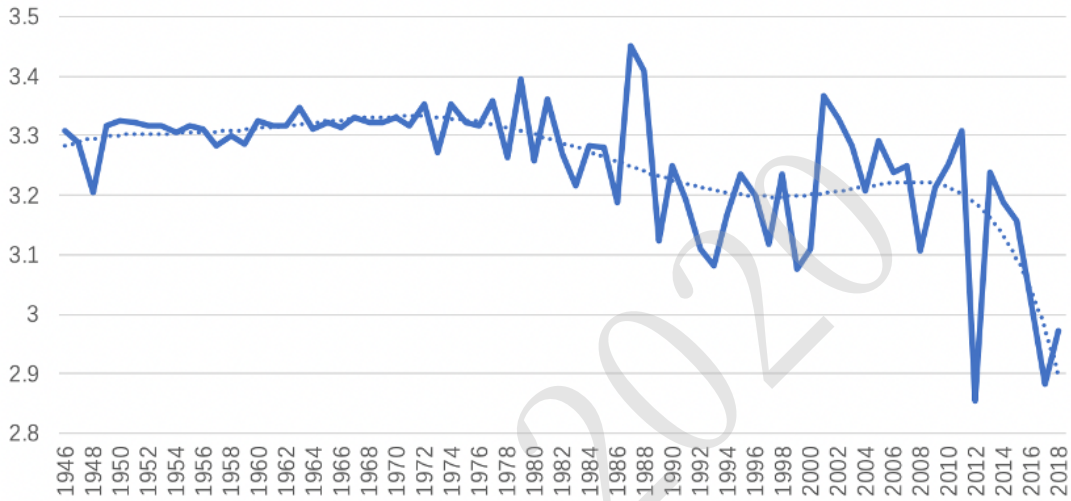


Figure 4: 与男性姓名搭配的形容词性别偏度历时变化

Figure 4反映了与男性姓名搭配，形容男性姓名的形容词的性别偏度的历时变化趋势，我们可以看出整体上形容男性姓名的形容词从过去具有典型男性偏向的形容词到现在偏向中性甚至女性的形容词。这表现了随着时间的推移，人们对于男性的社会角色形象的认知也有所转变，男性不一定要符合传统的"阳刚"、"威武"的特征，也可以"温柔"与"可爱"。

排名	形容男性姓名的形容词	形容女性姓名的形容词
1	伟大	倔强
2	狡猾	善良
3	憨厚	不幸
4	聪明	顽强
5	杰出	柔弱
6	倔强	漂亮
7	顽强	乐观

Table 7: 形容男性姓名和女性姓名的高频形容词TOP7

Table 7列出了形容男性姓名和形容女性姓名最多的7个形容词，也可以看出这些形容词都是侧重于形容人的性格特征，但对于女性来说，还包括了其外貌特征。Figure 5则反映了形容男性姓名和女性姓名最多的形容词"伟大"和"倔强"的词频历时变化。这可以体现出社会从关注"伟大"的男性到关注平凡普通的男性个体，并且越来越关注女性独立的个性。Figure 6则表明了过去平面媒体对人物的形容比较集中，主要为了体现伟人的引领作用，是一种集体主义精神的展现，如"伟大的毛泽东"，"伟大的马克思"；而现在，特别是改革开放后则关注到了每一个个体，且对不同个体的形容也体现了差异性和独特性。

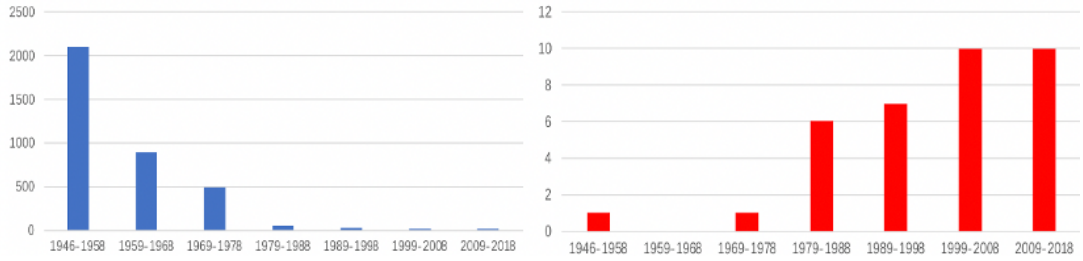


Figure 5: 男性姓名前搭配“伟大”（左）和女性姓名前搭配“倔强”（右）的词频历时变化

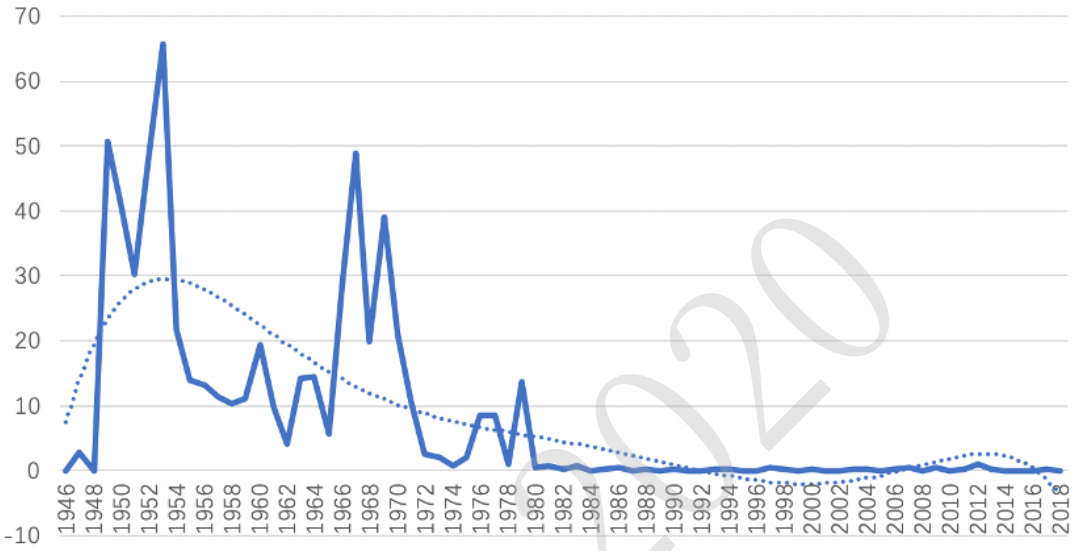


Figure 6: 搭配组数标准差的历时变化

5 结语

本文利用调查问卷和语料库的方法全面考察了汉语中形容词性别偏度的历时变化。本文的主要贡献有：构建了一个包含性别偏度的形容词数据集AGSS，可服务于未来的自然语言处理性别偏见消除任务；考察了不同群体的形容词性别偏度的差异并将形容词的性别偏度和其情感极性、情感强度联系起来；利用历时《人民日报》语料研究了不同时期的形容词性别偏度的差异，并和政治事件、经济发展等因素相联系，得出结论：“文革”期间呈现极偏向男性的语言风格，使用性别更平衡的形容词的时期，其经济发展水平更好；和不同性别的姓名搭配的形容词的历时变化，可以反映出我国的媒体从关注具有引领作用的伟人(”伟大”)到关注社会中的平凡却又独特的个体（“倔强”）的转变。鲁迅先生在其杂文《阿金》中曾写道：“我一向不相信昭君出塞会安汉，木兰从军就可以保隋；也不相信妲己亡殷，西施沼吴，杨妃乱唐的那些古老话。我以为在男权社会里，女人是决不会有这种大力量的，兴亡的责任，都应该男的负。但向来的男性的作者，大抵将败亡的大罪，推在女性身上，这真是一钱不值的没有出息的男人 (鲁迅, 1973)。”或许当我们的社会进入到了以一个人本身的特点而对其进行评价，而非利用群体的特性对其进行概括的时候，我们就真正走入了性别平等的新世界。

致谢

本文受教育部人文社会科学研究规划基金资助项目（18YJA740030）资助。感谢北京大学苏祺老师，北京师范大学胡韧奋老师，北京大学万明瑜博士、周洁同学，北京语言大学张颖、潘月、张三乐、杜冰洁、王伟康、邢百西同学参与讨论。感谢匿名评审老师提出的修改建议。感谢所有参与问卷调查的被试。

参考文献

Alvarez-Carmona M. A., Pellegrin L., and Montes-y-Gómez M., et al. 2018. A Visual Approach for Age and Gender Identification on Twitters. *Journal of Intelligent Fuzzy Systems*, 34(5):3133-3145.

Baker P.. 2010. Will Ms. Ever Be as Frequent as Mr.? A Corpus-based Comparison of Gendered Terms across Four Diachronic Corpora of British English. *Gender and Language*, 4(1)5.

Bolukbasi T., Chang K. W., and Zou J. Y., et al. 2016. Man Is to Computer Programmer as Woman Is to Homemaker? Debiasing Word Embeddings. *Advances in Neural Information Processing Systems*, 4349-4357.

Caliskan A., Bryson J. J., and Narayanan A.. 2017. Semantics Derived Automatically from Language Corpora Contain Human-like Biases. *Science*, 356(6334):4349-4357.

Fast E., Vachovsky T., and Bernstein M. S.. 2016. Quantifying Linguistic Signals of Gender Bias in An Online Fiction Writing Community. *Tenth International AAAI Conference on Web and Social Media*.

Garg N., Schiebinger L., and Jurafsky D., et al. 2018. Word Embeddings Quantify 100 Years of Gender and Ethnic Stereotypes. *Proceedings of the National Academy of Sciences*, 115(16):E3635-E3644.

Garcia D., Kassa Y. M., and Cuevas A., et al. 2018. Analyzing Gender Inequality through Large-scale Facebook Advertising Data. *Proceedings of the National Academy of Sciences*, 115(27):6958-6963.

Gaut T., Sun T., and Tang S., et al. 2020. Towards Understanding Gender Bias in Relation Extraction. *Proceedings of The 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*

Gonen H. and Goldberg Y.. 2019. Lipstick on A Pig: Debiasing Methods Cover up Systematic Gender Biases in Word Embeddings but Do Not Remove Them. *In Proceedings of the 2019 Conference of the NAACL: Human Language Technologies*, Volume 1 (Long and Short Papers):609–614.

Gonen H., Kementchedjhieva Y., and Goldberg Y.. 2019. How Does Grammatical Gender Affect Noun Representations in Gender-Marking Languages? *Proceedings of the 23rd Conference on Computational Natural Language Learning*, 463-471.

Graells-Garrido E., Lalmas M., and Menczer F.. 2015. First Women, Second Sex: Gender Bias in Wikipedia. *Proceedings of the 26th ACM Conference on Hypertext Social Media*, 165-174.

Hoyle A. M., Worlf-Sonkin L., and Wallach H., et al. 2019. Unsupervised Discovery of Gendered Language through Latent-Variable Modeling. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1706-1716.

Karimullah K.. 2020. Sketching Women: A Corpus-based Approach to Representations of Women's Agency in Political Internet Corpora in Arabic and English. *Corpora*, 15(1):463-471.

Kathleen A.. 2009. Politics, Gender and Conceptual Metaphors. Palgrave Macmillan, London, UK.

Madaan N., Mehta S., and Angrawaal T. S., et al. 2018. Analyze, Detect and Remove Gender Stereotyping from Bollywood Movies. *Conference on Fairness, Accountability and Transparency*, 92-105.

Michela M. and Monica R.. 2017. Gender Bias and Sexism in Language. *Oxford Research Encyclopedia, Communication*.

Sapir E.. 1929. The Status of Linguistics as a Science. *Language*, 5(4):207-214.

Sun T., Gaut A., and Tang S., et al. 2019. Mitigating Gender Bias in Natural Language Processing: Literature Review. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1630-1640.

Suzanne C. and Julia H.. 2014. *Gendering Women: Identity and Mental Wellbeing through the Lifecourse*. Policy Press, Clifton, UK.

Tan Y. C. and Celis L. E.. 2019. Assessing Social and Intersectional Biases in Contextualized Word Representations. *Advances in Neural Information Processing Systems*, 13209-13220.

Wagner C., Garcia D., and Jadidi M., et al. 2015. It's A Man's Wikipedia? Assessing Gender Inequality in An Online Encyclopedia. *Ninth International AAAI Conference on Web and Social Media*.

Wevers M.. 2019. Using Word Embeddings to Examine Gender Bias in Dutch Newspapers, 1950-1990. *Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change*, 92-97.

Whorf B. L. and Carroll J. B.. 1956. *Language, Thought, and Reality*. The MIT Press, Massachusetts, US.

Wilson F.. 1992. Language, Technology, Gender, and Power. *Human Relation*, 45(9):892-898.

Zhao J., Wang T., and Yatskar M., et al. 2017. Men also Like Shopping: Reducing Gender Bias Amplification Using Corpus-level Constraints. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2979-2989.

陈建美. 2008. 中文情感词汇本体的构建及其应用. 大连理工大学硕士学位论文. 大连, 中国.

陈莉娜. 2005. 小学语文教科书的人物性别、题材、人格形容词的定量研究. 第十届全国心理学学术大会论文摘要集.

崔红和王登峰. 2005. 中国人性别角色形容词评定量表的建构. 中国行为医学科学, 10:948-950.

刁晏斌. 2006. 现代汉语史. 福州人民出版社, 福州, 中国.

付超. 2018. 汉语形容词性别倾向性的考察及习得研究. 第十一届中文教学现代化国际研讨会论文集.

傅玉芳. 2010. 常用形容词分类词典*(*第三版*)*. 上海大学出版社, 上海, 中国.

黄伯荣和廖序东. 2017. 现代汉语*(*增订六版*)(*上册*)*. 高等教育出版社, 北京, 中国.

金灿灿和邹泓. 2009. "新人类"择偶征友标准及特点——基于BBS征友. 网络财富, 12:178-179.

靖元. 2007. 从网络征友看当代青年的择偶标准——对"我爱南开BBS"鹊桥版内容分析友. 青年研究, 02:9-16.

李逊和裴宜理. 1993. 革命的粗野. 文学自由谈, 4:33-39.

刘旭阳. 2017. 美国社会性别差异语义学之管窥:基于当代美国英语语料库的"Boy"和"Girl"搭配形容词描写. 天水师范学院学报, 37(01):67-72.

鲁迅. 1973. 且介亭杂文. 人民文学出版社, 北京, 中国.

孟秀文. 2008. 网上征友的自我呈现及其性别差异研究. 华中科技大学硕士学位论文. 武汉, 中国.

聂晶和郭明珠. 2010. 女大学生网络征友态度的十年变化状况——以北京大学未名BBS为基础的调查分析. 北京教育*(*德育*)*, 10:70-72.

邱明波. 2008. 亚文化视野下的军事词语泛化研究. 广西大学硕士学位论文. 南宁, 中国.

王悦和齐畅. 2011. 形容词运用的中性化趋势研究. 吉林省教育学院学报, 27(7):116-118.

杨晨和陈增祥. 2019. 数字有形状吗?数字信息精确性和品牌标识形状的匹配效应. 心理学报, 6:1-16.

张进. 2010. 基于剧本语料库的性别词汇研究. 文学界*(*理论版*)*, 4:118-119.

郑也夫. 1993. 礼语　咒词　官腔　黑话. 光明日报出版社, 北京, 中国.

中国社会科学院语言研究所词典编纂室. 2005. 现代汉语词典*(*第五版*)*. 商务印书馆, 北京, 中国.

周有光. 1995. 语文闲谈*(*上*)*. 三联书店, 北京, 中国.

用计量风格学方法考察《水浒传》的作者争议问题
——以罗贯中《平妖传》为参照

宋丽
清华大学人文学院
中国语言文学系，北京 100084
song-l19@mails.tsinghua.edu.cn

刘颖
清华大学人文学院
中国语言文学系，北京 100084
yingliu@mail.tsinghua.edu.cn

摘要

《水浒传》是独著还是合著，施耐庵和罗贯中是何关系一直存在争议。本文将其作者争议粗略归纳为施耐庵作、罗贯中作、施作罗续、罗作他续、施作罗改五种情况，以罗贯中的《平妖传》为参照，用假设检验、文本聚类、文本分类、波动风格计量等方法，结合对文本内容的分析，考察《水浒传》的写作风格，试图为其作者身份认定提供参考。结果显示，只有罗作他续的可能性大，即前70回为罗贯中所作，后由他人续写，其他四种情况可能性都较小。

1 引言

《水浒传》是我国古典长篇小说四大名著之一，讲的是北宋宣和年间，一众好汉聚义梁山泊反抗欺压，后被朝廷招安，为国征战，损兵折将，最终消亡的故事。然而《水浒传》的作者究竟是谁众说纷纭，施耐庵与罗贯中是否为同一人也尚无定论。温庆新(2018)根据多项研究成果指出明代人关于《水浒传》的作者主要有三种代表性意见，分别为施耐庵作、罗贯中作、施罗合作。现代文学界也围绕《水浒传》的作者身份展开了大量研究。刘冬(1992)、洪东流(2008)等认为施耐庵就是兴化人施彦端；王晓家(1998)、顾文若&焦中栋(1999)等认为施耐庵是罗贯中为避文祸而取的托名；罗尔纲(1984)从赞词、叙事、对待人民大众的态度等三方面对《水浒传》百回本和罗贯中《三遂平妖传》（即《平妖传》二十回本）进行对勘，得出《水浒传》前70回为罗贯中一人所著，后30回为他人续加的结论；吕乃岩(2008)通过将《水浒传》后半部分与罗贯中的其他作品相对照，得出《水浒传》前半部分为施耐庵所作，后半部分为罗贯中续加的结论；杨林(1999)、欧阳健(2003)、李永祜(2011)等认为《水浒传》是施耐庵作底本，罗贯中改写而成的。这些研究大多是从文献学、历史学的角度展开的，或多或少会存在主观性和疏漏，本文则利用统计学原理和计算机技术抽取文本特征，采用计量风格学的多种方法对《水浒传》的写作风格进行考察，试图为《水浒传》的作者身份认定提供一些参考。

由于《平妖传》是罗贯中所作的另一部著名小说[0]（或为冯梦龙增补，下文将对其版本问题加以说明），可以用于考察罗贯中与《水浒传》之间的关系，而施耐庵除《水浒传》外并无其他传世之作，故本文以罗贯中的《平妖传》为参照，将《水浒传》的作者身份争议粗略地划分为以下五种情况：

（1） **施耐庵作**：施耐庵与罗贯中并非同一人，《水浒传》为施耐庵一人独作，即《水浒传》和《平妖传》分别为两人所作，《水浒传》全书的写作风格应保持一致，且与《平妖传》不同。

（2） **罗贯中作**：施耐庵是罗贯中的托名，《水浒传》为罗贯中一人独作，即《水浒传》和《平妖传》为同一人所作，《水浒传》全书的写作风格应保持一致，且由于两部小说题材不同，故《水浒传》的写作风格会与《平妖传》有所差异。

（3） **施作罗续**：施耐庵与罗贯中并非同一人，《水浒传》前半部分为施耐庵所作，后半部分为罗贯中续写，即《水浒传》后半部分的作者与《平妖传》为同一人，写作风格应较为相近，且与《水浒传》前半部分有所差别。

[0]由于罗贯中的其他作品目前还没有高质量的人工标注文本，且现有的自动分析工具遵循的分词和词性标注标准与本文使用的台湾中央研究院近代汉语标记语料库并不完全相符，无法通过自动分析直接进行量化对比，故本文仅选取《平妖传》为参照文本。另外，由于施耐庵的真实身份、生卒年代尚无定论，故本文选取参照作品时忽略时代因素。对于其他相关作品的分析将在后续研究中进一步展开。

（4） **罗作他续**：施耐庵是罗贯中的托名，《水浒传》前半部分为罗贯中所作，后半部分为不知名的他人续写，即《水浒传》前半部分的作者与《平妖传》为同一人，写作风格应较为相近，且与《水浒传》后半部分有所差别。

（5） **施作罗改**：施耐庵与罗贯中并非同一人，《水浒传》底本由施耐庵所作，后罗贯中对全书进行了修改润色，但修改比重不详。也就是说《平妖传》的作者对《水浒传》进行了修改，使得《水浒传》的写作风格与《平妖传》有相似之处，但具体对哪些部分进行了修改，以及修改的篇幅则有待考察。

《水浒传》自传世以来出现了多种版本，据《水浒书录》(马蹄疾, 1986)可知，明刊本有25种，清刊本有47种。这些版本可大体分为繁本和简本两个系统，且文学界对繁本更为重视。繁本系统主要包括七十回本、百回本和百二十回本三种规模，每种规模又包括若干个版本。从大体上说，这三种规模的版本是内容依次增加的关系，七十回本的内容与百二十回本中的前70回基本一致，写到分封一百单八将为止；百回本在七十回本的基础上增加了接受招安、抗击辽国军队和剿灭方腊义军的故事；百二十回本又在百回本征方腊的情节之前增加了剿灭王庆和田虎所率领的起义队伍的故事，百二十回本中较为通行的是袁无涯本。《平妖传》有二十回本和四十回本两个版本系统，且它们分别有多种刻本，作者问题也存在一定争议：孙楷第(1958)等认为二十回本是罗贯中的原著，而四十回本为冯梦龙增补（这种观点较为流行）；欧阳健(1985)等认为四十回本是罗贯中原著，而二十回本是其删节本；等等。尽管如此，罗贯中是《平妖传》的主要作者这一观点几乎得到公认，也就是说无论哪个版本，《平妖传》中必然会体现出罗贯中的写作风格。由于受时代所限，近代汉语小说在版本方面大多都存在有待考证的问题，难以避免，且版本的考证问题并非本文的重点，故本文仅以内容最为完整且最为通行的《水浒传》繁本百二十回本中的袁无涯本、《平妖传》四十回本中的嘉会堂本为研究对象，运用计量语言学的研究手段，对《水浒传》的写作风格进行考察，从而推测以上五种情况的可能性大小。

由于《水浒传》作者身份争议较大的部分主要体现在前70回与后50回的差异，所以本文以70回结尾为界，将《水浒传》分为两部分。为方便阐述，下文分别用“SHZa”“SHZb”和“PYZ”代指三个待考察总体，SHZa中包含70个样本，即《水浒传》的前70回；SHZb中包含50个样本，即《水浒传》的后50回；PYZ中包含40个样本，即《平妖传》的全40回。另外，使用$Diff(X,Y)$表示两个总体之间的差异程度，如$Diff(SHZa,SHZb)$表示SHZa和SHZb之间的差异程度。上述五种作者身份的情况可简单总结为表1。

序号	作者争议	说明	量化表示
1	施耐庵作	施耐庵一人独作120回	$Diff(SHZa,SHZb)$最小，且$Diff(SHZ,PYZ)$较大
2	罗贯中作	罗贯中一人独作120回	$Diff(SHZa,SHZb)$最小，$Diff(SHZ,PYZ)$也较小
3	施作罗续	施作前70回，罗作后50回	$Diff(SHZb,PYZ)$最小
4	罗作他续	罗作前70回，不知名者作后50回	$Diff(SHZa,PYZ)$最小
5	施作罗改	施作底本，罗改写，改写程度不详	/

Table 1: 《水浒传》作者的主要争议情况归纳

2 语料概况及特征提取

两部小说的电子文本均来源于台湾中央研究院近代汉语标记语料库[1]，语料均已进行了人工分词和词性标注，质量较高，且均为繁体中文（该语料库中使用的词类标记共计59个，排除标题、外文标记、被解释的字、记录者姓名，以及两部小说中均未出现的引述句这5项，本文考察的词类共计54项，其中虚词13项，实词41项）。由于古代汉语使用句读进行断句，数字化古籍中的标点符号都是由古籍整理人员根据现代汉语标点规范人为增添的，因此下文中提及的所有统计数据和文本特征均不考虑标点。两部小说的基本统计数据见表2。

[1]http://lingcorpus.iis.sinica.edu.tw/cgi-bin/kiwi/pkiwi/kiwi.sh?ukey=-1316487165&qtype=0。

关于文本特征，一方面，本文提取了作者身份识别任务中较为常用，且已在多项实验中被验证有效的能够体现作者写作风格的特征，包括语法单元的长度、虚词、词类等。另一方面，虽然文本中实词的相关特征通常用于考察文本内容和主题，但也有不少学者从用词情况、对人物形象的塑造等角度来分析《水浒传》的作者(何心, 1954; 杨林, 1999; 李永祜, 2011)，可见考察《水浒传》的作者身份不能完全不考虑实词。为避免与内容高度相关的特征对作者身份判定造成影响，本文只补充提取在三个总体中均出现的高频实词Top500的词频作为特征。提取的特征共计1808个，见表3。

书名	《水浒传》	《平妖传》
章回数	120	40
总字数	726833	189356
总词数	546969	146134
总字型数	4217	3459
总词型数	19603	10922
总整句数	44225	9634
总小句数	114625	29034
平均每回词数	4558.08	3653.35

⁎ 由于该语料库采用Big5字符集，有少量字符无法显示，且文中所有诗词均被删除，故上表中关于字数、类符的统计结果有些许误差。

Table 2: 《水浒传》和《平妖传》语料的基本统计数据（不含标点）

特征	数量
词长、整句长、小句长的均值	3
词长、整句长、小句长的离散度	3
词汇丰富度（类符形符比）	1
每个虚词的出现次数（每千词）	247
在三个总体中均出现的高频实词Top500的出现次数（每千词）	500
不同词类的数量（每千词）	54
二元词类标记串[2]Top500的出现次数（每一千个标记串）	500
三元词类标记串Top500出现次数的（每一千个标记串）	500
总计	1808

⁎ 虽标点符号均为后人所加，但断句情况在一定程度上也能体现作者的写作风格，所以本文也提取了整句和小句的长度及离散度作为特征。

Table 3: 提取的特征及其数量

3 “施耐庵作”&“罗贯中作”&“施作罗续”的可能性分析

首先使用假设检验和聚类分析的方法对“施耐庵作”“罗贯中作”和“施作罗续”这三种情况的可能性进行考察。

3.1 假设检验

假设检验[3]可用于考察不同总体之间是否存在显著差异。由于本文考察的三个总体的样本量不大且不相等，为避免数据分布和方差的差异对假设检验结果造成影响，本文先针对每个特征，分别对三个总体做Shapiro-Wilk检验（适用于样本量小于2000时的正态性检验）(Shapiro & Wilk, 1965)和Levene方差齐性检验(Levene, 2004)，抽取出了在三个总体中均服从正态分布（显著性水平$\alpha = 0.05$[4]，下同）且满足方差齐性的特征56个（包括高频实词“去”、虚词“了”、词类VC（动作及物动词）、平均小句长等）。然后，分别针对这56特征，对SHZa、SHZb和PYZ这三个总体做单因素方差分析（可用于考察多个正态总体的均值之间差异是否显著，仅涉及一个变量），结果显示三个总体在47个方面存在显著差异。为进一步确认显著差异存在于哪几组数据之间，还利用LSD检验法(Fisher, 1935)进行了多重比较，得到了每两个总体中有显著差异的特征的数量，见表4（最小数量加粗，下同）。

针对余下特征，对三个总体做Kruskal-Wallis H检验[5]（通过秩和均值来考察多个总体差异是否显著，不依赖总体的分布，仅涉及一个变量。以下简称K-W检验）(Kruskal & Wallis, 1952)，结果显示三个总体在1227个方面存在显著差异，进一步做多重比较(Gibbons & Chakraborti, 2011)，每两个总体中有显著差异的特征数量见表4。

[2]“词类标记串”是指已进行词性标注的文本中相邻的n个词类标记形成的串，如句子“莊客/Na 報知/VE 史進/Nb”中三个词的词类标记分别为“Na（普通名词）”“VE（动作句宾动词）”“Nb（专有名称）”，则“Na VE”和“VE Nb”为二元词类标记串。“Na VE Nb”为三元词类标记串。

[3]假设检验的具体方法可见《概率论与数理统计》(茆诗松&周纪芗, 2000)。

[4]统计学上一般把概率⩽0.05的事件称为小概率事件。

[5]有学者将Kruskal-Wallis H检验称作单因素方差分析的非参数方法。

被比较对象	差异显著的特征数量		
	LSD检验（共47个特征）	K-W多重检验（共1227个特征）	合计
SHZa vs. SHZb	33	622	655
SHZa vs. PYZ	**29**	**419**	**448**
SHZb vs. PYZ	41	673	714

Table 4: 每两个总体中差异显著的特征数量对比

观察表4可知，在被实施多重比较的特征中，不论是进行LSD检验的，还是进行K-W多重检验的，《水浒传》前70回与《平妖传》呈现显著差异的特征数量都是最少的，也就是说与其他两组相比，它们在较少的方面存在差异。众所周知，这两部小说的题材不同（《水浒传》为英雄传奇小说，《平妖传》为神魔小说），但多重比较结果却显示出自同一部小说的《水浒传》前70回和后50回之间的差异程度较大，与《水浒传》后50回和《平妖传》的差异特征数量接近，反而是分别出自不同题材的两部小说的《水浒传》前70回和《平妖传》差异程度较小。

3.2 文本聚类

接下来采用两种较为常用的聚类方法——K均值聚类和层次聚类[6]来对全部160个样本进行聚类分析，考察两部小说中哪些章回的写作风格相似度高，哪些章回的写作风格差异大，写作风格相似度越高的章回越有可能出自同一位作者。为消除量纲的影响，在预处理时对所有特征均进行了Z-score标准化处理[7]。（本文使用的聚类和分类方法均在预处理时对特征进行了标准化处理，下文不再赘述。）

K均值聚类以k为输入参数，把n个对象的集合分为k个簇。本文指定$k=2$（即分为两类），文本之间的相似度使用欧氏距离计算。分类结果见表5（数字加粗表示被分到相应类别的章回数较多）。由于样本数据处于一个高维空间中，K均值聚类的结果难以用平面图表示，所以采用主成分分析法（Principal Component Analysis，简称PCA）(Jolliffe, 1986)进行维数约减，提取了最重要的两个主成分（保留的信息量分别为0.11和0.05），将结果映射到二维平面，作散点图1。从图中可见，160个样本明显被分为了两类，也就是说仅0.16的信息量即可得到近似线性二分的结果。

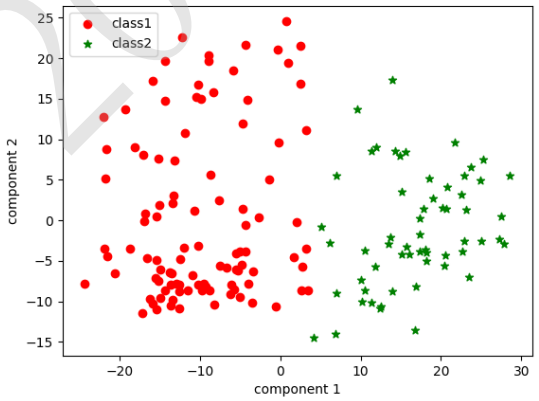


Figure 1: 两部小说各章回的K均值聚类结果（主成分Top2）

层次聚类将数据对象组成一棵聚类树，无需指定类别数量。本文用欧氏距离计算文本之间的相似度，用离差平方和计算类与类之间的相异度。聚类结果见图2，其中横坐标为各章回编号，纵坐标为文本之间的相似度。以150为界作一条横线，可将160个样本分为两类，类别内部的欧氏距离均小于150，而两个类别之间的欧氏距离超过了200。各章回的具体分类情况见表5。

观察K均值聚类和层次聚类的结果，发现《水浒传》前70回和《平妖传》中的大部分文本都聚在了同一类，而《水浒传》后50回中的大部分文本则聚在了另一类，这说明三个总体中，《水浒传》前70回和《平妖传》之间的差异程度最小，与第3.1节中假设检验的结果相符。

综合假设检验和聚类分析的结果可知，三个总体之间差异程度的关系应为：$Diff(SHZa, PYZ) < Diff(SHZb, PYZ) < Diff(SHZa, SHZb)$。然而根据表1可知，若《水浒传》为“施耐庵作”或“罗贯中作”，则$Diff(SHZa, SHZb)$应最小；而若是“施作罗续”，则$Diff(SHZb, PYZ)$应最小，据此可推断，“施耐庵作”“罗贯中作”和“施作罗续”这三种情况的可能性小。

[6]聚类分析的具体方法可见《数据挖掘：概念与技术》(Han & Kamber著, 范明&孟小峰译, 2007)。
[7]公式为$x^* = \frac{x-\bar{x}}{\sigma}$。

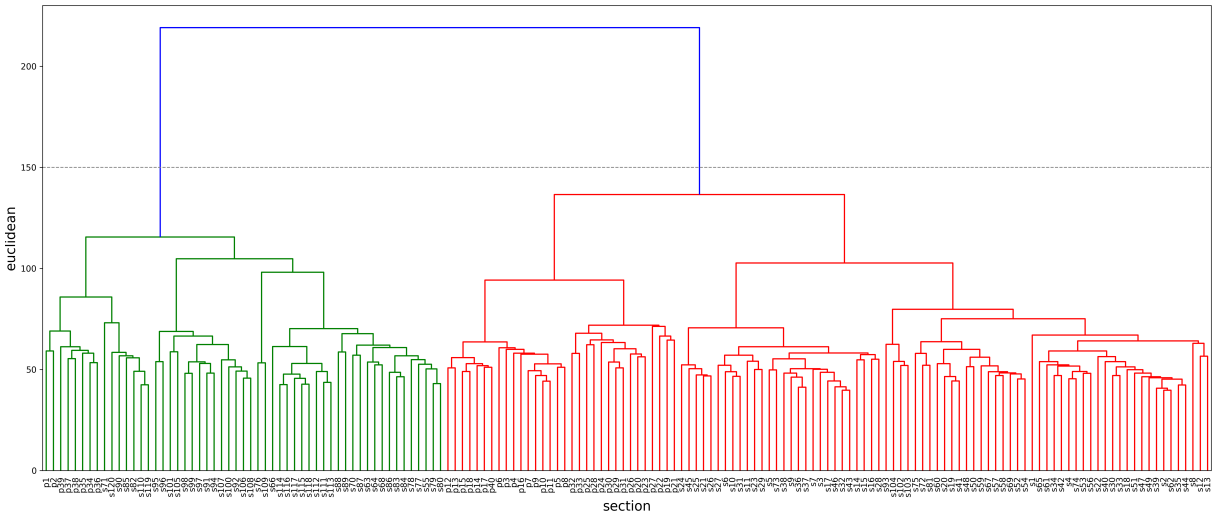


Figure 2: 两部小说各章回的层次聚类结果

文本	K均值聚类				层次聚类			
	类别一		类别二		类别一		类别二	
	章回编号	数量	章回编号	数量	章回编号	数量	章回编号	数量
SHZa	1-47; 49; 51-53; 56; 61-62; 65	**55**	48; 50; 54-55; 57-60; 63-64; 66-70	15	1-54; 56-62; 65; 67; 69	**64**	55; 63-64; 66; 68; 70	6
SHZb	72-74; 81; 93; 102-104	8	71; 75-80; 82-92; 94-101; 105-120	**42**	72; 74-75; 81; 93; 102-104	8	71; 73; 76-80; 82-92; 94-101; 105-120	**42**
PYZ	1-33; 35; 37-40	**38**	34; 36	2	3-33; 40	**32**	1-2; 34-39	8

Table 5: 三个总体中各章回的聚类结果详情

4 “罗作他续”的可能性分析

由第3节可知，SHZa、SHZb、PYZ这三个总体之间差异程度最小的应为SHZa和PYZ，这一结果与“罗作他续”相匹配。为进一步验证这一情况，本文利用文本分类方法支持向量机（Support Vector Machine，简称SVM）(Cortes & Vapnik, 1995)，从作者身份归属（Authorship Attribution，即判断待分类文本的作者是已知作者中的哪一个）和作者身份验证（Authorship Verification，即判断待分类文本的作者与已知作者是否为同一人）两个角度对上述三个总体加以考察。

4.1 作者身份归属

从作者身份归属的角度出发，假定《水浒传》前70回为罗贯中所作，后50回为另一不知名作者所作，考察《平妖传》中的各章回被分类到《水浒传》前70回和后50回的可能性大小。也就是说，假定SHZa和SHZb中的样本被分为不同的两类，用SVM考察PYZ中的各样本被分到这两类的可能性。由于SHZa和SHZb中的样本数量不均衡（70 vs. 50），故在实际操作时，随机剔除SHZa中的20个样本进行训练，重复10次，以最终输出的概率均值为预测结果，作堆积柱状图3。图中横坐标为PYZ中各章回的编号，纵坐标为概率值，每一章回被分到SHZa和SHZb的概率值之和都为1。

虽然总体看来，PYZ中的文本被分类到SHZa和SHZb的几率大致相当，但若除去开头和结尾的几个章回，则更多地被分到SHZa中。研究《平妖传》的多数学者认为其二十回本为罗贯中所作，而四十回本是冯梦龙增补而成。谭红(2013)从多方面对这两个版本进行了细致的对比分析，综合其分析结果可知，与二十回本相比，人物方面，四十回本中新增的人物（如九

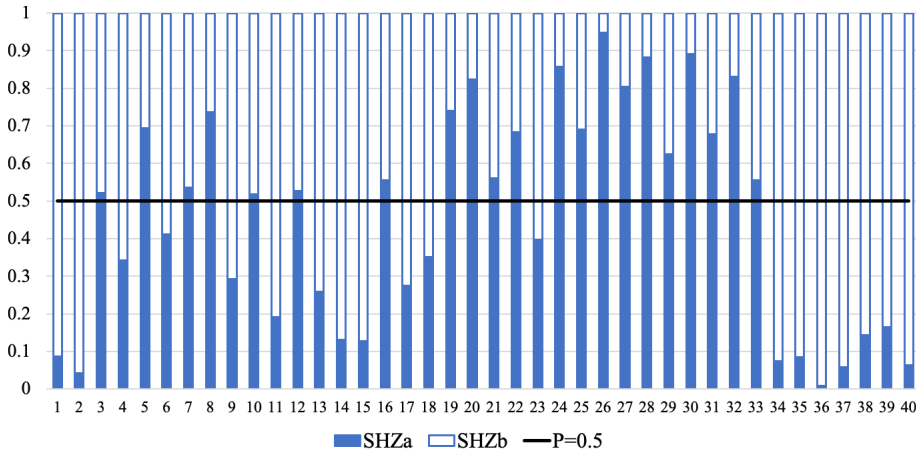


Figure 3: 《平妖传》中各章回被分类到《水浒传》前70回和后50回的可能性

天玄女、白猿神）和对原有人物（如王则、蛋子和尚）的改写大多出现于开篇第1回和结尾第37-40回；情节方面，四十回本新增和大幅扩充的内容主要出现在第1-18，34-40回。也就是说，中间的章回，尤其是第19-33回基本保留了罗贯中二十回本的原貌。这一结论与图3的结果高度匹配，可见《平妖传》中保留的罗贯中的写作风格与《水浒传》前70回很相近，所以《平妖传》与《水浒传》前70回的作者很可能都是罗贯中，这说明"施耐庵"可能只是罗贯中的一个托名，而《水浒传》前后写作风格差异大，后半部分可能是由他人续写。这一结果支持了"罗作他续"的说法。

4.2 作者身份验证

Koppel & Schler (2004)指出，由于主题、体裁、写作目的、写作年代等因素的变化，甚至出于掩饰身份的目的，同一个作者所作的不同作品在少量特征方面会有明显不同。也就是说，这些特征对作者身份的验证造成了困难。为了解决这种困难，他们提出了"揭露（Unmasking）算法"，其思路为将作者身份已知的文本和作者身份未知的文本拆分为多个组块，并假定它们分属不同的两类，不断删除对区分这两类文本最有用的特征，测量交叉验证时分类正确率下降的速度，若下降速度快，则表示两个文本难以区分，即作者为同一人，反之则作者并非同一人。在验证小说*The House of Seven Gables*的作者身份时，揭露算法在三位候选作者中明确选中了正确的作者Hawthorne[8]。Bevendorff et al. (2019)通过改变生成组块的方式，打破对组块长度的要求，将揭露算法引入了对短文本的作者身份验证任务中，也取得了与目前高性能算法相当的效果。本文参考揭露算法的思路，以小说章回为文本的自然组块，进行了如下实验：

迭代50次{

步骤①：对总体A和B，假定它们的作者并非同一人（即分属两个不同类别），利用LinearSVM模型训练分类器，做10折交叉验证，计算平均正确率

步骤②：去掉步骤①训练出的分类器中权重最高的5个特征和权重最低的5个特征

步骤③：回到步骤①

}

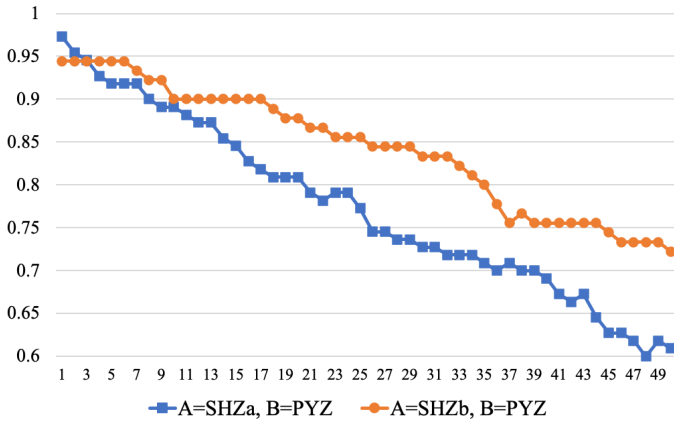


Figure 4: 揭露算法的分类正确率变化情况

[8]将*The House of Seven Gables*与其他作者的作品比较时，正确率下降缓慢，而与Hawthorne的作品比较时，正确率下降迅速。

分别设A=SHZa，B=PYZ；A=SHZb，B=PYZ[9]，各进行一次实验，将50次的分类正确率作折线图4，横坐标为迭代次数，纵坐标为交叉验证后的平均分类正确率。从图中可以很直观地看出，SHZa vs. PYZ的分类正确率的下降速度明显比SHZb vs. PYZ快，也就是说，《水浒传》前70回和《平妖传》的作者为同一人的可能性大，而《水浒传》后50回和《平妖传》的作者为同一人的可能性较小。这一结果也支持"罗作他续"的说法。

4.3 对诗词的补充分析

由于本文所考察的文本数据中的诗词均被剔除，也就是说上述统计数据并未能覆盖小说中诗词部分的信息，所以需对两部小说中的诗句、赞词等内容进行补充分析。罗尔纲(1984)曾对《水浒传》百回本和《平妖传》二十回本中的赞词进行对勘，发现《平妖传》二十回本中

水浒传		平妖传	
诗词	章回	诗词	章回
祥云迷凤阁，瑞气罩龙楼……隐隐净鞭三下响，层层文武两班齐。	1	祥云迷凤阙，瑞气罩龙楼……隐隐净鞭三下响，层层文武两班齐。	35
祥云笼凤阙，瑞霭罩龙楼。	72		
香焚宝鼎，花插金瓶……琥珀杯中，满泛着瑶池玉液……鳞鳞脍切银丝，细细茶烹玉蕊。	2	香焚宝鼎，花插金瓶……琥珀杯中，满泛瑶池玉液……玻璃碗，供熊掌驼峰。鳞鳞脍切银丝，细细茶烹玉蕊。	25
玻璃碗内，供献上熊掌驼蹄；琥珀杯中，满斟下瑶池玉液。	119		
桂花离海峤，云叶散天衢……冰轮展出三千里，玉兔平吞四百州。	2	桂华离海峤，云叶散天衢……冰轮碾破三千界，玉魄横吞万里秋。	18
门迎黄道，山接青龙。	9	门迎黄道，山接青龙	9
银河耿耿，玉漏迢迢……贪淫妒女心如铁，仗义英雄气似虹。	21	银河耿耿，玉漏迢迢……妖邪贼侣心如，忠义英雄气似虹。	38
尽道丰年瑞，丰年瑞若何？长安有贫者，宜瑞不宜多。	24	纷纷柳絮，片片鹅毛……万户银装，多少行人肠断。正是：尽道丰年瑞，丰年瑞若何；长安有贫者，宜瑞不宜多。	18
纷纷柳絮，片片鹅毛……万户银装，多少幽人成佳句。正是：尽道丰年好，丰年瑞若何？边关多荷戟，宜瑞不宜多。	93		
平生正直，禀性贤明……慷慨文章欺李杜，贤良方正胜龚黄。	27	平生正直，禀性贤明……果然是慷慨文章欺李杜，贤良方正胜龚黄。	29
身穿缟素，腰系孝裙……恰似嫦娥离月殿，浑如织女下瑶池。	32	身穿缟素，腰系麻裙……恰似嫦娥离月殿，浑如织女下瑶池。	31
犯由牌高贴，人言此去几时回……长休饭颡内难吞，永别酒口中怎咽。	40	两声破鼓响，一棒碎锣鸣……犯由牌高贴，人言此去几时回……长休饭，喉里难吞。永别酒，口中怎咽……刀剑林中刽子手，犹如追命鬼。	40
两声破鼓响，一棒碎锣鸣……刀剑林中，掌法吏犹如追命鬼。	62		
骊山顶上，多应褒姒戏诸侯；赤壁坡前，有若周瑜施妙计。	41	骊山顶上，料应褒姒逞英雄。扬子江头，不若周郎施妙计。	18
金钉朱户，碧瓦雕檐……若非天上神仙府，定是人间帝主家。	42	金钉朱户，碧瓦雕檐……若非天上神仙府，定是人间帝王家。	25
		金钉朱户，碧瓦盈檐。	28
正大仙容描不就，威严形像画难成。	42	苍形古貌，鹤发童颜。眼昏似秋月笼烟，眉白如晓霜映日……正大仙客描不就，威严形像画难成。	25
苍然古貌，鹤发酡颜。眼昏似秋月笼烟，眉白如晓霜映日……	53		
一群白鹤听经，数个青衣碾药。	53	仙童击鼓，一群白鹤听经；玉女鸣钟，数个青猿煨药；	28
凤落荒坡，尽脱浑身羽翼……吕虔亡所佩之刀，雷焕失丰城之剑。	56	凤落荒坡，脱尽浑身锦羽……吕虔亡腰下之刀，雷焕失匣中之剑；	38
六尺以上身材，二十四五年纪……着一双土黄皮油膀胛靴。	61	六尺以下身材，二十二三年纪……着一对土黄色多耳皮鞋，	24
唇若涂朱，睛如点漆，面似堆琼。	61	面如傅粉，体似凝脂，唇若涂朱，目如点漆。	35
人人要建封侯绩，个个思成荡寇功。	91	人人欲建封侯绩，个个思成荡寇功。	35
凤眼浓眉如画，微须白面红颜……七尺身材壮健。善会偷香窃玉，	101	凤眼浓眉如画，黄须白面高颧……六尺身材壮健。善会开弓发弩，	31
神器从来不可干，僭王称号讵能安?	113	神器从来不可干，僭王称制谁能安。	40

Table 6: 两部小说中的相似诗词及其所在章回

[9]由于该实验主要关注的是删除特征后分类正确率的变化速度，所以并未对两类样本数量的不均衡做额外处理。

有13处赞词被直接或经改写后插入了《水浒传》百回本的前70回中（插入15处）。这些赞词均被相应地保留在了《水浒传》百二十回本和《平妖传》四十回本中。此外，本文还通过字符串比对[10]，在两部小说中找到了另外9处或长或短的相似诗词，例如《水浒传》第41回描写众好汉火烧黄文炳家的场景时用了一句"**骊山顶上**，多**应褒姒**戏诸侯；赤壁坡前，有**若周瑜施妙计**。"《平妖传》第18回描写胡员外家解库起火的场景时用了一句"**骊山顶上**，料**应褒姒**逞英雄。扬子江头，不**若周郎施妙计**。"这两句话的句式、用词、内容都很相似，只改动了个别字词（相同的字词加粗）。

表6罗列了两部小说中24处相似的诗词以及它们各自的章回信息（受篇幅所限，只摘录部分相似内容）。其中仅有6处出自《水浒传》后50回，其他全都出自《水浒传》前70回。可见，《水浒传》前70回和《平妖传》中的诗词有较高的相似度，这进一步佐证了SHZa和PYZ很可能是出自同一人之手。当然，不可否认的是，诗词的相似度高也有可能是因为不同作者挪用并改写，这有待进一步考证。

5 "施作罗改"的波动风格计量分析

商韬&陈年希(1999)指出，宋元以来的许多话本小说和章回小说往往不是个人的独立创作，而是时代积累型的集体创作。所以仅凭上述分析，不能直接断定"施耐庵"只是罗贯中的一个托名，《水浒传》是罗作他续。或许《水浒传》也并非一人独著或有人续写，而是一部集体创作的，无法区分各章回分别由谁所写的作品。接下来，本文假定《水浒传》的确符合第五种情况"施作罗改"，引入波动风格计量的方法对二位作者的风格比重加以分析。

波动风格计量（Rolling Stylometry）是由Eder提出的一种考察合作型文本的文体特征或写作风格的方法，它将有监督的机器学习分类方法与序列分析相结合。Eder (2016)指出线性序列中元素的顺序与元素本身同样重要，文本中连续的部分可以体现风格特征的线性发展。这一思想最早被应用于语言学是1913年Markov (2006)[11]提出的马尔科夫链（Markov chains），只不过马尔科夫链的滑动窗口只包含连续的几个字符，而波动风格计量方法的滑动窗口则包含连续的上百或上千个词，其目的在于从文本中生成一连串含有重叠内容，前后相互关联的虚拟子样本，从而可以使用有监督的机器学习分类方法测试它们在整个文本中的风格一致性。与传统的方法相比，波动风格计量分析方法打破了文本中预设的界线，如章回、段落等，转而对一系列连续的，长度相等且包含重叠成分的样本进行分类，因此可以用于考察合作型文本中不同作者各自的贡献份额的变化情况。图5是该方法抽取子样本的示意图，其中k表示每个子样本包含的词数，d表示连续两个子样本中重叠的内容长度。Plecháč (2019)利用Shakespeare和Fletcher的各4部剧作，以500个高频节奏类型和500个高频词为特征，测试了这种方法的有效性[12]，结果显示，在Shakespeare的剧作中，Fletcher的贡献程度都非常低，反之亦然，若根据概率值大小判断作者身份，正确率则高达0.9977，可见波动风格计量分析方法对作者贡献程度的度量具有较高的有效性。

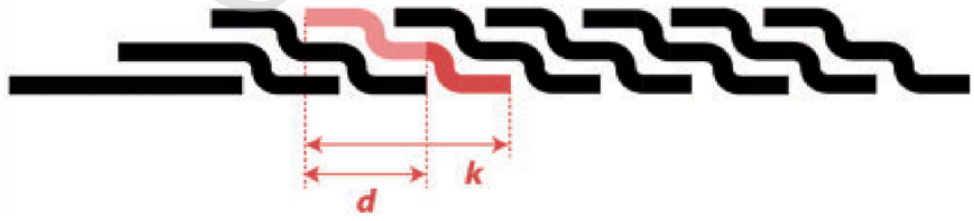


Figure 5: 波动风格计量方法抽取子样本的示意图（截取自Eder (2016)）

本文考察的《水浒传》的作者身份的第五种情况是"施作罗改"，其蕴涵为施耐庵与罗贯中并非同一人。由于除《水浒传》以外，施耐庵并没有其他传世之作，所以无法选用其他作品作为施耐庵的训练集。综合持"施作罗改"观点的多位学者的分析，他们虽然对施耐庵和罗贯中对《水浒传》的具体贡献各持己见，但几乎都认为施耐庵的底本主要是以宋江的故事为主线，罗贯中在前期保留了与宋江有关的主要情节，即约为后人总结的"宋十回"（第33-42回）。从

[10]诗词比对选用的是简体版文本。
[11]该篇论文成文于1913年，2006年由*Science in Context*期刊出版。
[12]与Eder (2016)不同的是，Plecháč (2019)设置的滑动窗口以剧作的台词（line）为单位，而非单个词。

第3.2节聚类分析的结果可知，这些章回也的确被聚在了同一类。所以，本文提取“宋十回”覆盖的97个样本作为施耐庵的训练样本，以《平妖传》的284个样本作为罗贯中的训练样本，用波动风格计量的方法对《水浒传》全书进行测试。

抽取子样本时，本文设定k=5000，d=4500，即对训练样本和测试样本，均每隔500词截取一个子样本，每个样本规模为5000词，也就是说每两个连续样本中有4500词的内容重叠。文本分类方法同样选取SVM。因滑动窗口中有句子被拆分，故排除了与句长相关的特征。另外，由于训练样本数量不均衡，故在实际操作时，随机选取《平妖传》中的97个样本参与训练，重复50次，以最终输出的概率均值为预测结果。图6显示了测试结果，其中横坐标为1085个测试子样本的编号（根据文本内容排序），纵坐标为SVM的分类可能性（概率值），由于是二分类问题，故每一个样本被分到两类的可能性之和均为1（为便于作图，分类为施耐庵的可能性用正数表示，分类为罗贯中的可能性用负数表示）。黑色曲线为分到两类的可能性的均值，若落在深色区域，则表示该部分内容主要为施耐庵所写；若落在浅色区域，则表示该部分内容主要为罗贯中所写。有底色的部分为抽取的施耐庵的训练样本，所以测试结果显示被分为施耐庵所作的可能性接近1。点线（……）为《水浒传》每10回的分界线，虚点线(—·—·—·)为前70回和后50回的分界线。

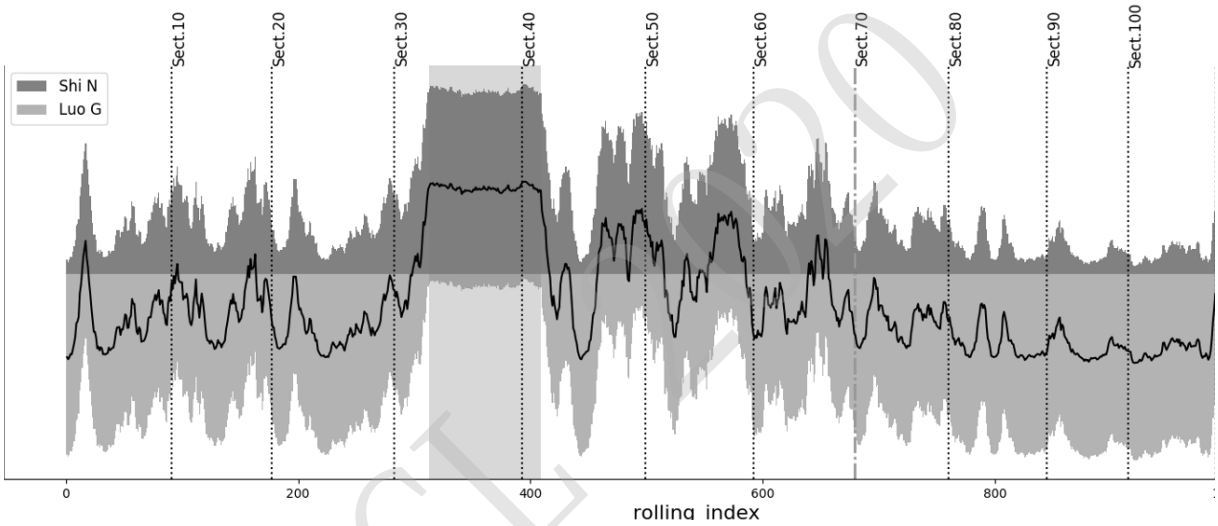


Figure 6: 《水浒传》作者身份识别的波动风格计量测试结果

从图6中可见，在假定《水浒传》是“施作罗改”的前提下，波动风格计量方法的测试结果的确显示《水浒传》全书中，二者的风格占比存在明显的波动。但是，除了本就被选为施耐庵训练样本的部分，其余部分的黑色曲线大多落在了罗贯中的区域，约第1-32，43-46，52-56，60-110，117-120回，也就是这些部分都主要由罗贯中所写。罗贯中对《水浒传》的贡献度很高，且显然远超施耐庵，由此可以推测，“施作罗改”的可能性较小，即使真的存在所谓施耐庵的底本，在经过罗贯中的改写后，也已经基本不复原貌。

由于事先已做了“施作罗改”的假定，且没有施耐庵的其他作品作参照，所以同样还需进一步结合文献学和历史学的知识对该测试结果加以考证。下面结合波动风格计量的测试结果与《水浒传》的文本内容，就相关研究成果中的部分观点加以比对分析：

（1）侯会(1986)、杨林(1999)认为《水浒传》开篇关于鲁智深、林冲等人的故事（即第1-13回）和描写武松的“武十回”（从第23回武松打虎开始，到第32回武行者醉打孔亮结束）这些与宋江的故事毫无联系的内容均为罗贯中增补。波动风格计量的测试结果显示，前32回都应主要出自罗贯中之手，所以该观点仅与测试结果部分相符。

（2）李永祜(2011)认为杨志押运生辰纲（第16回）是施耐庵重点描写的内容；宋江私放晁盖、杀掉阎婆惜后逃亡避难（分别在第18、21-22回）等情节均出自施耐庵之手；第72回柴进潜入宫禁，在睿思殿看到素面屏风的场景是施耐庵根据自己的真实经历模仿的；而宋江率军队征讨辽国（第83-89回）的故事是罗贯中设计出的鲁智深经历的重要关节。波动风格计量的测试结果显示，第16、18、21、22、72、83-89回均主要为罗贯中所作，所以这几个观点中只有征辽情

节由罗贯中设计这一个观点与测试结果相符。

（3）孟繁仁(1990)、宣啸东(1991)等认为“征田虎”部分的人物许贯忠是罗贯中在修改施耐庵创作的底本时创作出的自己的虚像，也就是说关于许贯忠的内容均是由罗贯中主笔撰写。原著中许贯忠于第90回作为燕青巧遇的故交出场，带他到家中做客，提醒他功成名就之时要及时抽身，送他画作，着墨较多，第91、94回也对许贯忠有所提及。波动风格计量的测试结果显示，罗贯中对第90-94回的改动的确较大，所以该观点与测试结果相符。

（4）杨林(1999)认为宋江等人平方腊有功，升官受赏是施耐庵底本的结局，而他们功成名就之后被害死的情节并非出自施耐庵之手，而是罗贯中在元、明朝代更迭之际看到乱世的残酷现实后所增益的内容。原著中征方腊的情节从第110回开始，到第119回以宋军平定方腊叛乱，梁山好汉得到封赏，宋江衣锦还乡结束，而宋江、卢俊义等人被蔡京、高俅等奸臣迫害致死的情节出现在第120回。波动风格计量的测试结果显示，罗贯中对第120回的改动的确较大，对第111-116回的改动较小，所以该观点与测试结果部分相符。

总的来说，波动风格计量分析的结果仅与“施作罗改”的少数观点相符，其他观点诸如施耐庵集撰了前人所作的水浒故事，罗贯中删去了施耐庵底本的一些内容等等，仅仅基于本文的测试结果无法得到验证，有待进一步考察。

6 结论及未来展望

本文以《水浒传》的作者身份为研究对象，根据前人研究将《水浒传》的作者争议粗略地归纳为施耐庵作、罗贯中作、施作罗续、罗作他续、施作罗改等五种情况，以罗贯中的另一部小说《平妖传》为参考，利用统计学原理和计算机技术对文本特征进行抽取，采用计量风格学的多种方法对《水浒传》的写作风格进行考察，试图为《水浒传》的作者身份认定提供参考。

首先通过假设检验、K均值聚类和层次聚类发现，在《水浒传》前70回、《水浒传》后50回、《平妖传》这三个总体中，《水浒传》前70回和《平妖传》的写作风格之间差异最小，从而得出结论：《水浒传》为施耐庵作、罗贯中作、施作罗续的可能性小，而罗作他续的可能性大。然后，从作者身份归属和作者身份验证两个角度出发，利用文本分类方法SVM，结合对两部小说中相似诗词的补充分析，进一步验证了罗作他续可能性大这一结论，这或许可以为“施耐庵是罗贯中的化名”这一观点提供佐证。此外，还采用波动风格计量的方法对施作罗改的情况加以考察，测试结果显示，“施作罗改”的可能性也较小，即使的确存在施耐庵的底本，罗贯中的改写也已使其基本不复原貌，通过与原著内容的比对可知测试结果仅与施作罗改的少数观点相符。综合而言，《水浒传》作者争议的五种情况中只有罗作他续可能性大，其他四种情况可能性都较小。

由于施耐庵与罗贯中究竟是什么关系这一问题尚无定论，而本文在采用有监督的机器学习分类方法时对施耐庵的身份做了假定，而且本文尚未考察“罗作他续”中“他”的身份，所以未来还需进一步结合文献学和历史学的知识对分类结果的可靠性加以考证，并探究罗作他续中“他”的身份。此外，未来还需进一步结合《水浒传》和《平妖传》的文本内容对本文的计量结果加以分析。

致谢

感谢审稿人的中肯建议。本研究得到了教育部社会科学研究一般项目“语体特征的自动提取和研究”（17YJAZH056），国家社会科学基金重大项目“基于大数据技术的古代文学经典文本分析与研究”（18ZDA238），以及清华大学人文社科振兴项目“基于大数据技术的明清小说分析和研究”（2019THZWJC38）的资助。

参考文献

Bevendorff J, Stein B, Hagen M, et al. 2019. Generalizing Unmasking for Short Texts. *North American Chapter of the Association for Computational Linguistics*, 654–659.

Cortes C, Vapnik V. 1995. Support-Vector Networks. *Machine learning*, 20(3):273–297.

Eder M. 2016. Rolling Stylometry. *Digital Scholarship in the Humanities*, 31(3):457–469.

Fisher R A. 1935. *The Design of Experiments*. Oliver and Boyd, Edinburgh and London.

Gibbons J D, Chakraborti S. 2011. *Nonparametric Statistical Inference*. Taylor and Francis Group, Florida.

Jolliffe I T. 1986. *Principal Component Analysis*. Springer Verlag, New York.

Koppel M, Schler J. 2004. Authorship Verification as a One-class Classification Problem. *International Conference on Machine Learning*.

Kruskal W, Wallis W A. 1952. Use of Ranks in One-criterion Analysis of Variance. *Journal of the American Statistical Associationt*, 47(260):583–621.

Levene H. 1960. Robust Tests for Equality of Variances. *Contributions to probability and statistics*, 278–292.

Markov A A. 2006. An Example of Statistical Investigation of the Text "Eugene Onegin" Concerning the Connection of Samples in Chains. *Science in Context*, 19(4):591–600.

Plecháč P. 2006. Relative contributions of Shakespeare and Fletcher in Henry VIII: An Analysis Based on Most Frequent Words and Most Frequent Rhythmic Patterns. *arXiv: Computation and Language*.

Shapiro S S, Wilk M B. 1965. An Analysis of Variance Test for Normality (Complete Samples). *Biometrika*, 52(3–4):591–611.

(加)Han J, Kamber M著. 范明, 孟小峰译. 2008. 数据挖掘：概念与技术. 机械工业出版社, 北京.

顾文若, 焦中栋. 1999. "施耐庵"为罗贯中之托名. 晋阳学刊, (1):107–108.

何心. 1954. 水浒研究. 上海文艺联合出版社, 上海.

洪东流. 2008. 水浒解密考. 学林出版社, 北京.

侯会. 1986. 《水浒》源流管窥. 文学遗产, (4):64–71.

李永祜. 2011. 施耐庵和罗贯中对《水浒传》成书的贡献. 菏泽学院学报, 33(4):24–37.

刘冬. 1992. 施耐庵探考. 南京出版社, 南京.

罗尔纲. 1984. 从罗贯中《三遂平妖传》看《水浒传》著者和原本问题. 学术月刊, (10):22–32.

吕乃岩. 2008. 试说罗贯中续《水浒》. 北京大学学报*(*哲学社会科学版*)*, (2):68–76.

马蹄疾. 1986. 水浒书录. 上海古籍出版社, 上海.

茆诗松, 周纪芗. 2000. 概率论与数理统计. 中国统计出版社, 北京.

孟繁仁. 1990. "许贯忠"是罗贯中的虚象. 晋阳学刊, (4):21–29.

欧阳健. 1985. 《三遂平妖传》原本考辨. 中华文史论丛, (3):149–165.

欧阳健. 2003. 罗贯中研究三题. 东南大学学报*(*哲学社会科学版*)*, 5(5):104–111.

商韬, 陈年希. 1986. 用《三遂平妖传》不能说明《水浒传》的著者和原本问题——与罗尔纲先生商榷. 学术月刊, (2):55–59.

孙楷第. 1958. 日本东京所见小说书目. 人民文学出版社, 北京.

谭红. 2013. 《三遂平妖传》世德堂本与嘉会堂本比较研究. 山东师范大学, 山东.

王晓家. 1998. 《水浒传》作者考论. 陕西人民出版社, 西安.

温庆新. 2018. 文献传播学视野下的《水浒传》作者研究. 中国文化研究, (2):123–132.

宣啸东. 1991. 许贯忠之原型即罗贯中辩. 晋阳学刊, (3):66–69.

杨林. 1999. 罗贯中散论. 海南师范大学学报*(*社会科学版*)*, (3):15–22.

多轮对话的篇章级抽象语义表示标注体系研究*

黄彤[1]，李斌[1]，闫培艺[1]，计婷婷[1]，曲维光[2]

1.南京师范大学　文学院，江苏　南京
2.南京师范大学　计算机科学与技术学院，江苏　南京
huangtong_njnu@126.com, libin.njnu@gmail.com, ypyheta@gmail.com
ting@163.com, wgqu@njnu.edu.cn

摘要

对话分析是智能客服、聊天机器人等自然语言对话应用的基础课题，而对话存在大量情感短语、省略、语序颠倒等现象，对句法和语义分析器的影响较大，对话自动分析的准确率相对书面语料一直不高。其主要原因在于缺乏严整的多轮对话形式化描写方式，不利于后续的分析计算。因此本文在梳理国内外针对对话的标注体系和语料库的基础上，提出了基于抽象语义表示的篇章级多轮对话标注体系，探讨了篇章级的语义结构标注方法，给出词语和概念关系的对齐方案，为称谓语和情感短语增加了相应的语义关系和概念，调整了表示主观情感词语的论元结构，并规定了对话中一些特殊现象，设计了人工标注平台，为大规模的多轮对话语料库标注与计算研究奠定基础。

关键词： 抽象语义表示；多轮对话；标注体系；语义计算；标注体系

Research on Discourse-level Abstract Meaning Representation Annotation framework in Multi-round Dialogue

Huang Tong[1], Li Bin[1], Yan Peiyi[1], Ji Tingting[1], Qu Weiguang[2]

1.School of Chinese Languageand Literature, Nanjing Normal University
Nanjing, Jiangsu, China
2.School of Computer Scienceand Technology, Nanjing Normal University
Nanjing, Jiangsu, China
huangtong_njnu@126.com, libin.njnu@gmail.com, ypyheta@gmail.com
ting@163.com, wgqu@njnu.edu.cn

Abstract

Dialogue analysis is the basic topic of natural language dialogue applications such as intelligent customer service and chat robots. The dialogue corpus is quite different from the regular written corpus. There are a number of complex phenomena such as vocatives, emotional phrases, omissions, word order reversal, redundancy, etc. Compared with the semantic analysis, the accuracy of automatic dialogue parser has been relatively low compared to the written corpus. The main reason is that the lack of rigorous formal description of multiple rounds of dialogue is not conducive to subsequent analysis and quantitative research. Therefore, we make a survey onthe tagging system and corpus for dialogues, then proposes a discourse-level multi-round dialogue tagging system based on abstract meaning representation. It specifically discusses the discourse-level semantic structure annotation method, gives the alignment scheme of

基金项目：国家社科基金项目(18BYY127)、国家自然科学基金(61772278)、江苏省高校哲学社会科学优秀创新团队建设项目。

words and concept relations, adds corresponding semantic relations and concepts for appellations and emotion phrases, adjusts the argument structure of subjective emotion words, and some special phenomena in the dialogue are stipulated, and a manual annotation platform is designed to lay the foundation for large-scale multi-round dialogue corpus annotation and quantitative research.

Keywords: abstract meaning representation , multi-round dialogue , annotation scheme , semantic computing , chinese information processing

1 引言

近年来，伴随着人工智能的浪潮，问答系统、智能助手、聊天机器人等成为了研究的热门，人们希望机器能够像人一样思考，与人类对话，这就要求机器要能够理解、处理人的对话内容，对话分析是自然语言对话应用的基础，口语对话的分析逐渐受到重视 (宗成庆,1999)。

但就目前而言，多轮对话的篇章分析仍存在问题：首先，目前对话语义分析往往以处理普通文本的方式分析对话，导致自动分析效果较差。语义分析大多仍处在相对规范的书面文本的层面上，口语对话不同于书面语，对话中存在更多省略、语法不规范等现象，并且分析口语对话不再局限于单句的分析，需要考虑上下文的信息，这些都给机器自动分析对话增加了难度。Adams (2017) 尝试使用不同模型对对话语料进行依存解析，得到的F值仅有85.7%和80.3%（带依存关系的评测方式LAS），而常规语料均能达到90%以上，存在一定差距，对话解析的效果不甚如意。其次，多轮对话缺乏整体的篇章表示体系和语料建设。目前的语料库资源大多是以书面语料为主，专门针对对话的语料较少，而面向对话的语料库和语料标注规范的研究主要集中在对话行为、篇章关系等特定领域，一般只标注说话人、话轮信息、词性标注或句法分析结构，而忽视话轮间应答关系、话轮内部小句的关系，以及省略恢复、指代消解等难题。对话在篇章层面上的语义结构、应答逻辑没有得到有效的研究和表述。因此需要高质量的口语对话资源以推动语义理解模型的发展 (郑桂东, 2018)。

本文提出了一种针对对话的语义表示方法——对话抽象语义表示（Dialogue Abstract Meaning Representation，DAMR），来解决篇章级多轮对话的语义表示问题。这个方法基于中文抽象语义表示（CAMR）改进而来。抽象语义表示（AMR）作为一种新兴的句子语义表示方法，采用单根有向无环图来表示句子的语义结构 (Bonial et al., 2013)，能够有效解决句子中的论元共享、省略、冗余、语序错乱等难题，并进行了多语言的理论和计算实践 (Oepen et al., 2019)，标注了上万句英文语料[1]和汉语语料[2]。不过，AMR 虽然已经能较好地表示句子语义，但由于对话语料和常规书面语存在较大差异，例如省略（省略主语、宾语）、独立的称呼语、情感短语（如“哈哈哈”）、冗余等，且目前CAMR仅针对单句进行了标注，而对话标注必然是篇章级别的，因此不能直接套用CAMR的规范来表示中文对话的语义，需要根据对话特点对CAMR的框架和规范进行调整、改进和扩充，使之能够表示多轮的对话语料。

因此，我们提出DAMR（Dialogue Abstract Meaning Representation，对话抽象语义表示）继承了CAMR的框架和理论，DAMR是一种针对中文对话的篇章级句子语义表示方法，DAMR从4个方面进行了改进：（1）改进概念关系对齐的语法，将篇章信息其融合到语料标注中；（2）针对对话特点，增加概念标签和关系标签；（3）调整了部分词语的论元结构；（4）对一些对话中的称呼语、情感短语特殊现象进行了规定。

全文结构如下：第2节总结了国内外对话语料的标注体系和方法，第3节介绍了数据来源和AMR标注体系，第4节介绍了DAMR针对对话做出的改进，第5节是结论和未来工作。

2 相关工作

专门针对对话的标注体系和语料库较少，由于主要面向智能机器人，因此多为限定领域（例如旅游行程制定、地图导航、智能音箱）的标注，且标注重点在于对话行为（反应说话者的意图、话语的结构）、篇章关系（句子之间的关系）等，而完整的对话标注体系还需要包括对话的基本信息（说话人编号、话轮等）、指代信息（共指和回指）、句法语义信息（词类、句法结构、语义）等。表1给出了下文提到的对话语料库的标注内容。

[1]https://catalog.ldc.upenn.edu/LDC2020T02
[2]https://catalog.ldc.upenn.edu/LDC2019T07

语料库	对话基本信息	指代信息	篇章结构	句法语义信息	对话行为
LUNA(2007)		✓		✓	✓
MATE(1999)		✓		✓	✓
Martinez(2002)	✓			✓	✓
ISO24617-2(2010)	✓				✓
Zhou(2010)	✓				✓
周小强(2018)	✓		✓		✓

表 1. 语料库标注信息

2.1 对话行为信息标注及语料

多层对话行为标注（Dialogue Act Markup in Several Layers，DAMSL）是应用最为广泛的一个面向任务的通用领域的标注体系，DAMSL在四个维度上对对话行为进行标注，包括：交际状态（Communicative Status）记录话语是否完整，信息层面（Information Level）标注话语的特征，向前功能（the Forward Looking Function）记录当前话语与之后话语的联系、向后功能（the Backward Looking Forward）记录当前话语与之前话语的联系 (ALLEN, 1994)。在DAMSL 提出之后，一部分学者使用DAMSL 标注体系对语料库进行标注，其中最为出名的是Switchboard（SWBD）电话语料库，其目的是进一步提高自动语音识别的语言模型 (Jurafsky et al., 1997)。MRDA （Meeting Recorder Project）对话标注体系则是在SWEB-DAMSL的基础上修改的标注体系，用于标注ICSI（International Computer Science Institute）项目的英语会议多人对话内容，形成了ISCI-MRDA 语料库 (方称宇，2013)。

Bunt (2010)认为DAMSL的维度存在模糊性，提出了一种新的体系DIT++。DIT++细分若十个维度，如活动行为、交际管理、话轮管理等，并规定了每个维度下的交际功能，设计了两类标签：通用目的功能（general-purpose functions）和特定维度功能（dimension-specific functions）标记集，两个标记集下又分多层多个标签。DIT++体系已应用于多个语料库中，如DIAMOND人人对话库、面向任务的AMI人人对话库等 (方称宇，2013)。随着对话行为标注体系的不断发展，Bunt (2010)等人根据以DAMSL和DIT++等多个对话行为标注体系的特点，集各家所长提出多维度的对话行为标注国际标准：ISO24617-2，借鉴DIT++设定了九大维度，包括任务、自我反馈、启他反馈、话轮管理、时间管理、社会义务管理、自我交际管理、语篇构建等，各个维度下设计了相应的对话行为标签。除了通用领域的对话行为标注，还有部分针对特殊领域的标注语料库，如美国基于查询铁路交通的人机对话语料——TRIANS (Allen and Core, 1997)、查找路线的人人对话语料——英国HCRC语料 (HCRC group,1996)。这些标注体系都根据各自的语料特点设定了限于该领域的相应的标签。

汉语对话行为标注随着国外DA的发展，也开始受到重视，但对此的研究仍然有限。王珊等 (2016)建立了一个电视台访谈节目语料库，基于国外对话行为的研究，通过对语料库中的问答句子的分析，设计了汉语的单层级的对话行为的类别。周强 (2017)基于国外DAMSL、SWBD-DAMSL等标注体系，设计了五大标记集，各个标记集下面再分不同标记，并借鉴了ISO标准中的维度设计。

我们认为，在同一句话中对话行为可能包括多个，而说话人的意图有时也无法体现，对话行为也体现不出来，同时，AMR可根据原有的语义关系标签根据语义表示相应的意图或语用功能，例如语气“mode”标签可以表示说话人“祈使”、“询问”等意图，因此DAMR暂时不引入对话行为的标签，更注重使用原有体系表达说话者的实际语义。

2.2 篇章关系信息标注及语料

对话中篇章关系标注主要沿用宾州篇章树库（Penn Discouse TreeBank，PDTB）、修辞结构篇章树库（Rhetorical Structure Theory Discourse Treebank，RST）两大体系。

PDTB仅考虑相毗邻的句子之间的关系，借鉴了谓词论元结构，以连接词（connective）为核心分别定义了两个论元arg1和arg2，连接关系包括显性关系（Explicit）、隐性关系（Inplicit）、替代关系（AltLex）、实体关系（EntRel）、无关系（NoRel），如果没有显性的连接词，标注人员要根据自己的判断表示出其连接关系，同时设定了多层多类语义关系标

签 (PDTB-Group, 2009)。Sara (2010)等人将PDTB体系用于LUNA口语对话语料库中，针对对话语料的特征对意义标签进行了调整。Xue等 (2016)也同样将PDTB体系的用于标注SMS短信息对话，根据信息对话的特点对标签进行增删。

修辞结构理论（RST）将篇章关系称为修辞（rhetorical）关系，设定了两种修辞关系：单核心和多核心，修辞关系所连接的篇章单位如果存在主次区别，那么就是单核心关系，反之就是多核心关系。RST 与PDTB最大区别在于其篇章结构树有层次，每个修辞关系都可以连接两个或多个篇章单位，这些篇章单位又可以组成大的篇章单位和其他篇章单位形成修辞关系，最终一个篇章形成一个有层次的篇章结构树 (Carlson et al., 2001)。Stent (2000)首次将RST 用于标注面向任务的对话语料中，针对对话或是标注领域特有的特点，新增了一些修辞关系（如问答关系），并为某些范围过于广泛的标签设置了更具体的下级标签。

中文AMR中规定了10种篇章关系，我们将沿用这些关系来标注对话中的篇章关系，因为对话话题较为分散，因此存在篇章关系的两个或多个句子不仅局限于相毗邻的两个句子中，同时也会根据对话的实际特点增加相应的篇章关系标签。

2.3 综合信息标注语料库

综合标注的对话语料库指标注了多种信息的语料库，包括上文提到的对话行为、篇章结构，还有语义信息、共指等信息。

LUNA语料库是一个跨语言（意大利语、波兰语、法语）、跨领域的人人、人机对话语料库，采用了层标注，第一层为语义标注，第二层为领域属性标注，以及非必须的其他层，包括谓词结构框架、对话行为、指代信息等 (Raymond, 2007)。如图1，领域属性标注层标注句子中的语义块所属的领域及其属性，以“属性-值”对构成，语义块来自第一层的语义标注；谓词结构框架借鉴了FrameNet 框架标注语义结构，为预先设定好的领域设定框架；再填入相应的元素。对话行为沿用DAMSL 体系标签；同时，LUNA 语料库标注了共指信息，将可标记共指的元素标记为given或new，如果标为given，则找出最近发生的对象并增加指针指向它。

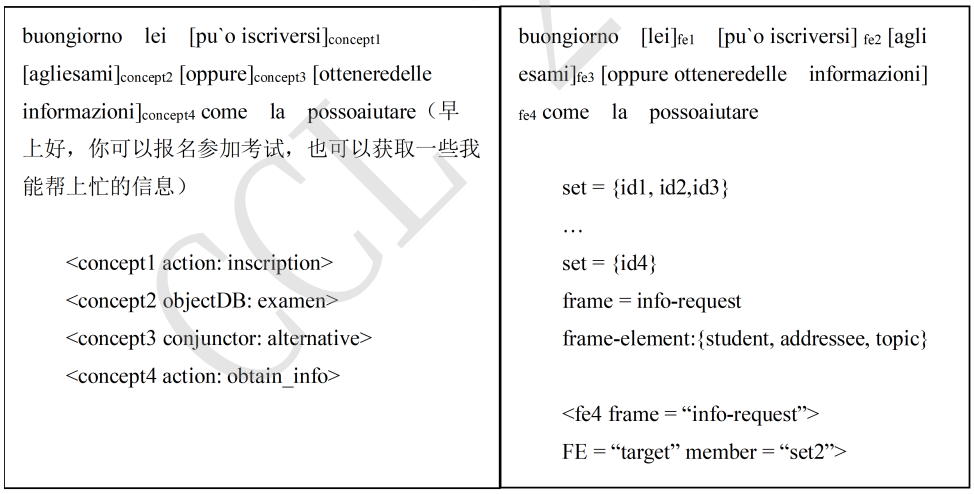


图 1. LUNA标注方法示例

其他还有较为知名的语料库还有Martinez (2000)在铁路信息系统的对话语料上标注了三层标签，分别为对话行为、框架（Frames）和实例（case），对话行为基于TRAINS体系的标签进行了调整；框架借用FrameNet的思想，为具体任务设置相应框架；实例则用来填充框架的槽；MATE 语料库标注了语义信息、对话行为、共指信息 (Poesio et al.,1999)；Zhou (2010)建立了一个汉语的旅游领域的语料库，共标注了十三层信息，包括话轮、主题、说话者信息、分词词性信息、拼音、语音转录、语音边界、句子重音、音量、非语言信息、基于ICSI-MARA体系的对话行为、形式错误信息、情绪。

LUNA、MATE、Martinez建立的语料库都是面向任务的语料库和标注方法，因此其意义标注仍是从对话行为出发，更注重抽取出说话者所要实现的功能意图，再根据意图设定论元结构，无法完整地表示句子的语义。在指代标注上，LUNA等其他标注共指的语料库都只涉及名词，将上下文中共指的元素用同样的id连接依赖，忽略了指向一个完整事件的代词，因此不利

于判断指示词和先行语之间的关系和指代消解的实现。另外，这些语料库的重点仍然是单句的语义标注，没有将有相应问答或其他对应关系的句子表示出来。

周小强 (2017)等人设计了一个交互式问答语料的关系结构标注体系。除标注了对话行为类别外，还标注了问答中的语义匹配关系和语义补充关系。其对应关系只限于句子和句子之间的关系，但在实际语料中情况更为复杂，有补充和匹配关系的不一定为整个句子，可能只是句子中的一部分，因此这种方法存在问答点对应不明确的问题。

3 数据来源及AMR体系介绍

3.1 数据来源

我们在改进的中文抽象语义表示标注平台上试标部分中文短信息SMS对话语料[3]以分析对话标注存在的问题。该语料总共有15000篇对话，我们从中选取了10篇对话、475个句子进行试标注，语料信息包括话语编号、说话人编号、时间信息。我们对其进行预处理，增加了话轮编号信息。选其作试标语料主要因为：短信对话保留了日常对话的基本要素和特征，同时避免了肢体语言或现实环境语境对录音转写语料内容的影响。

3.2 AMR体系

抽象语义表示（Abstract Meaning Representation, AMR）是一种新兴的语义表示方法，它用单根有向无环图来表示句子语义，将句子中的实词抽象为概念节点，实词之间的关系则抽象为带有非核心语义关系标签的有向弧，忽略了虚词和一些较虚的语义（冠词、时态、单复数），允许增加、删除或修改概念 (Bonial et al., 2013)。

在这个基础上，O'Gorman (2018)等人提出了标注多句AMR（Multi-sentence AMR, MS-AMR），即将AMR拓展到篇章层面，但只关注了篇章中的共指现象，标注了名词、动词、代词、隐形角色的共指关系，MS-AMR设定了三种共指关系：一致关系、部分-整体关系、成员-集合关系。Bonial等人 (2020)针对人机对话语料对AMR 进行了改进，构建了对话AMR体系，主要有以下几点扩充：在AMR的最上层设定了36个对话行为标签；增加了时、体标签；针对该人机对话语料的用途设定了空间参数。

李斌 (2017)在AMR体系的基础上提出融合概念对齐的一体化标注方案，针对汉语特有现象进行了改进，形成了中文抽象语义表示方法（Chinese Abstract Meaning Representation）。CAMR的改进如下：为量词、时、体新增了语义关系标签；还原重叠式，如"试试"还原为"试"；组合离合式，如把"睡一会觉"合成概念"睡觉"；为复句关系增加了关系概念标签。

使用CAMR能够更完整、合理地标注对话：

第一，AMR关注的并非句子中的具体词语，而是句中抽象的概念和关系，允许增加、删除或修改概念，利用这个特点，可以在一定程度上解决对话中的倒序、冗余等情况，也可以对对话中省略的概念进行恢复（如图2中的"整治"），将话语中的语义合理地表示出来。

第二，CAMR进行了对齐改进，采用句中的序列进行编号，实现了概念与句中单词的对齐，有利于合理地表示指代、省略等情况，也有助于照应语和先行语之间关系的标注。

第三，CAMR为复句关系添加了并列、因果、让步、条件、转折、解释说明、选择、目的、递进、时序等10个标签，如图2中的并列复句关系"and"。DAMR可用这10 个复句关系标签表示对话篇章结构关系。

第四，CAMR新增的dcopy和refer用来标注两个概念之间的关系，有助于省略和指代照应的标注。

但CAMR标注对话存在一些问题，比如目前CAMR仅标注单句，复句关系仅限于同一句中的标注，一些在句中没有充当明确成分的词语无法标注（称呼，叹词)等。因此，我们在CAMR的基础上进行了改进，具体标注方法在第4节中说明。

4 对话AMR标注体系

我们在改进版的中文抽象语义表示标注平台上试标注了500句中文短信息SMS对话语料，尽可能在现有CAMR体系的基础上标注，同时根据标注对话遇到的问题对其进行调整，以期扩充CAMR的兼容性。对话中会有一些特有的成分，例如称呼、表示情绪的成分，存在指代照应

[3]https://catalog.ldc.upenn.edu/LDC2016T13

运河[1] 的[2] 整治[3] 改善[4] 了[5] 该[6] 县[7] 的[8] 投资[9] 环境[10]，[11] 吸引[12] 了[13] 外商[14] 投资[15] 。[16]

```
x18/and
  :op1() x4/改善-01
      :arg0( ) x3/整治-01
          :arg1(x2/的) x1/运河
      :aspect( ) x5/了
      :arg1( ) x10/环境
          :mod( ) x9/投资
          :poss(x8/的) x7/县
              :mod( ) x6/该
  :op2( ) x12/吸引-01
      :arg0( ) x3/整治-01
      :arg1( ) x15/投资-01
          :arg0( ) x14/外商
      :aspect( ) x13/了
```

图 2. CAMR示例

的距离较远的现象，话轮间问答不直接对应，出现大量的省略，诸如说话人/听话人人称代词的省略。因此针对对话的特点在以下几点对CAMR做出了改进：实现双层标签概念对齐、增加若干个标签、修改部分词语的论元结构、并规定了一些对话中的特殊现象的标注。

4.1 概念对齐

DAMR的每个句子都包含以下字段：语篇编号、话轮编号、句子编号、说话人编号、问答位置信息（见表2）。

语篇编号	话轮编号	句子编号	说话人编号	句子	问答位置信息
3	48	83	151460	过[1] 一会[2] 和[3] 你[4] 说[5]	-
3	49	84	131525	好[1]	-

表 2. DAMR语料字段

前5个字段根据语料顺序自动分配，问答位置信息由人工标注，标注答句所对应的问句的位置信息，如果非问答对应则不标注(本文其他例句如不涉及问答则省略该字段)。

CAMR的概念标签采用xn的形式，n是根据输入的已分词的原始句子序列分配的有序编号。人工补充的概念则由标注系统分配随机编号。目前的CAMR的编号仅适用于单句，无法跨句子进行标注，因此为了实现篇章级别的标注，DAMR 采用了双层编号，即用sn_xn来对齐句中的概念，sn根据输入的句子序列分配，xn 则仍旧根据词语在句中的序列分配。样例见图3。

```
s83_x5/说-01
   :arg3(x3) x4/你
      :time() x1/过-01
            :arg1() x2/一会

s84_x1/confirm
   :arg1() s83_x5/说
```

```
x5/说-01
   :arg3(x3) x4/你
      :time() x1/过-01
            :arg1() x2/一会

x1/好-01
   :arg0() x3/说
```

图 3. DAMR/CAMR概念对齐示例

为减轻标注人员的操作量，当句子只出现当前句子的概念，则仅使用xn标签，当出现其他句子的概念时，才完整表示sn_xn。

4.2 新增标签

DAMR沿用了CAMR的5个核心语义关系标签、44个非核心语义关系标签和109个专名概念。argx （x∈[0,4]）表示核心语义角色关系，每个谓词的每个义项都有自己的核心语义角色框架。非核心语义关系是指核心语义关系之外的语义角色关系，CAMR规定了在AMR的基础上规定了目的、处所、时间等44种对所有谓词通用的非核心语义关系。根据对话语料特点，DAMR新增了4个概念标签和2个非核心语义关系标签以兼容对话中会出现的语义关系。

语篇编号	话轮编号	句子编号	说话人编号	句子
16	400	709	131525	太[1] 土[2] 了[3]

s709_x5/speak
:arg0() x4/speaker
:arg2() x6/hearer
:arg1() x2/土-01
:degree() x1/太
:aspect() x3/了

图 4. speak概念

4.2.1 说话speak

DAMR为对话新增了speak、speaker和hearer概念，对话中的每一个句子的根节点都为概念speak，概念speak规定了三个论元，分别为：arg0: speaker（说话人）；arg1: thing speak（说话内容）；arg2: hearer（听话人）。说话人speaker和听话人hearer 为新增概念，标注时需根据实际语义标注出话语的说话人和听话人。如图4。

本文其他例句会省略speak、speaker、hearer概念以使例子更清晰。

语篇编号	话轮编号	句子编号	说话人编号	句子
2	85	144	135882	555[1] ，[2] 复习[3] 好[4] 痛苦[5]
2	85	145	135882	……
2	86	146	138459	嗯[1]

s144_x5/痛苦-01
:arg0() x6/speaker
:arg1() x3/复习
:degree() x4/好

s146_x1/confirm
:arg1() s144_x5/痛苦

图 5. confirm概念

4.2.2 肯定confirm

对话是交互的，听话人会对说话人的话语表达态度，最常见的是对上一句的肯定（是的、嗯嗯等）,针对这种情况，DAMR新增了一个confirm概念。具体示例如图5，句146 的根节点是肯定概念confirm，“嗯”是对“复习好痛苦”的肯定，标注时将“嗯”抽象为概念“confirm”，不再单独表示出来。

4.2.3 情感:feeling

对话中说话人会用多种形式表达自己的心情，如“哈哈”“呵呵”“呜呜”等，以及在线上对话文本中会出现的表情包，甚至是单纯的标点符号，如“。。。”、“...”，DAMR新增了非核心语义关系标签“feeling”来表示这种语义。

如图5所示，“555”表示说话人的心情，将其置于根节点“痛苦”的下层。由于心情的表示形式太过复杂，例如“呵呵”可表示偏正向的愉悦情绪，而现在网络上的新兴用法也将“呵呵”表负面情绪，因此为了避免标注的不统一，目前DAMR只使用“feeling”标签，不区分具体的情绪类别。另外，一部分表情并非表示心情，而是表示“再见”、“你好”等概念，对于这部分表情，DAMR要求对其进行语义转写，将其真正语义表示出来。

4.2.4 称呼:naming

存在称呼语是对话中突出的特点，称呼本身不存在于谓词概念的论元结构中，为了更好地表示对话中的称呼现象，DAMR引入非核心语义关系标签“naming”。注意与原标签“name”区分，称呼并不等同于实际名字“name”，称呼是动态的，“name”则是静态的。

如图6所示，根节点speak支配的论元分别为arg0说话人、arg1说话内容、arg2听话人，因为说话人称呼听话人为“王老师”，因此naming在arg2节点的下层。如果称呼就是听话人的名字，则“naming”和“name”同时出现，都在arg2节点的下层。

语篇编号	话轮编号	句子编号	说话人编号	句子
20	549	967	151461	王老师[1] 你[2] 是否[3] 可以[4] 给[5] 出[6] 更[7] 多[8] 的[9] proposition[10]

s967_x12/speak
:arg0() x12/speaker
:arg2() x4/hearer
:naming() x1/王老师
:arg1() x4/可以-01
:arg0() x5/给-01
:arg0() x2/你
:arg1() x10/proposition
:quant(x9/的) x8/多
:degree() x7/更
:direction() x6/出
:mode() x3/interrogative

图 6. naming关系标签

语篇编号	话轮编号	句子编号	说话人编号	句子
20	567	997	138158	今天[2] 的[3] 电影[4] 好看[5] 么[6] ？[7]

x5/好看-01
:arg0() x1/电影
:time(x3/的) x2/今天
:argh() x4/hearer
:mode() x6_x7/interrogative

图 7. ”好看”的论元结构

4.3 论元结构

对话中的话语常常带有说话人或者听话人的主观态度，在CAMR中没有将这种态度表示出来，因此DAMR 对一部分谓词的论元结构进行了修改，增加了一个论元：argh （态度持有人）。目前修改的谓词是含有主观态度的形容词或短语，如“好看”“丑”“很有主意”，这部分词语原来只有一个论元：arg0（entity describe）。

如上图所示，“好看”除了arg0（电影）外，还增加了态度持有人argh，根据语义，“好看”的态度持有者是听话人hearer。

4.4 对话中的特殊现象

4.4.1 问答句的对应

问答是对话中常见的形式，但是在对话中说话人可能同时进行两个或多个话题，问句和答句不一定为相邻句，问答的语义对应较为分散，并且答句不一定正面回答问句，为体现问句和答句之间的语义联系，DAMR在每个句子上增加一个字段，可以将答句与问句联系起来。

语篇编号	话轮编号	句子编号	说话人编号	句子	**问答位置信息**
4	149	257	138375	那[1] 个[2] socio[3] 难[4] 不[5] 难[6] 整[7] ？[8] ？[9] ？[10]	
4	149	258	138375	嗯[1] 是[2] 的[3] 小[4] 野兽[5] 最近[6] 也[7] 满[8] 虚弱[9] 的[10]	
4	150	259	138194	哎[1] 我[2] 觉得[3] 我[4] 重新[5] 上[6] 了[7] —[8] 次[9] socio[10] 似的[11]	**s257_x12**

表 3. DAMR语料字段

```
s257_x12/or
    :op1() x4/难-01
        :arg0() x7/整-01
            :arg1() x3/socio
                :mod() x1/那
                :cunit() x2/个
    :op2() x6/难-01
        :polarity() x5/-
        :arg0() x7/整-01
    :mode() x7_x8_x9/interrogative
```

图 8. 疑问句

问句位置信息有两个维度。第一个维度为答句所对应的问句编号，第二个维度为所对应问句的根节点。如上图，问句257的根节点为x12，因此答句259 的问句位置信息为s257_x12。

4.4.2 问句的省略

由于对话双方处于同一个语境中，完成对话理解所需的背景知识是两者共享的，因此对话中的一方提问常常会省略很多成分，在标注时，需要根据句子实际语义将省略的问句成分表示出来。如图9，句2仅用一个问号表示说话者的疑问，其完整语义为：为什么说atac疯了，在标注时需将实际语义标注出来。

4.4.3 人称省略

在对话中说话人常常会省略自己或听话人的人称，标注时要将省略的说话人或听话人补出来。如图10，“弄完”的施事为听话人，在标注时我们将省略的hearer补充出来。

4.5 小结

我们改进了原CAMR标注平台，加入了篇章对话信息（语篇编号、话轮编号、说话人编号），通过对476 句对话语料的标注，针对对话特点新增了标签，处理了称呼语、情感短语等对话特有现象，规定了省略、话轮间应答关系的标注，使CAMR 体系从单句拓展到篇章级别。

语篇编号	话轮编号	句子编号	说话人编号	句子
1	1	1	151430	atac[1] 疯[2] 了[3]
1	2	2	131525	？[1]

s1_x3/疯-01
:arg0() x1/atca
:aspect() x3/了

s2_x2/amr-unknown
:cause-of() s1_x3/疯
:mode() x1/interrogative

图 9. 省略问句

语篇编号	话轮编号	句子编号	说话人编号	句子
7	143	232	131525	嗯[1] 不过[2] 先[3] 把[4] 小说[5] 弄完[6] 吧[7]

s232_x18/contrast
:arg1() x10/confirm
:arg1() s231_x16/causation
:arg2(x6/不过) x10/弄完-01
:arg0() x24/hearer
:arg1(x4/把) x5/小说
:time() x3/先
:mode() x7/imperative

图 10. 人称省略

5 结论及未来工作

近年来对话系统的发展越来越受到重视，对话语义的形式化表示的作用愈发凸显，但国内目前还没有较完整的标注体系表示对话的语义。本文梳理了国内外对话标注体系和语料库的发展，在CAMR体系的基础上进行改进扩充：实现跨单句层面的概念对齐，新增适用于对话语料的概念标签和非核心语义关系标签，修改词语的论元结构，规定了问答句对应、省略等对话中特有现象的标注，形成了对话标注体系DAMR。这些改进有利于解决对话中的省略和跨句子指代等问题，使问答点的对应更明确，更完整地表达说话者的语义，对合理表示对话语义，为对话的自动理解与分析有较大价值。

在今后的工作中，第一，我们将加强对对话语义特点的研究，尝试标注语音对话转写语料，针对实际对话特点和新出现的问题完善DAMR标注体系，使之能够适用各个领域的对话语料，以验证DAMR的效果；第二，使用DAMR标注体系标注语料，构建一个大规模的对话AMR语料库，并进行统计分析。第三，我们希望通过对对话的标注语料库进行学习，提高对话自动分析的效果。

参考文献

Allison Adams. 2000. Dependency Parsing and Dialogue Systems. UPPSALA University.

Amanda Stent. 2000. Rhetorical Structure in Dialog. *Proceedings of the First International Conference on Natural Language Generation*, 247–252.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, Nathan Schneider. 2013. Abstract Meaning Representation for Sembanking *Linguistic Annotation Workshop and Interoperability with Discourse*, 178–186.

Carlos D.Martinez. 2002. A Labelling Proposal to Annotate Dialogues. *Proceedings of the Third International Conference on Language Resources and Evaluation(LREC)*, 1577–1582.

Christian Raymond. 2007. The LUNA Corpus: an Annotation Scheme for a Multi-domain Multi Lingual Dialogue Corpus. *Proceedings of the 11th Workshop on the Semantics and Pragmatics of Dialogue*, 185–186.

Claire N. Bonial, Lucia Donatelli, Jessica Ervin, Clare R. Voss. 2019. Abstract Meaning Representation for Human-Robot Dialogue. *Proceedings of the Society for Computation in Linguistics*.

Claire Bonial, Lucia Donatelli, Mitchell Abrams, Stephanie M. Lukin, Stephen Tratz, Matthew Marge, Ron Artstein, David Traum, Clare Voss. 2020. Dialogue-AMR: Abstract Meaning Representation for Dialogue. *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, 684–695.

Harry Bunt. 2010. The DIT++ Taxonomy for Functional Dialogue Markup. *Proceeding of 8th Int. Conf. on Automous Agents and Multiagent Systems(AAMAS,2009)*, 13–24.

Harry Bunt, Jan Alexandersson, Jean Carletta, Jae-Woong Choe, Alex Chengyu Fang, Koiti Hasida, Kiyong Lee, Volha Petukhova, Andrei Popescu-Belis, Laurent Romary, Claudia Soria, David Traum. 2010. Towards an ISO Standard for Dialogue Act Annotation. *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*.. 2548–2555.

HCRC group. 1996. *HCRC Dialogue Structure Coding Manual*. University of Edinburgh.

James Allen, Mark Core. 1997. *Draft of DAMSL: Dialog Act Markup in Several Layers*. Lancaster University.

James Allen, Peter Heeman. 1994. *TRAINS Spoken Dialog Corpus*. University of Rochester.

Jurafsky D, Shriberg E, Biasca D. 1997. *Switchboard SWBD-DAMSL Shallow-Discourse-Function Annotation Coders Manual*.

Keyan Zhou, Aijun Li, Zhigang Yin, Chengqing Zong. 2010. CASIA-CASSIL: a Chinese Telephone Conversation Corpus in Real Scenarios with Multi-leveled Annotation *Proceedings of the International Conference on Language Resources and Evaluation(LREC 2010)*. 2407–2413.

Lynn Carlson, Daniel Marcu, Mary Ellen Okurovsky. 2001. Building a Discourse-Tagged Corpus in the Framework of Rhetorical Structure Theory *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue*.

M. POESIO, F. Bruneseaux, L. Romary. 1999. The MATE Meta-scheme for Coreference in Dialogues in Multiple Languages. *ACL Workshop Towards Standards and Toolos for Discourse Tagging*, 65–74.

Nianwen Xue, Qishen Su, Sooyoung Jeong. 2016. Annotating the Discourse and Dialogue Structure of SMS Message Conversations. *Proceedings of LAW X –The 10th Linguistic Annotation Workshop*. 180–-187.

Tim O'Gorman, Michael Regan, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Martha Palmer. 2018. AMR Beyond the Sentence: the Multi-sentence AMR corpus. *Proceedings of the 27th International Conference on Computational Linguistics*. 3693–-3702.

PDTB-Group. 2009. *The Penn Discourse Treebank 2.0 Annotation Manual*. University of Pennsylvania.

Sara Tonelli, Giuseppe Riccardi, Rashmi Prasad. 2016. Annotation of Discourse Relations for Conversational Spoken Dialogs. *Proceedings of International Conference on Language Resources and Evaluation(LREC2010)*. 2084–2090.

Stephan Oepen, Omri Abend, Jan Hajic, Daniel Hershcovich, Marco Kuhlmann, Tim O'Gorman, Nianwen Xue, Jayeol Chun, Milan Straka, Zdenka Uresova. 2016. MRP 2019: Cross-Framework Meaning Representation Parsing. *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*. 1–27.

方称宇, 曹竟, 刘晓月. 2013. 基于语料库的最新ISO会话行为标注体系的研究：从SWBD-DAMSL到SWBD-ISO. 当代语言学. 15(4): 439–458.

李斌, 闻媛, 宋丽，卜丽君，曲维光，薛念文. 2017. 融合概念对齐信息的中文AMR语料库的构建. 中文信息学报. 31(6):93–102.

王珊, 刘锐. 2016. 谈话节目语料库的构建与会话结构分析. 中文信息学报. 30(6): 140–146.

郑桂东. 2018. 多轮对话语料构建中的离群对话分析. 哈尔滨工业大学.

周强. 2017. 汉语日常会话的对话行为分析标注研究. 中文信息学报. 31(06):75–82.

周小强, 王晓龙, 陈清财. 2017. 交互式问答的关系结构体系及标注. 中文信息学报. 32(5): 1–10.

宗成庆，吴华，黄泰翼，徐波. 1999. 限定领域汉语口语对话语料分析. 全国第五届计算机语言联合学术会议. 115–122.

发音属性优化建模及其在偏误检测的应用

郭铭昊
语言资源高精尖创新中心/北京
北京语言大学信息科学学院/北京
gmhgmh8000@163.com

解焱陆
语言资源高精尖创新中心/北京
北京语言大学信息科学学院/北京
xieyanlu@blcu.edu.cn

摘要

近年来，发音属性常常被用于计算机辅助发音训练系统（CAPT）中。本文针对使用发音属性的一些难点，提出了一种建模细颗粒度发音属性（FSA）的方法，并在跨语言属性识别、发音偏误检测中进行测试。最终，我们得到了最优平均识别准确率约为 95%的属性检测器组；在两个二语测试集上的偏误检测，相比基线，基于 FSA 方法均获得了超过 1%的性能提升。此外，我们还根据发音属性的跨语言特性设置了对照实验，并在上述任务中测试和分析。

关键词：发音属性；偏误检测；属性识别

Speech attributes optimization modeling and application in mispronunciation detection

Minghao Guo
Beijing Advanced Innovation Center for Language Resources/Beijing Beijing Language and Culture University/Beijing
gmhgmh8000@163.com

Yanlu Xie
Beijing Advanced Innovation Center for Language Resources/Beijing Beijing Language and Culture University/Beijing
xieyanlu@blcu.edu.cn

Abstract

In recent years, Speech attributes are often used in computer-aided pronunciation training systems (CAPT). This paper proposes a method for modeling fine-grained speech attributes (FSA) for some difficulties in using speech attributes, and tests in cross-language attribute recognition and mispronunciation detection. In the end, we obtained an attribute detector group with an optimal average recognition accuracy rate of about 95%; the mispronunciation detection on the two second language test sets, based on the FSA method achieved a performance improvement of more than 1% compared to the baseline. In addition, according to the cross-language characteristics of speech attribute, we set up a comparative experiment and tested and analyzed in the above tasks.

Keywords： Speech attribute; Mispronunciation detection;Attribute recognition

1 引言

近年来，随着二语学习需求的增长，学习汉语的人越来越多。基于自动语音识别的计算机发音训练系统（The computer-aided pronunciation training system）不仅能够满足当下学习者碎片化学习时间的需要还能弥补传统课堂教学的劣势。它的主要核心功能有：（1）提供反馈；（2）评估发音质量。从反馈形式的角度看，CAPT 系统可大致分为发音质量打分和发音偏误检测两种类型，发音偏误检测任务的目标则是以高精度检测发音错误并给出对应的纠音反馈。研究发现，即使以简单的形式提供纠音反馈，也能够改善学习者在音素层级的发音质量（Neri A, 2006）。用于提供纠音反馈的研究有很多，例如利用拓展识别网络创建一个音素级的发音偏误检测和诊断的模型（Harrison A M, 2009），利用发音属性来提供诊断性反馈等。

通过研究人类识别语音的过程，人的记忆单元中字词存储的基本单位是段，并且通过一系列的特征集合来相互区分，这些用于描述语音学发音并区分语音段的特征称为“区分性特征”。这些特征可以从语音的不同方面定义，如发音位置、发音方式等，而这些“区分性特征”叫做发音属性（Speech Attribute）。目前，发音属性在二语学习中主要用于提供纠音反馈、简化二语语料库标注等，而发音属性的定义方法多采用国际音标标准。

外国学生在学习汉语时出现的发音偏误，往往就是由于发音位置等发音属性的不准确导致的。二语学习者受母语负迁移等作用影响，其发音属性常常会倾向于母语中相似音的发音属性，同样地，如果二语中的发音属性在其母语中缺失，则学习者将很难正确掌握新的发音方法。目前，在偏误检测任务上应用发音属性的方法有：发音偏误趋势建模、发音属性特征提取、多语言发音属性建模等。

Cao 等根据来自于发音人发音位置和发音方法等发音属性的不准确，定义了包括高化、低化、前化、后化等发音偏误趋势（Cao, 2010）。Li 等人基于发音偏误趋势的属性特征提取，用于提供诊断性反馈（Li, 2016）。但是，上述方法也存在很多局限性，例如高度依赖拥有准确标注信息的大规模二语语料库。采用多语言建模发音属性的原因在于，发音属性具备跨语言特性，且当二语者在发音时发生母语负迁移现象，其偏误发音的发音属性会包含两种语言的发音属性。因此，若同时建模两种语言的发音属性，将有助于检测偏误发音的发音属性（Duan, 2017）。理论上，通过多语言发音属性建模，有助于建模任意母语背景二语者语料的发音属性。采用多语言发音属性建模也存在难点，例如：难以建模所有已知语言、汉语与其他语言发音属性定义存在差异（如，汉语元音“i”）。以上应用中，使用发音属性的方法往往采用国际音标的定义，但是由于汉语和其他语言在发音属性的定义上存在差异，国际音标无法准确地描述汉语的发音属性。

假设在没有足够的二语数据集的情况下，本研究针对整合多母语描述发音偏误方法的难点，提出了一个以学习汉语为目的发音属性定义和优化建模方法，即细颗粒度的发音属性（FSA），将有助于改善汉语的发音偏误检测任务。在此基础上，检测属性检测器的跨语言能力，以及探究面对不同母语背景学习者语料时上述方法检测发音偏误的能力。根据发音属性具备可跨语言的特点，我们还探究了单语言训练的属性检测器的跨语言能力，通过控制建模时的上下文信息，降低了单语言属性检测器对汉语数据的过度适应，并设置了多个对照实验分别采用不同的上下文信息建模，在汉语和英语两个测试集中进行属性检测，最后对比双语言属性检测器的检测结果来进行分析。由于跨语言建模发音属性具备描述发音偏误的能力，我们还在母语为日语和俄罗斯语的学习者测试集上，进行次音段级和音段级的发音偏误检测。

2 发音属性的定义

本研究从四个方面对汉语声母进行了描述：发音位置（PA）、发音方式（MA）、是否送气（AS）、清浊音（VO）。而汉语元音部分则包括四个类别:舌位前后（TF）、舌位高低（TH）、唇形圆展（RO）、PA 和 VO。需要强调的是，在声学音标中辅音和元音的发音属性定义不同

（Siniscalchi，2008），因此我们分别对辅音和元音的发音属性进行建模，并尝试将它们在 PA 分类中合并建模。由于所有的汉语元音在 AS 和 VO 中都没有子分类，所以我们将它们的详情放在声母发音属性定义中呈现。

我们将所有的汉语辅音与 IPA 一一映射，根据 IPA 上对应音素的知识信息，找到我们需要的属性信息并给予分类标签。在 PA 类别中，所有元音部分都将被标记为“vowels”，其中声母的四个类别使用映射表 1 的音素分类中产生（C.Zhang, 2011）。在表 1 中，汉语辅音以拼音形式首先列出，其次则是以音素表示的英语辅音。该表还列出了英文中存在但中文中不存在的属性，这些属性没有参与建模，以此不难看出汉语和英语属性的区别。例如，英语中没有 AS 属性分类，以及 Timit 音素集中只有一个清化的元音“axh”。

Attributes		Phone set (Ch/En)	
P A	Bilabial	b p m	p b m w
	Labiodental	f	f v
	Alveolar	d t l n	t l el ch sh jh zh dx nx
	Dental	c s z	s dh en n r z th d
	Retroflex	zh ch sh r	
	Palatal	j q x	y
	Velar	g k h	k g ng
M A	Stop	g p d t g k	t p k b d g
	Fricative	f s sh r x h	sh th f hh dh hv v w zh s z
	Affricate	z zh c ch j q	ch jh
	Nasal	m n	en m nx ng n
	Lateral	l	el l
	Approximant		dx
	Tap or Flap		r y

表 1.中英文辅音属性类别表（部分）

汉语韵母由多个元音和鼻元音组成（en、an 等)，相对于声母来说比较复杂。因此，我们将每个汉语韵母描述为一组 IPA 音素，然后根据这些音素得到每个韵母的属性集。表 2 中列出了四个汉语元音属性类别，列出了汉语单元音和英语音素的属性分类。

此外，汉语和英语的元音在舌位上有很大的差异。在过去的研究中，将音素舌位前后大致分为三大类：前、中、后（MullerM，2017），这样简单的分类显然不能完全体现汉语元音在舌位前后的位置。为了找到更好的描述汉语的舌位前后的分类方法，我们将表示汉语元音分为五类和七类分类建模。由于五分类的舌位可以直接对应于声母的发音位置，所以我们在 PA 类中同时对韵母和声母进行建模，而在 TF 类中更详细地分为七类。将 PA 与 TF 进行比较，可以看出两种分类方法的差异，如表 2 所示。另外，汉语的声母在 TF、RO 和 TH 中被标记为“辅音”。值得注意的是，汉语韵母中存在着三种属性维度，它们描述了汉语韵母中存在的属性数量。例如，汉语的最后一个“iao”被描述为三个 IPA 音素，所以它在每个类别中都有三维属性。

Attributes		Phone set (Ch/En)	
P A	Dental	ii	
	Retroflex	iii	
	Palatal/Front	i v	iy ih ae eh
	PA-Central	a	ax ix ux axh axr er
	Velar/Back	u	aa ah ao uw uh

TH	High	i ii iii v u	ix iy ux uw
	Second H		ih uh
	Half H		
	Middle		axh axr ax
	Half L		ah ao eh er
	Second L		ae
	Low	a	aa
TF	Front 2	ii	
	Front 1	iii	
	Front	i v	ae eh iy
	Half F		ih
	Central	a	axh axr ax er ix ux
	Half B		uh uw
	Back	u	aa ah ao

表 2.中英文辅音属性类别表（部分）

3 基于 FSA 方法的优化建模

3.1 时延神经网络的设计

对连续语流数据下的语音任务来说，由于语音是一种时序序列，上下文信息对于声学模型的性能影响非常关键，在发音偏误检测任务中也是同样。TDNN 其优点在于多层网络训练时对输入特征具有较强的时序建模能力、描述了语音特征在时间序列上的关系、具备时间不变性且不需要对样本标注进行时间定位。适用于本研究的关键在于 TDNN 对动态语音分类任务具有相当好的性能表现（Waibel, 1989）。图 1 所示是本研究训练发音属性时的 TDNN 模型结构，这种 TDNN 结构对时间序列输入数据[10,11,12]具有有限的动态响应。假设 t 是当前帧，在输入层（layer1），帧[t-2，t+2]被拼接在一起。层 2,3 和 4 我们分别将帧[t-1，t+2]，[t-3，t+3]和[t-7，t+2]拼接在一起。总的来说，神经网络的左上下文为 13，右上下文为 9。

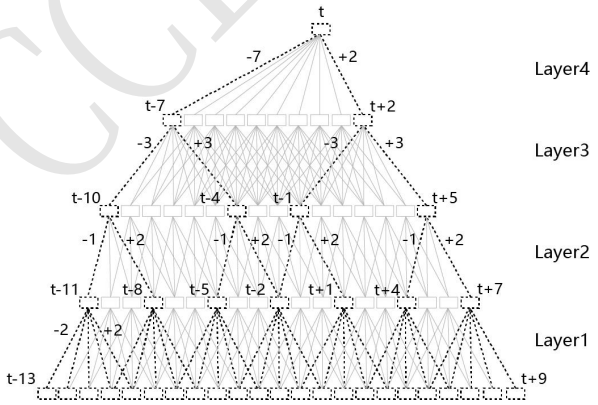


图 1.本研究的 TDNN 网络结构

3.2 I-Vector 特征的提取

我们使用所有训练集特征建立 GMM 建模通用背景模型，得到 GMM 训练的统计量后重新训练 GMM，得到 UBM，其中训练特征 40 维，高斯数 512 个；使用 UBM 初始化 i-vector，获取正规化（CMVN）的特征后验概率，计算统计量，根据统计量计算最后的 i-vector 模型 F，其中 s 维度为 512*40，m 维度 512*40，w 维度是 100，因此 T 维度为 512*100*40；拼接之后使用 CMVN 和 LDA 进行特征处理，根据特征和 UBM 获取每个话者的超向量，根据超向量 s、UBM、

F 模型，得到 i-vector 特征（S.Xue, 2014; M. Karafiat, 2011; N. Dehak, 2010）。最终得到 100 维的 i-Vector 特征，和 49 维的 MFCC 特征共同训练发音属性检测器组。

3.3 优化训练数据不平衡问题

建模时，汉语声母和韵母的建模分离和属性分类差异导致训练数据分布不平衡。例如，声母属性分类器中无用的标签“vowels”包含了近一半的训练数据。我们采用基于音素背景建模（phone-basedbackgroundmodel，PBM）的方法来解决这一问题，其关键是将无用分类和数据量庞大的分类进行多标签表示，就像在说话者或话语验证的方法，通过非属性类划分获得多标签。下图为本研究在属性检测器中使用 PBM 方法建模的示例图，该示例图为非属性类“vowels”化子标签的做法，以建模发音方式（PA）为例，横坐标为属性标签名，纵坐标为属性标签数量，蓝色是原始标签数据量，橘色是 PBM 算法进行数据平衡后的各标签数量。可以看到，蓝色部分“vowels”标签数量远大于其他标签，但是该标签在 PA 中没有任何意义，这样的数据分布会导致模型训练不平衡；而使用 PBM 后的橘色部分，将原标签“vowels”的数量平均分为四个子标签“vowels”、“vowels-a”、“vowels-b”、“vowels-c”，这样数据分布相对平滑。

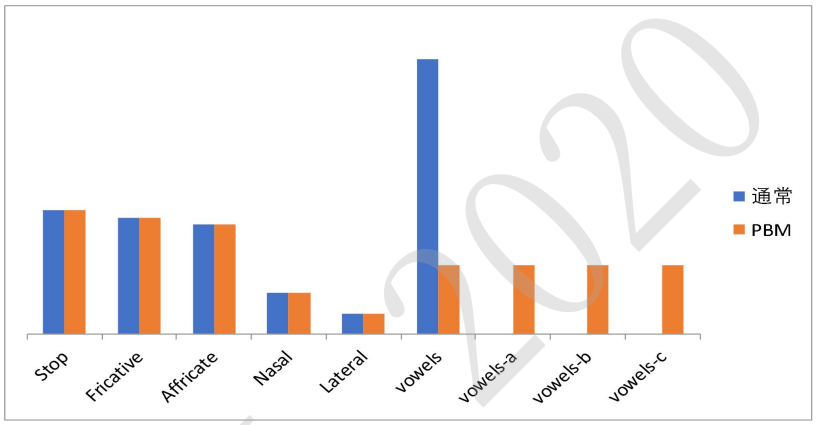


图 2.使用 PBM 方法对 PA 建模的数据分布

3.4 基于 FSA 的双语言属性检测器

众所周知，发音属性具备跨语言特性，为了探究基于 FSA 方法的跨语言属性识别能力，我们通过设计实验对照组，观察单语言和双语言训练的发音属性检测器在双语言属性识别任务中的性能对比。因为整合所有语言的发音属性本身比较难，我们还探索了单语言训练的属性检测器是否具备良好的多语言属性检测能力。但是 TDNN 和基于属性 HMM，两者同时建模发音属性的方法，有可能使得模型过于适应汉语发音习惯，而弱化发音属性原本的语言通用性质。因此，我们通过减少建模时使用的上下文的信息弱化模型对单个语言的适应性和依赖性，之后对比双语言训练的属性检测器的性能来验证这样做的可行性。弱化上下文信息的属性建模，我们采用 Monophone-HMM 和普通 DNN 模型作为对照组。

3.5 基于 FSA 的不同母语背景发音人的发音偏误检测

利用上述已被验证的语言之间共享发音属性的结论，可在发音偏误检测任务中用于建模发音偏误。由于二语者受到母语负迁移的影响，其发音偏误的发音属性常常会倾向于母语中的相似发音的发音属性，也就是说偏误发音实际上是介于二语者的母语和第二语言之间的发音。利用这一点，结合整合语言的属性检测器，可用于直接建模该发音人的发音偏误。理论上，在跨语言属性检测任务中性能良好的属性检测器，拥有描述不同母语背景学习者的发音偏误的能力。

为此，针对上述基于 FSA 的单语言和双语言训练的属性检测器，我们在不同母语背景学习者的发音偏误检测任务上进行测试，通过分析两组属性检测器在该任务上的性能，来验证是否

跨语言属性识别性能良好的属性检测器，也会拥有更好的描述发音偏误的能力。我们使用的两种二语语料测试集，分别为母语俄语的发音人和母语日语的发音人。

4 实验设计和结果

4.1 实验对照组

我们通过对比上下文相关的 HMM（triphone）组合 TDNN、上下文无关的 HMM（monophone）组合 TDNN、上下文无关的 HMM（monophone）组合 DNN 的三种建模发音属性的方法设置对照实验，在英语、汉语属性识别任务中的观察三个对照实验的性能，来测试单一语言训练数据下的三种方法建模发音属性时的跨语言能力。

同时，为了更直观地观察上述三个对照实验的效果，我们单独设置了一个对照实验，采用上下文相关的 HMM（triphone）组合 TDNN 的建模方法，数据上使用汉语和英语双语语料作为训练集，两语言训练数据量比例为 1:1，训练数据总量同上述三种方法一致，同样在英语、汉语属性识别任务中观察性能。

在发音偏误检测任务上，我们在两个测试集上设置了总计四组对照实验，两个测试集分别母语为俄语的学习者的中文语料、母语为日语的学习者的中文语料。四组对照组实验为：单语言训练属性检测器组+俄语背景学习者测试集；单语言训练属性检测器组+日语背景学习者测试集；双语言训练属性检测器组+俄语背景学习者测试集；双语言训练属性检测器组+日语背景学习者测试集。

最后一项对照组实验在于基于 FSA 的属性建模和基线属性建模两种方法的对比，我们针对二语学习任务设计了细颗粒度的发音属性定义并根据该定义建模了七种属性检测器，组成了前端属性检测器组。其中，只有两种属性与基线属性定义差距较大，即舌位前后、舌位高低，因此我们设置了两个对照组，分别观察在这两种属性上基于 FSA 和基线属性建模的两项后端任务的性能，即属性识别性能、偏误检测性能。下图为对照实验设计示意图。

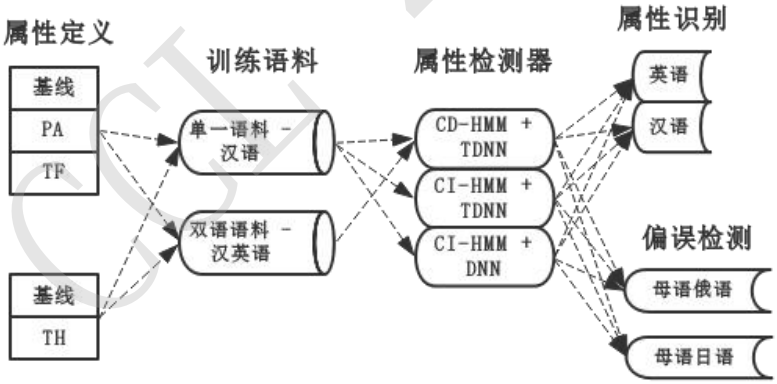


图 3.对照实验示意图

4.2 建模框架

本研究通过借鉴 ASAT 框架的整合思路，设计了基于 FSA 的建模框架。前端特征提取模块包含了一组属性分类器，用于提取属性后验概率，用于后端发音偏误检测任务，可以在不同维度上定义偏误检测，即超音段层级（如时长），音段层级（如音素替换），和次音段层级（如清化音素）（K.N.Stevens，2000）（G.Fant，1973）。本研究主要在次音段层级完成偏误检测实验，以及前端属性提取器的性能测试，具体过程框架如图 4 所示。

使用 MFCC 作为输入特征，设置对照组分别为 CD-HMMs、CI-HMMs，每个对照组包含七个基于发音属性的 HMM 模型；使用 MFCC 和 i-Vector 作为输入特征，两组基于属性的 HMM 做神经网络初始化，经过 PBM 的数据平衡后，建模基于属性的 TDNN 和 DNN，总计四个对照实验，每个对照实验七个模型；在每个前端分类器模型中，生成当前帧在该分类器中每个属性

的概率，即帧层级属性后验概率，作为前端输出。总计两个后端任务，将每个属性分类的帧层级后验概率用于评估基于 FSA 建模方法在中文和英文测试集上性能，之后进入强制对齐处理后转化为音素级后属性验概率进行次音段级发音错误检测，即中英文属性测试和次音段、音段偏误检测两项任务。

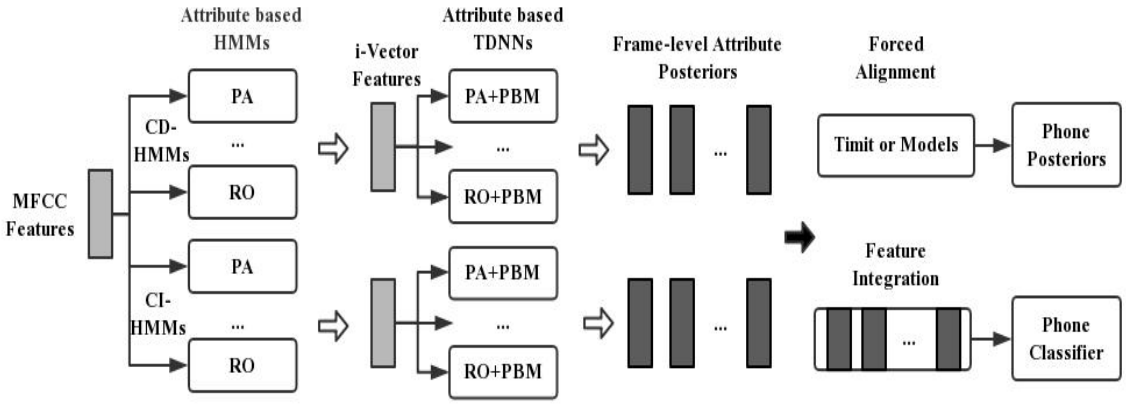


图 4.基于 FSA 的建模框架

根据后端任务的不同，分别对属性后验概率进行两种处理方式：在属性识别任务中，中文测试数据使用模型训练的强制对齐得到音素后验概率（Phone Posteriors），英文测试数据使用数据库自带的音素边界标注做对齐得到英文音素后验概率（Phone Posteriors）；在次音段偏误检测任务中，我们采用语音识别的整合过程（Phone Classifier）。此外，通过融合七种属性次音段级的偏误检测，我们完成了对不同母语背景的二语学习者语料库的音段级发音偏误检测，其中音频的音素边界信息是通过单独对二语语料本身建模后，经过强制对齐得到的。

4.3 发音属性检测

我们使用的语料库来自中国国家高新技术项目 863（S.Gao，2000），以及开源的 Aishell178 小时普通话语料库，英语语料库分别使用来自 LibriSpeech 和 Timit。单语言训练的属性检测器的训练数据共使用了 1800 名说话者（约 300 小时）的 25 万个话语进行声学建模，双语言训练的属性监测器的训练数据使用了 LibriSpeech 和 Aishel 两个语料库的数据，共 20000 条数据，约 300 小时，与单语言对照组的训练数据量保持一致，英语语料和汉语语料的比例为 1:1，充足的数据保证了基于 FSA 方法建模的鲁棒性。属性识别实验的的测试集有两个，一个是来自 Aishell 语料库的 6000 条中文数据，另一个是来自 Timit 的 6000 条英文数据。

我们对单语言训练的属性检测器在母语（Ch）和跨语言（En）发音属性检测任务上进行了评估；通过两种建模方法（上下文相关 CD、上下文无关 CI）和两个神经网络 DNN 模型、TDNN 模型，每个对照组包含三组对照实验（Triphone、Monophone、CI）。所有属性识别的实验结果如图 5 和图 6 所示。

由图 5 可知，上方三条曲线表示在汉语上测试（Ch）上表现出可靠的性能，即三个对照实验性能均在 80%以上，且上下文相关和 TDNN 组合建模（Triphone-Ch）的准确率，高于上下文无关和 DNN 组合（CI-Ch）建模的准确率。下面三条曲线在跨语言测试集（En）中表现出相对较低的检测准确率，尤其是元音部分，这表现出英语元音的结构与汉语差别很大，但是上下文无关和 DNN 组合（CI-En）建模的准确率，趋势上高于上下文相关和 TDNN 组合（Triphone-En）的准确率。经过更深入的观察，在跨语言属性检测任务中的多个属性检测器，如擦音（Fricative）和浊音（Voiced），可以获得较好的准确性（最高 93%和 78%）。我们还发现，在英语测试集上分类更精细的 TF 的属性集（见表 2）精度略优于 PA 分类（见表 1）。此外，由依赖于上下文的建模方法并不比上下文独立的建模优异，甚至 CI 方法在某些属性上也具有更高的效果，验

证了发音属性的语言独立性。

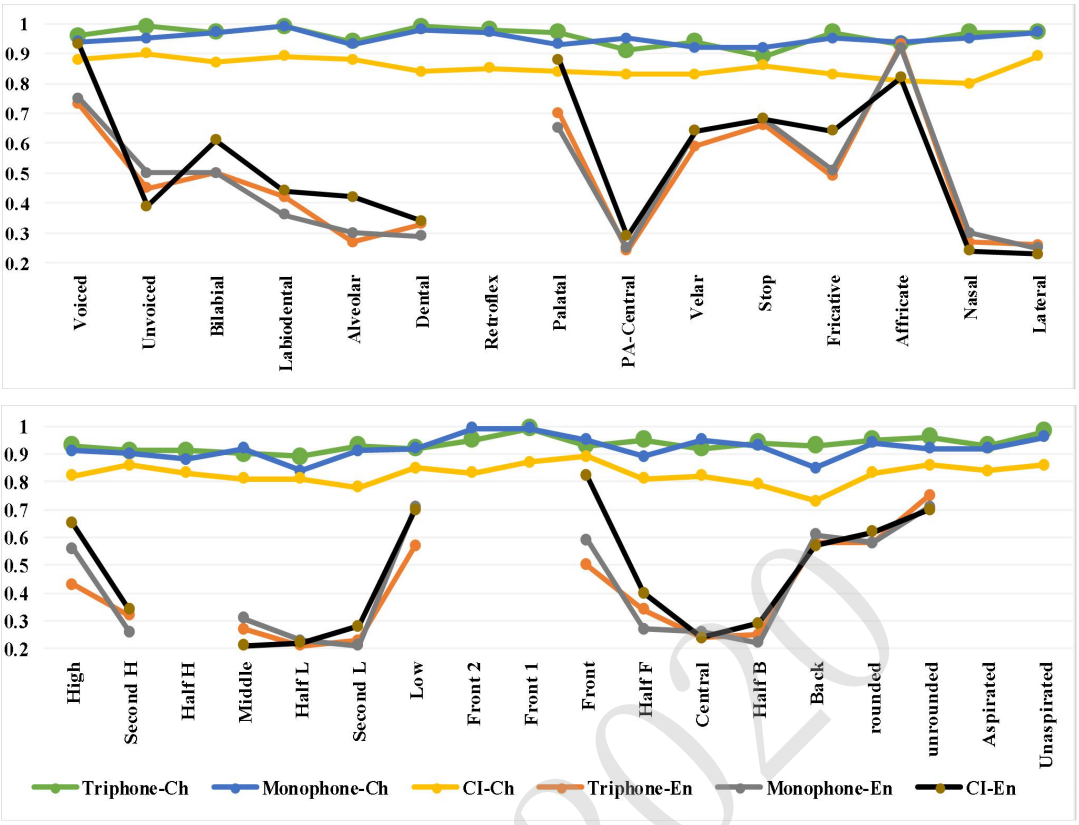
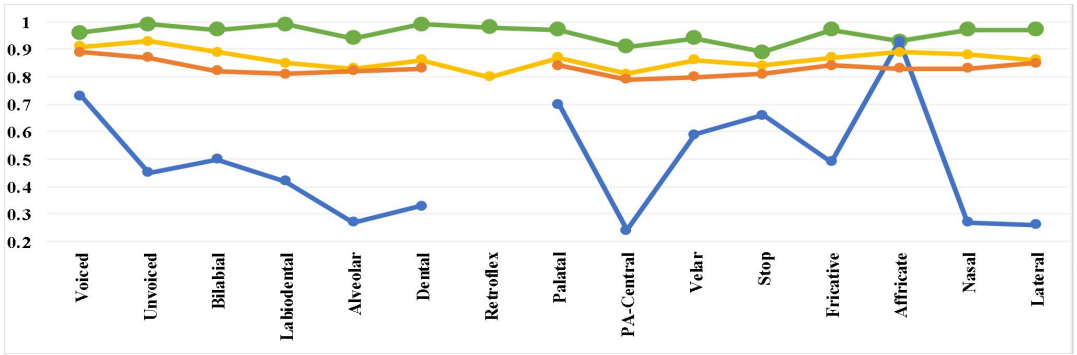


图 5.在汉语和英语上的基于 FSA 方法的检测准确率

我们同样对单语言训练的和双语言训练的属性检测器，在母语（Ch）和跨语言（En）发音属性检测任务上进行了评估，对应两个对照组（Ch、En），每个对照组包含两个对照实验（Monolingual、Bilingual），其中两个对照组上下文相关组合 TDNN，两个对照组除训练数据不同以外无其他差别。另外，由于英语中并没有 AS 属性，所以我们使用 PBM 方法平衡了双语训练集数据来训练 AS 属性检测器。所有属性识别的实验结果如图 6 所示。如图，准确率最高的两个曲线为汉语测试集上的属性识别结果（Ch），识别准确率在 80%以上，且单语言训练的属性检测器识别准确率（Monolingual-Ch）均高于双语言训练的属性检测器识别准确率（Bilingual-Ch）。图中下两条曲线反映了英语属性识别对照组的情况，其中双语言属性检测器识别准确率（Bilingual-En）远高于单语言属性检测器识别准确率（Monolingual-En）。



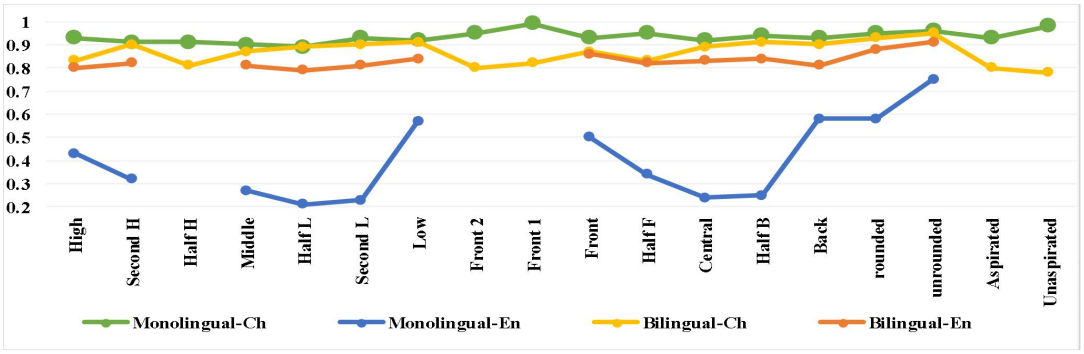


图 6.在汉语和英语上的基于 FSA 方法的检测准确率

4.4 发音偏误检测

发音偏误检测任务采用的二语语音数据库，使用北京语言大学汉语言语料库（J.-S.Zhang，2010），其中包含母语为俄语的普通话学习者的 1000 条语音，和母语为日语的普通话学习者 1000 条语言。为了在次音段级和音段级上检测发音偏误，我们使用了两个指标，即 F-score 和诊断准确率（DA）来评估发音错误检测的性能。

$$DA = \frac{N_M + N_C}{N} * 100\ \% \qquad \text{公式（4.1）}$$

$$F\text{-}score = \frac{2 * Precision * Recall}{Precision + Recall} \qquad \text{公式（4.2）}$$

其中 N_M 为检测到的真实偏误数，检测结果与人工标注一致。N_C 是系统检测到的真实正确发音的个数。Precision 真实偏误数和所有检测到的发音错误的数量的比值，称为准确率。其中 Recall 为真实偏误数和测试集中发音错误总数的比值，称为召回率。N 为测试集中音素或属性的个数。

在单语言训练的属性检测器组的对照实验中，我们选取了 7 个具有最好属性识别性能的分类器，并在两种测试集上对次音段级偏误检测性能进行评估，之后将它们整合至音段级的偏误检测中。我们可以看到，不论是母语为俄语还是日语的发音人测试集，基于 FSA 的方法都可以很好地检测出不同的发音属性的偏误，尤其单语言训练的属性检测器组（Monolingual-DA），次音段级诊断准确率均在 83%以上。更精细地体现汉语舌位变化的 TF、PA、TH，不论是在两个测试上（母语为俄语或日语）还是单语言或双语言训练（Monolingual-DA，Bilingual-DA）的属性检测器，表现均高于基线的 T-T 和 H-H。

	VO	AS	MA	PA	TH	TF	RO	T-T	H-H
Monolingual-DA	89.4%	89.0%	87.2%	83.4%	85.9%	84.3%	88.2%	82.7%	85.2%
Bilingual-DA	83.6%	82.4%	78.8%	72.6%	74.4%	73.9%	82.9%	72.2%	73.6%

表 3.母语俄语学习者次音段偏误检测

	VO	AS	MA	PA	TH	TF	RO	T-T	H-H
Monolingual-DA	91.4%	90.2%	88.6%	85.2%	86.7%	87.0%	90.7%	83.9%	85.7%
Bilingual-DA	84.7%	83.4%	81.2%	74.1%	76.0%	75.0%	83.6%	73.6%	74.9%

表 4.母语日语学习者次音段偏误检测

在母语为俄语的二语者的测试集上，通过对比，单语言训练和双语言训练的属性检测器组的整体诊断准确率（Monolingual-DA，Bilingual-DA），我们发现双语言训练的属性检测器组的偏误检测诊断准确率低于单语言训练的属性检测器。在母语为日语的二语者的测试集上的偏误检测结果，总体上母语为日语的学习者的次音段偏误检测准确率比母语为俄语的学习者要高， 可能是因为母语为日语的学习者的汉语总体水平高于母语为俄语的学习者。通过对比关于单语言训练和双语言训练的属性检测器组的诊断准确率（Monolingual-DA，Bilingual-DA），同样地， 双语言训练的属性检测器总体表现低于单语言属性检测器。

	FSA-based（M/B）		Segment-based（M）	FSA-based（M/B）		Segment-based（M）
F-score	71.5%	61.2%	63.5%	74.7%	62.8%	67.9%
DA	86.5%	78.4%	84.3%	88.5%	79.7%	85.8%

表 5.母语俄语/日语学习者音段偏误检测

将上述单语言和双语言训练的属性检测器组分别整合后，与同数据量训练（Aishell，约 300 小时）的基于音段的偏误检测相比（Monolingual，M），在两种母语背景学习者的测试集上，基于FSA 的偏误检测诊断准确率更高，F-score 更高，验证了本研究提出方法的有效性。此外，在母语为日语的发音人测试集中，音段偏误检测的性能均优于在母语为俄语的发音人测试集中的性能，包括基线系统（Segment-based）的基于音段的偏误检测诊断准确率；双语言属性检测器组整合后（Bilingual，B），用于音段偏误检测，在 DA 和 F-score 上低于单语言属性检测器组整合后的结果，这与次音段偏误检测中的结果一致。

5 结论

我们提出了一种基于细颗粒度发音属性（FSA） 识别并在发音偏误检测中应用。 实验结果表明， 在使用单一语言训练时，该方法提取了可靠的帧层级发音属性的准确率，均在 90%以上；在跨语言测试中，通过修改建模时使用的上下文信息降低了检测器在汉语上的过度适应， 建模时使用的上下文信息越少，单语言属性检测器性能越好，验证了发音属性的跨语言特性； 但是，使用上下文信息最少的属性检测器组，跨语言测试的准确率也远低于双语言属性检测器在英语属性识别任务中的性能，证明语言间音素结构的巨大差异依然有很大影响。在汉语属性识别任务中，单语言训练相比双语言训练的属性检测器组，准确率平均高出 7%，这表明双语言属性检测器，没有很好地表现出发音属性的语言独立性。相比单语言训练，双语言属性检测器组在英语属性识别任务中的性能提升明显，体现了属性的语言通用性。

在二语学习者的偏误检测实验中，使用基于FSA 的方法相比于传统发音属性定义的基线系统，次音段级别偏误检测任务中都表现了更优的性能，表明基于FSA 的方法在偏误检测任务中更能体现汉语语言发音的特点；同时，同数据量训练的基于发音属性的方法（单语言）比起基于音段的方法，在音段偏误检测任务中获得了更好的检测性能，近一步验证了基于 FSA 方法的有效性。

理论上，该方法可以应用于任何母语背景的学习者，我们通过在母语背景为俄语、日语的发音人语料库上的发音偏误检测，测试双语言训练相比单语言训练的的属性检测器，是否能拥有更好的描述发音偏误的能力，实验结果显示，单语言训练的属性检测器性能更优。经过分析， 可能由于双语训练使用的第二语料库，并非使用发音人的第一语言，即日语和俄语；双语属性检测器在汉语属性识别任务中准确率低于单语言属性检测器，即没有体现属性的语言独立性。

致谢

本论文受到国家社科基金项目（18BYY124），语言资源高精尖创新中心项目 (KYR17005)，北京语言大学梧桐创新平台项目（中央高校基本科研业务费专项资金）（19PT04），北京语言大学一流学科团队支持计划（GF201906）项目资助。本文通讯作者为解焱陆。

参考文献

Ambra Neri, Catia Cucchiarini, Helmer Strik. ASR-based corrective feedback on pronunciation: does it really work[J]. proceedings of interspeech icslp pittsburgh pa september, 2006:1982-1985.

C. Zhang, Y. Liu, and C.-H. Lee, “Detection-based accented speech recognition using articulatory features,” in Proc. ASRU, 2011.

Cao W , Wang D , Zhang J , et al. Developing a Chinese L2 speech database of Japanese learners with narrow-phonetic labels for computer assisted pronunciation training[C]// INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010. DBLP, 2010.

Duan R , Kawahara T , Dantsuji M , et al. Effective articulatory modeling for pronunciation error detection of L2 learner without non-native training data[C]// ICASSP 2017 - 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017. Florian Metze, Articulatory Features for Conversational Speech Recognition, Ph.D. thesis, Karlsruhe, Univ., Diss., 2005, 2005

G. Fant, Speech Sounds and Features. Cambridge, MA, MIT Press, 1973.

Harrison A M, Lo W K, Qian X, et al. Implementation of an extended recognition network for mispronunciation detection and diagnosis in computer-assisted pronunciation training[C]//SLaTE. 2009: 45-48.

J.-S. Zhang, W. Li, et al, “A Study On Functional Loads of Phonetic Contrasts Under Context Based On Mutual Information of Chinese Text And Phonemes,” in Proc. ISCSLP, 2010.

K. N. Stevens, Acoustic Phonetics. Cambridge, MA, MIT Press, 2000.

Li, W., Siniscalchi, S. M., Chen, N. F., & Lee, C. H. 2016. Improving non-native mispronunciation detection and enriching diagnostic feedback with DNN-based speech attribute modeling. IEEE International Conference on Acoustics, Speech and Signal Processing (pp.6135-6139). IEEE.

M. Karafiat, L. Burget, P. Matejka, O. Glembek, and J. Cernocky, “iVector-based discriminative adaptation for automatic speech recognition,” in 2011 IEEE Workshop on Automatic Speech Recognition & Understanding.

Muller M , Franke J , Waibel A , et al. Towards phoneme inventory discovery for documentation of unwritten languages[C]// ICASSP 2017 - 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017.

N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, Front-end factor analysis for speaker verification IEEE Transactions on Audio Speech & Language Processing, vol. 19, no.4, pp. 788-798, 2011IEEE, Dec. 2011, pp.

S. Gao, et al, “Update of Progress of Sinohear: Advanced Mandarin LVCSR System At NLPR,” in Proc. ICSLP, 2000.

S. M. Siniscalchi et al, “Toward a detector-based universal phone recognizer,” in Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, Las Vegas, Nevada, USA, Mar./Apr. 2008, pp. 4261–426

S. Xue, O. Abdel-Hamid, H. Jiang, L. Dai, and Q.-F. Liu, “Fast Adaptation of Deep Neural Network based on Discriminant Codes for Speech Recognition,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. PP, no. 99, pp.

Waibel A , Hanazawa T , Hinton G E , et al. Phoneme recognition using time-delay neural networks[J]. IEEE Transactions on Acoustics Speech and Signal Processing, 1989, 37(3):328-339.

基于抽象语义表示的汉语疑问句的标注与分析

闫培艺[1]，李斌[1]，黄彤[1]，霍凯蕊[1]，陈瑾[1]，曲维光[2]
1. 南京师范大学　文学院，江苏　南京
2. 南京师范大学　计算机科学与技术学院，江苏　南京
ypyheta@gmail.com; libin.njnu@gmail.com; iwanttardis@163.com;
kairui.huo.nj@gmail.com, chenn_jin@163.com; wgqu@njnu.edu.cn

摘要

疑问句的句法语义分析在搜索引擎、信息抽取和问答系统等领域有着广泛的应用。计算语言学多采取问句分类和句法分析相结合的方式来处理疑问句，精度和效率还不理想。而疑问句的语言学研究成果丰富，比如疑问句的结构类型、疑问焦点和疑问代词的非疑问用法等，但缺乏系统的形式化表示。本文致力于解决这一难题，采用基于图结构的汉语句子语义的整体表示方法—中文抽象语义表示（CAMR）来标注疑问句的语义结构，将疑问焦点和整句语义一体化表示出来。然后选取了宾州中文树库CTB8.0网络媒体语料、小学语文教材以及《小王子》中文译本的2万句语料中共计2,071句疑问句，统计了疑问句的主要特点。统计表明，各种疑问代词都可以通过疑问概念amr-unknown和语义关系的组合来表示，能够完整地表示出疑问句的关键信息、疑问焦点和语义结构。最后，根据疑问代词所关联的语义关系，统计了疑问焦点的概率分布，其中原因、修饰语和受事的占比最高分别占26.53%、16.73%以及16.44%。基于抽象语义表示的疑问句标注与分析可以为汉语疑问句研究提供基础理论与资源。

关键词： 疑问句；抽象语义表示；语义角色；中文信息处理

Chinese Interrogative Sentences Annotation and Analysis Based on the Abstract Meaning Representation

Peiyi Yan[1], Bin Li[1], Tong Huang[1], Kairui Huo[1], Jin Chen[1], Weiguang Qu[2]
1. School of Chinese Language and Literature, Nanjing Normal University, Nanjing, Jiangsu
2. School of Computer Science and Technology, Nanjing Normal University, Nanjing, Jiangsu
ypyheta@gmail.com; libin.njnu@gmail.com; iwanttardis@163.com
kairui.huo.nj@gmail.com; chenn_jin@163.com; wgqu@njnu.edu.cn

Abstract

The syntactic and semantic analysis of interrogative sentences has a wide application in the fields of search engines, information extraction and question answering systems. The NLP systems usually use a combination of classification and syntactic analysis to process interrogative sentences, with poor accuracy and efficiency. The interrogative sentence has rich linguistic research results, such as interrogative sentence structure types, etc., but it lacks systematic formal representation. We use Chinese Abstract Semantic Representation (CAMR) based on graph structure to annotate. The data comes from Penn Chinese Treebank 8.0, Chinese textbooks for elementary schools, and the Chinese translation of *Little Prince*, for a total of 2071 sentences. All kinds of interrogative words are represented by the combination of the interrogative concept—amr-unknown and the semantic relationship, which can represent the key information of

the interrogative sentence, the question focus and the semantic structure of the interrogative sentence. Finally, we calculate the probability distribution of the focus, of which the cause, modifier, and argument accounted for the highest proportion, respectively accounting for 26.53%, 16.73%, and 16.44%. Interrogative sentences annotating and analysis based on abstract semantic representation provides a better theory and resources for the study of Chinese interrogative sentences.

Keywords: interrogative sentences , abstract meaning representation , semantic roles , Chinese information processing

1 引言

随着人工智能的发展，自动问答(Sankar et al., 2019)、对话机器人(冯升, 2014)等系统成为了研究热门(Sankar et al., 2019)，其中疑问句的自动理解是自然语言处理中一项非常基础而复杂的任务。而现阶段疑问句的自动分析则主要采用问句分类(Madabushi et al., 2016)、句型识别(Maredia et al., 2017)、疑问焦点语义角色标注(彭洪保 等, 2009)等方法，精度和效率不理想。同时，随着聊天机器人(Hancock et al., 2019)、智能问答(Fan et al., 2019)等系统的发展，疑问句的自动分析越来越重要，这就需要从整体结构上把握疑问句的语义，为自动句法分析提供基础。

然而，传统的疑问句分析存在三个问题。首先，疑问句表示需要将问句分类和依存分析分别进行建模计算后再进行组合，效率较为低下。其次，现有问句分类方法难以解决一句多问的情况。例如图7例句“谁知道怎么赢？”是特指疑问句且拥有两个疑问焦点，当下分析方法难以清楚表示此类疑问句结构。最后，目前标注体系缺乏对省略、指代消解、小句关系等语言现象的有效表示方法，因此难以完整地表示疑问句的语义结构。

在语言学领域，疑问句相关研究集中在疑问句的结构类型等方面。而汉语疑问句以其结构复杂、形式多样等特点备受关注，如邵敬敏 (1996)、闫亚平 (2019)、赵睿艺 (2019)等，但是在形式化表示方面的研究较少，对计算没有直接帮助。

因此，本文尝试通过一种新的语义表示方法——抽象语义表示（Abstract Meaning Representation，AMR）来描写汉语疑问句，解决疑问句的疑问焦点、疑问结构、省略等问题，形成一个完整的疑问句语义表示体系，来服务于汉语疑问句理论和自动分析研究。本文通过2000多句真实语料的标注，测试了抽象语义表示的形式化表征能力，并统计分析出疑问句在疑问焦点和疑问结构上的分布特点。

全文结构如下：第1节梳理了疑问句的理论以及形式化表示的研究脉络。第2节总结了使用抽象语义表示标注汉语各类疑问句的特点，介绍了数据来源和标注方法。第3节统计了疑问概念标签amr-unknown的语义关系，分析了疑问代词的语义功能特点。第4节是结论和未来工作。

2 相关工作

疑问句是人们在日常生活中经常使用的一种句型，也是问答系统、搜索引擎、信息抽取领域中的主要使用句型。从传统语法时期就受到国内外语言学界的关注，相关研究不断进行。

2.1 疑问句的理论研究

传统语法时期，疑问句的研究主要围绕分类和表达效果展开，如Curme et al. (1931)、Jespersen (1933)。从语法角度根据表层结构将其分为一般、特殊、选择以及附加疑问句，认为疑问句除了表示询问等情感外，还有寒暄等语用含义。这些研究以描写为主，虽比如Nesfield (1911)也提到了变换（transformation），但未能触及到疑问句在句法语义层面的内在规律。此时期值得一提的是疑问代词，其研究成果较多，主要集中在指示代词和疑问代词的对比分析方面(Diessel, 2003)。结构主义语言学强调句子在语法研究中的重要性。布拉格学派提出了主位的概念，认为主位是一个句子的话题。主位的提出和疑问焦点的相关理论在某种程度上是一样的。Vachek et al. (1968)还提出了标记性（markedness）理论，最开始用来分析音位的区别性特征，后来人们也用来分析疑问句标记。

以Chomsky为代表的生成语法学派最有代表性的成果是对疑问句语序生成机制的分析。英语疑问句通常把系动词、助动词及疑问词置于句首，这和汉语保持原位不一样。生成学派将小

句的根设置为一个CP，英语助动词和疑问词在疑问句中从原位移入CP的C位；而在肯定句中，这个C由that充当。Chomsky (1973)针对特殊疑问句提出了wh-移位说，但该学派只关注句法层面疑问句的生成机制，不关注语义层面的表示。Baker (1970)认为疑问句在本质上是在生成时包含了一个疑问成分[+Q]。系统功能语法认为言语功能通过语气选择体现在合乎语法规律的小句中。Halliday et al. (2014)认为对一个语言项目进行分类时，应该按照精密度的阶，由一般逐步趋向特殊，对每一个选择点上的可选项给以近似值。

国内疑问句的研究历来属于语气范畴。马建忠 (2010)把语气分为传信和传疑。陆俭明 (1982)标志着疑问句的研究从宏观分类转向微观描写。吕叔湘 (1985)把疑问语气分为“询问、反诘、测度”三种，并将疑问句分为特指问和是非问两类，对疑问句的形式与功能关系等进行了讨论。在疑问句分类方面，王力 (1985)把疑问句分为：叙述句、描写句和判断句。黄伯荣 (1985)提出疑问句类型有特指问、是非问、正反问和选择问四类。邵敬敏 (1996)第一次将语法、语义、语用三个平面的理论运用到汉语疑问句的研究中，标志着汉语疑问句研究进入了新阶段。此后，疑问句理论研究成果也越来越多。在疑问代词方面，黎锦熙 (1992)认为有些疑问代词有“不定称”和“虚指”的用法，还有邵敬敏 等 (1989)、刘月华 (1985)等文的研究。

通过对国内外疑问句理论研究的梳理，可看出国外侧重于通过疑问句的形式探究疑问句本质，不断完善其生成机制。国内虽对疑问句进行了细致描写，比如分类体系等，这些有助于学科语言教学和句法理论研究，但对于疑问句的语义结构问题涉及较少，未能从整体上刻画疑问句的语义。

2.2 疑问句的形式化表示研究

随着疑问句理论的不断发展，国内外不断有学者尝试对疑问句进行表示，大致分为两类，一类是建立疑问句语料库，确定标注体系，另一类是一般语料库附带对疑问句标注方法的简单说明。

首先是疑问句语料库，国外比较著名的是Clark et al. (2004)从TRC 评测语料中抽取了1171句以what开头的疑问句，主要标注了词性信息。Judge et al. (2006)构建了一个含有4000句疑问句的语料库，数据主要来源于TREC跟踪测试集，以期生成的句法分析树对问答系统有所帮助。Myers (2007)针对法语wh-疑问句中不同句法结构可以表示相同语义的特点，建立了法语疑问句语料库。Mrozinski et al. (2008)提供了一个关于提问“为什么”疑问句的语料库，695句语料来源于维基百科。还使用Amazon Mechanical Turk框架收集了问句的匹配答案。Sidi et al. (2011)构建了马来语疑问知识语料库，以期完善马来语语法和语义规则。

接着是一般语料库中的疑问句标注，宾州树库选取了华尔街日报的真实语料，着重标注了句子中的短语结构和短语功能(Marcus et al., 1993)。布拉格依存树库主要由形态层（morphological level）、句法层(analytical level)和语义层(tectogrammatical level)构成(Alena et al., 2000)。这两个大型语料库数据丰富，但是都没有为疑问句设计系统的表示方案，对其处理较为简单。

国内关于疑问句形式化表示的研究发展比较缓慢，比较著名的是山西大学彭洪保（2010）的基于汉语框架网的疑问句语义角色标注语料库，其语料主要来源于山西旅游景点，共计3011句疑问句。该语料库提出了一种根据疑问句目标词共现率来判别疑问句所属框架的方法。李茹 等 (2009)的小型疑问句语料库包含1566句关于旅游景点五台山的疑问句，主要根据焦点进行了疑问句类别统计。

关于疑问句分类体系，国内较为著名的是哈尔滨工业大学的分类体系。文勖 等 (2006)在UIUC(毛先领 等, 2012)的基础上，根据汉语特点将疑问句分为人物、地点、数字、时间、实体、描述、未知七大类，以及根据实际情况又定义了60小类。在一般语料库中，也没有对疑问句的标注方法进行补充说明，比如哈尔滨工业大学依存语料库、清华大学语义依存网络语料库等。下面以哈工大的依存库为例，对“谁想去公园啊？”进行标注示例：

哈工大语义依存分析不像以往简单进行语义角色标注等浅层语义分析，而是通过依存结构将词汇之间的语义关系表示出来。在图1中，Aft表示感事，dCont表示操作的客事，Dir表示趋向，mPunc表示标点标记，mTone表示语气标记。句子的基本架构较为清晰，但对于疑问信息的表示还不够明确。例如，我们需要根据“谁”来确定疑问焦点，但是“谁”也有无疑而问的情况，例如“谁也做不出来。”同时，“啊”的意义也比较多样，仅根据mTone也难以判断其疑问含义。疑问句最重要的就是清楚知道该句到底在问什么，就是我们所说的疑问焦点是什么。该句是特指疑问句，那么疑问代词就是疑问焦点，我们需要将其标注出来，点明其语义关系，才有

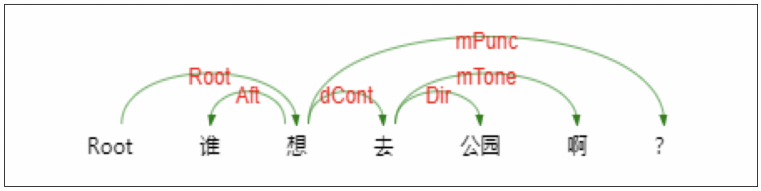


图 1: “谁想去公园啊？”的语义依存树分析

利于计算机的自动分析，上图并没有标识出疑问焦点。再者该分析也忽略了“去”“想”和“谁”的论元共享关系，不利于把握完整的深层语义。

随着自然语言处理的发展，国内外学者越来越重视疑问句的形式表示。国外集中在词性标注等方面；而国内关注分类等研究。总体而言，这些研究对于疑问句整体语义表示研究涉及较少，且研究重点较为分散，不利于计算和自动分析，也不利于系统研究。作为自然语言处理界新兴的句子语义表示方法，抽象语义表示能够更为完整地表示整句的语义结构和疑问结构信息。因此本文将基于抽象语义表示来标注汉语疑问句，系统介绍其标注方法，统计疑问焦点的语义关系等相关信息，以期对疑问句的研究和自动语义分析起到一定作用。

2.3 抽象语义表示研究

抽象语义表示（AMR）是一种新兴的完整的句子语义表示方法。它将句子中的词语抽象为概念，分析概念之间的语义关系，并将这些语义关系抽象为带有语义关系标签的有向弧，把句子语义抽象为一个单根有向无环图(Banarescu et al., 2013)。AMR将句子中词语抽象为概念，用图结构来表示概念以及概念之间的关系，拥有新增、删除、替换的抽象机制(Bos, 2016)。利用这一机制，AMR可突破表层句法结构的差异，将深层的语义结构统一表示出来。

AMR为英语制定，李斌 等 (2017)针对汉语特有的语法特点完善标注体系，形成了中文抽象语义表示（CAMR）。在CAMR标注体系中概念的编号不再由标注器随机分配，而是先对句子进行分词，根据词语序列分配相应编号。下面以“谁想去公园啊？”为例，对改进后的CAMR标注方法进行简要展示。

谁 [1] 想 [2] 去 [3] 公园 [4] 啊 [5]？ [6]	谓词信息
x2/想-01	想:01
:arg0 x1/amr-unknown	arg0:people described
:arg1 x3/去-01	arg1:thoughts of arg0
:arg1 x4/公园	
:arg0 x1/amr-unknown	
:mode x5_x6/interrogative	

图 2: “谁想去公园啊？”的CAMR表示

“谁”在该特指疑问句中是**疑问焦点**，是理解语义的关键，用**核心语义关系arg0（原型施事）和疑问概念amr-unknown**共同来表示，并且使用**关系mode和概念interrogative**点明了疑问语气类型。相对于哈工大的依存库来说，CAMR兼顾了“想-01”、“去-01”和“谁”的论元共享关系，语义结构表示较为完整。并且分词对应编号实现了语义图中的概念与原句词语的对齐。

自2013年标注规范公开发布以来，AMR语料标注工作不断推进。目前AMR已经有近五万句的英文语料库，语料内容来自新闻等领域。CAMR也公布了中文《小王子》语料库，还有向LDC提交的10000句对齐版的标注语料[1]，语料内容除CTB8.0外，还兼顾语文课本、微博等领域的数据。在自动分析方面，F值达到了61%(吴泰中 等, 2019)。本文主要基于CAMR对汉语疑问句进行标注。

[1]https://catalog.ldc.upenn.edu/LDC2019T07

3 数据来源及标注

3.1 数据来源

本文语料主要是从已经标注过的语料中抽取出来的疑问句：来源一是CTB8.0版的10149句网络媒体语料，其中疑问句1215句；二是2001年人教版一到六年级的语文课本中的8696句语料(戴玉玲 等, 2020)，其中疑问句692句，三是和英文《小王子》句对齐的中文小王子1563句，其中疑问句164句，共计2017句疑问句。

3.2 CAMR表示疑问句的特点

通过1.2节的梳理，我们可以发现：以往的疑问句形式化表示没有完整的标注体系，研究重点集中在分类和语义角色标注上。如果要理清疑问句的句子语义结构，这些是不够的。

CAMR的标注体系在AMR的基础上，根据汉语特点进行了优化，形成了一套较为完整的疑问句标注方法，具体特色如下：一是**设置虚节点标签**。CAMR使用xn(n∈N)的形式表示虚节点，n是根据输入的原始句子（基于分词结果）序列分配的有序编号。若为人工添加，则由系统随机分配。这样一来就实现了概念、关系与词的对齐。特别地，对于部分形式意义较为凝固的构式成分，CAMR将其整体作为一个谓词标注或只标注其表层义。AMR中的虚节点标签由概念单词的首字母表示，对于首字母相同的概念，不容易区分。二是**标注疑问语气**。语气信息对句子语义影响很大，尤其在书面汉语中。汉语没有严格意义上的形态变化，语气词和语法意义之间是多对多的关系，是否添加标点符号“？”、是否具有语气词等都会使整句的情感和语义发生变化。三是**既可以从整体上理解疑问句的深层语义结构，又能清晰把握疑问焦点的语义关系**。以往的疑问句研究多集中在分类和浅层语义分析上，CAMR允许根据句子语义增删概念节点，允许论元共享，如图2所示。它可以通过图结构清晰而完整地将整句语义表示出来。再加上疑问概念amr-unknown与不同语义关系的搭配使用设置，我们可以清楚地知道句子的疑问焦点是什么、具有什么样的语义关系，以及疑问焦点的对齐信息。

3.3 数据标注

CAMR中的语义关系分为两种：核心和非核心语义角色关系。用形如“argx(x∈[0,4])”的5个标签来表示核心关系，用如“cause（起因）”等48个语义标签来表示非核心语义角色关系。表1列出了CAMR表示疑问句常用的语义关系标签以及含义。

关系标签	含义	关系标签	含义
:arg0	原型施事	:degree	程度
:arg1	原型受事	:location	地点
:arg2	工具等	:manner	方式
:arg3	出发点等	:mod	修饰
:arg4	终点等	:poss	领属
:cause	起因	:quant	数字
:domain	陈述	:time	时间

表 1: 常用语义关系标签以及含义

处理疑问句时，除了常规的标注操作外，需要特别注意的是对疑问语气和疑问代词的处理。上表中的关系标签mode在CAMR中对应imperative祈使、interrogative疑问、expressive感叹和judgement判断四种语气概念，即用mode和interrogative共同表示疑问语气，将其标注在整句的根（root）上，若遇到有多个分句的长句，其中最后一个分句有语气的，语气则标注在此分句的根上。

主要标注对象有标点符号“？”、疑问语气词“吗”等。当句子中只有“？”或者疑问语气词时，疑问语气由“？”或者疑问语气词单独承担；当两者一起出现时，疑问语气由其共同承担。但当一个句子有多种语气时，如“他为什么这样呢！”既有疑问又有感叹，此时由“呢”承担疑问语气，由“！”承担感叹语气，将这两种语气都表示出来。最后，疑问代词“谁”、“什么”等使用概念标签amr-unknown搭配不同的语义关系标签来表示。

本文的疑问句标注借鉴现代汉语传统的分类体系(邵敬敏, 1996)——将疑问句分为是非疑问句、选择疑问句（包含正反疑问句）和特指疑问句三大类，同时也兼顾了其他一些特殊的疑问句结构，各类疑问句使用的主要关系及概念标签如表2。

类别	关系标签	概念标签
是非疑问句	:mode（语气）	interrogative（疑问）
选择疑问句（包含正反疑问句）	:mode（语气）、:opx（并列/选择）、:polarity（极性）	interrogative（疑问）、or（选择）
特指疑问句	:mode（语气）	interrogative（疑问）、amr-unknown（疑问代词）

表 2: 各类疑问句的基本关系及概念标签

3.3.1 是非疑问句

对于是非疑问句，CAMR使用关系标签mode和表示疑问的概念标签interrogative共同描写句子的疑问语气。

男孩 [1] 被 [2] 找到 [3] 了 [4] 吗 [5]？ [6]
x3/找到-01
:arg0(x2/被) x8/person
:arg1 x3/男孩
:aspect x4/了
:mode x5_x6/interrogative

图 3: “男孩被找到了吗”的CAMR表示

图3例子中“？”和“吗”一起承担了疑问语气，用“_”连接分词编号。“被找到”表示被动，因此增加了虚节点person来引出“找到”的行为施事，其标签编号由系统随机分配。再者，增加了词语和概念关系的对齐信息，使得虚词对应于概念节点或节点之间的关系弧上(Li et al., 2019)，“被”字引出施事，标注在了实词“男孩”和“找到”之间的有向弧上。另外AMR不标注体，CAMR根据汉语特点增加了关系标签aspect用于标注助词“着”、“了”等。

另外，是非疑问句中经常出现的“是不是”、“是否”等副词成分，如“他是否收集蝴蝶标本呀？”。这些副词是对事件的真实性进行发问，本质上也属于是非疑问句的范畴。所以CAMR在处理这些成分时，也会将其抽象表示为关系mode和疑问概念interrogative。

3.3.2 选择疑问句

CAMR 在处理选择疑问句时，会将表示选择概念的“或者”“还是”等替换为概念or。同时，搭配关系标签operator x，即opx，一起使用。另外在正反疑问句中，使用关系polarity和概念“-”表示否定概念。

你 [1] 喝 [2] 茶水 [3] **还是** [4] 咖啡 [5]？ [6]
x2/喝-01
:arg0 x1/你
:arg1 x4/or
:op1 x3/茶水
:op2 x5/咖啡
:mode x6/interrogative

你 [1] 走 [2] 不 [3] 走 [4] 啊 [5]？ [6]
x10/or
:op1 x2/走-01
:arg0 x1/你
:op2 x4/走-01
:arg0 x1/你
:polarity x3/-
:mode x5_x6 /interrogative

图 4: 选择（包含正反）疑问句的CAMR表示

在图4左例中，“还是”被等价替换为or，关系标签op1和op2对选择项进行了说明。右边例子中的选择项“走”和“不走”属于正反两种情况，将“不走”中的否定项“不”等价替换为“-”。

3.3.3 特指疑问句

在特指疑问句中，会将“什么”、“怎么”等疑问代词抽象为概念amr-unknown。

谁 1 帮 2 了 3 窝（我）4 这么 5 大 6 的 7 忙 8？ 9	你们 1 发现 2 了 3 **谁 4 的 5** 玩具 6？ 7
x2_x8/帮忙-01	x2/发现-01
:aspect x3/了	:arg0 x1/你们
:arg0 x1/amr-unknown	:arg1 x6/玩具
:arg1 x4/我	**:poss(x5/的) x4/amr-unknown**
:degree x6/大	:aspect x3/了
:degree x5/这么	**:mode x7/interrogative**
:mode x9/interrogative	

图 5: 特指疑问句的CAMR表示

图5左例中，“帮忙”是一个离合词，CAMR把“帮”和“忙”连接合并处理，且可将“窝”更正为正确的概念“我”。但是在比如哈工大的语义依存分析体系中，“帮”和“窝（我）”的关系则无法显示出来。在右边的例子中，CAMR使用关系标签poss表示“谁”和“玩具”之间的领属关系，“的”作为语义比较虚的词语，将其标注在“谁”和“玩具”之间的关系上。

3.3.4 其他疑问句的处理

一是“非疑问句+疑问小句”类附加问结构。该结构通常是由一个陈述小句，加逗号（也可不加），最后加上一个“是吧”、“是吗”等疑问小句组成。因为CAMR表示的是句子深层结构的抽象语义，所以语序对其标注没有影响。所以“是吗”等疑问小句本质还是对前面陈述句所表达事实的质疑，如图6左侧例子。

二是“难道”类反问结构。在CAMR中，关系标签mod（modifier），用来表示一般的修饰关系，用来表示衔接上下文的关系词，如“难道”“又”“再”等，如图6右侧例子。

大家 1 找到 2 他 3 了 4，5 **是 6 吗 7**？ 8	**难道** 1 女孩 2 发现 3 他 4 了 5？ 6
x2/找到-01	x3/发现-01
:arg0 x1/大家	:arg0 x2/女孩
:arg1 x4/他	:arg1 x4/他
:aspect x4/了	**:mod x1/难道**
:mode x6_x7_x8/interrogative	:aspect x5/了
	:mode x6 /interrogative

图 6: 附加问和反问类疑问句的CAMR表示

图6中的“是吗”是附着在陈述小句“大家找到他了”上，是对“大家找到他了”这个特定事实的质疑，所以将“是吗？”一起抽象为表示疑问语气的关系mode和概念interrogative，该结构的疑问句语义在本质上与是非疑问句无异。

三是间接问句。疑问短语可以单独成句，也可以作为一个结构成分出现在另一个句子中，通常是充当宾语。疑问短语做宾语有两种类型，一是全句为陈述句，如“你了解这是为什么。”这时宾语已经失去了疑问性质和功能。故不关注该类用法。二是全句为疑问句，如图7左侧例子。

四是自问自答类的设问句。自问和自答是设问句不可分割的一个整体，可以看出发问者其实是无疑而问，如图7右侧例子。采用multi-sentence（多句关系）概念标签来处理多个句子之间的关系，与关系标签sntx（x∈N*）配合使用。

在这一节中，我们对是非、选择（包含正反）、特指这三大类疑问句的标注方法进行了举例说明，同时也对一些特殊的疑问句结构进行了标注展示。CAMR既可以处理常规的疑问句标注，表达出深层的语义结构，也可以较好地表示一些无疑而问等特殊的疑问句表达。

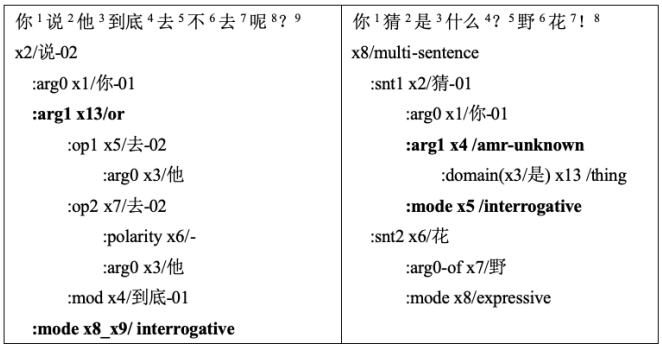


图 7: 间接问句和设问句的CAMR表示

4 统计分析

虽然CAMR无需借助分类系统分析疑问句的语义结构，但我们也可以利用表2相关标签统计出三大类疑问句的占比情况，如表3。从表中可以看出，特指疑问句的占比最高，达51.71%，选择疑问句最少，只有4.73%。

类别	次数/比例
是非疑问句	994/43.56%
选择疑问句（包含正反疑问句）	108/4.73%
特指疑问句	1180/51.71%

表 3: 各类疑问句的比例分布

4.1 特指疑问句的疑问焦点

CAMR允许根据句子语义增删概念节点，允许论元共享，既可以通过图结构清晰而完整地将整个句子深层语义表示出来，又可以通过语义关系和疑问概念amr-unknown搭配使用等把握疑问焦点信息，这对于我们准确理解疑问句非常有帮助。吕叔湘 (1985)指出“回答问话，一般不用全句，只要针对疑问焦点，用一个词或短语就够了”。对于疑问句来说，我们需要清楚的就是疑问句是针对什么提出疑问，疑问语义中心在哪里，即疑问焦点在哪里(唐燕玲 等, 2009)，这对于计算机自动分析是非常重要的。是非疑问句是对整个句子的客观事实提出疑问，那么疑问焦点就落在了整句的语义上；选择疑问句有选择项，那么opx关系标签所对应的概念标签就是我们需要关注的疑问焦点语义项。

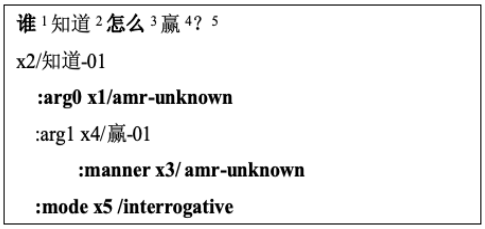


图 8: “谁知道怎么赢？”的CAMR表示

但是特指疑问句比较特殊，具有不一样的构成要素——疑问代词，比如“怎么”、“什么”、“哪里”等。疑问代词作为句法功能和意义的结合，是特指疑问句的疑问焦点(唐燕玲 等, 2009)。林裕文 (1985)也指出“特指是对准疑问代词回答的”。再加上有的特指疑问句不止一个疑问焦点，仅从疑问句分类角度难以准确把握完整的语义信息，如图8所示，该句有“谁”和“怎么”两个疑问焦点，分别具有arg0（原型施事）和manner（方式）两种语义关系，传统计算研究方法难以直接处理。针对特指疑问句要素特点，CAMR使用疑问概念amr-unknown，同时搭

配各种语义关系来共同表示疑问焦点信息。疑问代词的不同使用方法可能会有不同的语义关系，下面将通过统计数据详细分析疑问代词语义角色的分布特点，总结疑问代词的语义功能特点。

4.2 疑问概念amr-unknown的语义关系特点

本文对2071句疑问句中的1410个疑问代词所对应的1410个概念amr-unknown的语义关系信息进行了统计，不同语义关系的使用分布情况如表4。

语义关系（含义）	次数/比例	语义关系（含义）	次数/比例
:cause（起因）	**373/26.53%**	:source（源）	10/0.71%
:mod（修饰）	**236/16.73%**	:purpose（目的）	8/0.57%
:arg1（原型受事）	**232/16.44%**	:degree（程度）	6/0.43%
:arg0（原型施事）	168/11.90%	:value（值）	3/0.21%
:manner（方式）	134/9.50%	:destination（目的地）	2/0.14%
:quant（数字）	58/4.11%	:beneficiary（受益者）	2/0.14%
:arg2（间接宾语、工具等）	52/3.68%	:day（天）	2/0.14%
:opx（选择项）	38/2.70%	:arg3（出发点、收益者等）	1/0.07%
:time（时间）	26/1.83%	:frequency（频率）	1/0.07%
:domain（陈述）	21/1.48%	:direction（方向）	1/0.07%
:poss（领属）	19/1.35%	:topic（话题）	1/0.07%
:location（处所）	16/1.13%	合计	1410/100%

表 4: 疑问概念amr-unknown的语义关系分布

从表4可以看出，在本次统计中，疑问概念amr-unknown各类语义关系有23种，总共出现了1410次，但分布不平衡，使用频率较高的前三大类依次是cause、mod以及arg1，分别用来提问原因、修饰成分以及原型受事，分别占比26.53%、16.73%以及16.44%。在出现的4种核心语义关系中，概念amr-unknown为受事的语义关系最常见。在出现的4种核心语义关系中，概念amr-unknown为受事的语义关系最常见，施事、间接宾语次之。非核心语义关系有19种，种类比较多，且出现总次数是核心语义关系的两倍左右，达67.87%。这些不同的语义关系代表的是说话人不同的提问对象，弄清疑问代词的不同语义关系是什么，是我们把握特指疑问句语义重点所在，也是问答系统提高回答准确率的关键所在。

4.3 小结

通过对2071句疑问句的标注，我们可以看出CAMR可以完整而清晰地表示出汉语疑问句的整体结构。以往处理疑问句的方法，比如问句分类、依存分析等，很难完整表示出疑问句结构的深层语义。通过对1410个疑问概念amr-unknown的语义角色种类进行统计分析，发现cause、mod以及arg1的语义关系使用最为频繁。在CAMR的标注体系下，处理疑问句有一套完整的标注体系，无需设置分类标签，通过语义关系标签就可以知道句子的疑问焦点是什么，位置在哪里，从而准确把握整句的语义结构。

5 结论及未来工作

随着自然语言处理领域的不断发展，其中以问答系统最为突出，疑问句的形式化表示越来越受到各界学者的重视，但是由于汉语疑问句形式多样，结构复杂，目前还没有比较完整的标注体系可以很好地表示汉语疑问句的整体结构。本文首先梳理了国内外疑问句的相关理论与计算研究。接着使用改进之后的CAMR体系针对2071句汉语疑问句，对不同结构类型疑问句的标注方法进行了说明。最后对1410个疑问概念amr-unknown的语义关系种类进行了统计分析，发现其非核心语义角色的使用频率最高。这一标注体系不需要进行疑问句分类，就可以更好地描写疑问代词的功能，把握其语义关系，对问答系统作出正确回答有很大的帮助。

在未来工作中，我们会扩大汉语疑问句的语料规模，丰富语料类型，关注口语化的疑问句表达，进而继续完善CAMR标注体系，推动相关理论研究。最后，希望通过标注语料库进行机器学习，不断提高CAMR语义自动分析效果，推进疑问句的自动分析和应用。

致谢

本文得到以下基金项目的支持：国家社科基金项目（18BYY127）；国家自然科学基金（61772278）；江苏省高校哲学社会科学优秀创新团队建设项目，在此一并感谢。

参考文献

Alena Böhmová, Jan Hajic, Eva Hajicová, Barbora Hladká. The Prague Dependency Treebank: A Three-Level Annotation Scenario[A]. In Treebanks: Building and Using Parsed Corpora[C], Amsterdam: Kluwer, 2000:103-127.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, Michael Auli. ELI5: Long Form Question Answering[J]. arXiv: Computation and Language, 2019: 3558–3567.

Angel Maredia, Kara Schechtman, Sarah Ita Levitan, Julia Hirschberg. Comparing Approaches for Automatic Question Identification.[C]// Proceedings of the 6th Joint Conference on Lexical and Computational Semantics, Vancouver, Canada: Association for Computational Linguistics, 2017: 110-114.

Baker, Carl L. Notes on the Description of English Questions: The Role of an Abstract Question Morpheme[J]. Foundations of language, 1970: 197-219.

Bin Li, Yuan Wen, Li Song, Weiguang Qu, Nianwen Xue. Building a Chinese AMR Bank with Concept and Relation Alignments[J]. Linguistic Issues in Language Technology, 2019, Vol 18.

Braden Hancock, Antoine Bordes, Pierre-Emmanuel Mazare, Jason Weston. Learning from Dialogue after Deployment: Feed Yourself, Chatbot![C]// Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy: Association for Computational Linguistics, 2019: 3667-3684.

Chinnadhurai Sankar, Sandeep Subramanian, Chris Pal, Sarath Chandar, Yoshua Bengio. Do Neural Dialog Systems Use the Conversation History Effectively? An Empirical Study[J]. arXiv: Computation and Language, 2019.

Chomsky, Noam. Conditions on Transformations[A]. Andersen, Stephen and Paul Kiparsky. A Festschrift for Morris Halle[C], New York: Holt, Rinehart and Winston, 1973:232-286.

Curme, George Oliver. A Grammar of the English Language in Three Volumes. Vol. 3. [M]. Berlin：Indogermanische Forschungen,1931.

Diessel, Holger. The Relationship between Demonstratives and Interrogatives[J]. Studies in Language, 2003, Vol 27.3: 635-655.

Fatimah Sidi, Marzanah A. Jabar, Mohd Hasan Selamat, Abdul Azim Abdul Ghani, Md Nasir Sulaiman,Salmi Baharom. Malay Interrogative Knowledge Corpus[J]. American Journal of Economics and Business Administration, 2011, 3(1): 171-176.

Harish Tayyar Madabushi, Mark Lee. High Accuracy Rule-based Question Classification using Question Syntax and Semantics[C]// the 26th International Conference on Computational Linguistics: Technical Papers, Osaka, Japan: The COLING 2016 Organizing Committee, 2016: 1220-1230.

Jespersen, Otto. The System of Grammar[M]. London:G. Allen & Unwin ltd, 1933.

Joanna Mrozinski, Edward W. D. Whittaker, Sadaoki Furui. Collecting a Why-question Corpus for Development and Evaluation of an Automatic QA-system[C]//Proceedings of ACL-08: HLT, Columbus, Ohio: Association for Computational Linguistics, 2008: 443-451.

Johan Bos. Expressive Power of Abstract Meaning Representations[J]. Computational Linguistics, 2016(3): 527–535.

John Judge, Aoife Cahill, Josef van Genabith. Question Bank: Creating a Corpus of Parse-Annotated Questions[C]// Proceedings of the 21st International Conference on Computational

Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics, Sydney, Australia: Association for Computational Linguistics, 2006: 497-504.

Josef Vachek. The Linguistic School of Prague[J]. Journal of the American Oriental Society, 1968: 369.

Banarescu L, Bonial C, Cai S, et al. Abstract Meaning Representation for Sembanking[C]// Linguistic Annotation Workshop and Interoperability with Discourse, Sofia, Bulgaria: Association for Computational Linguistics, 2013:178–186.

Lindsy Lee Myers. WH-interrogatives in Spoken French: A Corpus-based Analysis of their Form and Function[D]. Diss. 2007.

Michael Alexander Kirkwood Haliday. An Introduction to Functional Grammar[M]. London: Edward Arnold, 1985.

Mitchell Marcus, Mary Ann Marcinkiewicz, Beatrice Santorini Building a Large Annotated Corpus of English: the Penn Treebank [J]. Computational Linguistics, 1993: 313-330.

Nesfield, John Collinson. Idiom, Grammar, and Synthesis[M]. London: Macmillan and Co. Ltd, 1929.

Stephen Clark, Mark Steedman, James Curran. Object-Extraction and Question-Parsing Qsing CCG[C]// Conference on Empirical Methods in Natural Language Processing, Barcelona, Spain, 2004:111-118.

戴玉玲, 戴茹冰, 冯敏萱, 李斌, 曲维光. 基于关系对齐的汉语虚词抽象语义表示与分析[J]. 中文信息学报, 2020, 34(04): 21-29.

冯升. 聊天机器人系统的对话理解研究与开发[D]. 北京邮电大学, 2014.

黄伯荣. 陈述句, 疑问句, 祈使句, 感叹句[M].上海:上海教育出版社,1985.

李斌, 闻媛, 宋丽, 卜丽君, 曲维光, 薛念文. 融合概念对齐信息的中文AMR语料库的构建[J]. 中文信息学报, 2017, 31(06): 93-102.

黎锦熙. 新著国语文法[M]. 北京: 商务印书馆, 1992.

李茹, 王文晶, 梁吉业, 宋小香, 刘海静, 由丽萍. 基于汉语框架网的旅游信息问答系统设计[J]. 中文信息学报, 2009, 23(02): 34-40.

林裕文. 谈疑问句[J]. 中国语文, 1985, (2): 91-98.

陆俭明. 由“非疑问句形式+呢”造成的疑问句[J]. 中国语文, 1982: 640.

刘月华. “怎么”与“为什么”[J]. 语言教学与研究, 1985(04): 130-139.

吕叔湘. 疑问. 否定. 肯定[J]. 中国语文, 1985(4): 274.

马建忠. 马氏文通[M]. 北京:商务印书馆, 2010.

毛先领, 李晓明. 问答系统研究综述[J]. 计算机科学与探索, 2012, 6(3): 193-207.

彭洪保, 李茹, 段建勇. 基于汉语框架网的问句语义角色自动标注研究[C]. 中国计算机语言学研究前沿进展（2007-2009）. 北京: 清华大学出版社, 2009:220-225.

彭洪保. 基于汉语框架网的问句语义角色标注研究[D]. 山西大学, 2010.

邵敬敏. 现代汉语疑问句研究[M]. 上海: 华东师范大学出版社, 1996.

邵敬敏, 赵秀凤. “什么”非疑问用法研究[J]. 语言教学与研究, 1989(1): 26-40.

唐燕玲, 石毓智. 疑问和焦点之关系[J]. 外国语(上海外国语大学学报), 2009, 32(01): 51-57.

王力. 中国现代语法[M]. 北京: 商务印书馆, 1985.

文勖, 张宇, 刘挺. 基于句法结构分析的中文问题分类[J]. 中文信息学报, 2006, 20(2): 33-39.

吴泰中, 顾敏, 周俊生, 曲维光, 李斌, 顾彦慧. 基于转移神经网络的中文AMR解析[J]. 中文信息学报,2019, 33(04): 1-11.

闫亚平. 汉语附加问句句法形式的浮现与发展[J]. 汉语学报, 2019(03): 21-29, 95.

赵睿艺. 现代汉语“疑问代词+V+不是V”构式研究[D]. 华中科技大学, 2019.

语用视角下复述句生成方式的类型考察

马天欢
暨南大学/ 广东，广州
527738121@qq.com

摘要

本文将汉语母语者的160份复述文本与其原文进行以小句为单位的逐句比对，发现其中出现了6484对复述句对。从其生成的方式来看，可以分为改换词语和重铸整句两大类。以语用学原理对这些复述句进行分析，发现与以往研究的复述现象不同的是：句对间往往不具有相同的逻辑语义真值，但在特定语境下却能传达同一个语用意义，具有等效的语用功能。这说明在自然语言处理中，识别进入真实交际中的复述句不仅依赖语法、语义知识库，还需要借助含有语用知识和语境信息的知识库。

关键词： 复述句；语用；生成方式；知识库

A Pragmatic Study of Generation Method of Paraphrase Sentence

Ma Tianhuan
Jinan University / Guangzhou,Guangdong
527738121@qq.com

Abstract

In this paper, 160 paraphrase sentences of native Chinese speakers are compared with the original text sentence by sentence, and it is found that there are 6,484 pairs of paraphrase sentences. From the way of its generation, it can be divided into two categories: the change of words and the recasting of whole sentences. Based on the pragmatic principles, it is found that the paraphrase differs from previous studies in that sentence pairs often do not have the same logical semantic truth value, but they can convey the same pragmatic meaning and have equivalent pragmatic functions in a specific context. This shows that in natural language processing, the recognition of paraphrase sentences in real communication depends not only on the knowledge base of grammar and semantics, but also on the knowledge base containing pragmatic knowledge and contextual information.

Keywords: Paraphrase sentence , Pragmatic , Generation method , Knowledge base

1 引言

自然语言理解中的“复述”被认为是对相同语义的不同表达（Barzilay & Mc Keown，2001，转引自赵世奇，2009）。Glickman et al （2011）认为识别语义上等同的文本片段是文本理解应用的一项基础工作，因而G.M.Olson认为“复述”是判别计算机是否理解自然语言的标准之一，宋睿等（2019）也认为这是一项“经典的自然语言处理研究任务”。目前复述的研究已取得相当丰硕的成果（如胡金铭等，2013；刘明童等，2018），且应用在诸多领域中，如机器翻译、自动问答、自动文摘等，其研究价值可见一斑。

与此相关的研究包括语义相似性计算（Liu et al，2011; 刘宏哲、须德，2012）、文本蕴涵（郭茂盛等，2016），以及语料中同义词的识别和挖掘（郗亚辉，2016），这些都与近义表达密切相关。

在这类研究中，归纳类型是一项重要的研究内容。赵世奇等（2009）、姚振宇（2015）都总结出了主要的几类复述现象，包括同义词复述、语态变换、语序变换、句子结构变化、基于推理的复述等。马彬彬（2019）总结的13种复述现象，补充了外部知识引入、直述和间述变换等几类。在文本蕴含方面，任函等（2017）归纳的16类蕴涵现象，主要包括词汇相同而语序不同、词汇之间存在整体与部分、上下义关系等；金天华等（2019）考察了文本蕴含的成因，并归纳了词汇、句法异构、常识和社会经验3类。

总的来说，复述以及与之相关的近义表达所研究的对象中，多数研究者所关注的语言现象多为脱离语境的词或句，仅着眼于其静态抽象的词义或句义，如李新良、袁毓林（2013）所考察的动词蕴含关系只考虑词的概念义，并没有考虑语言单位在进入真实交际中所产生的语用意义。

而开始关注语境语用因素的有刘松（2014）发现目前问答系统研究中缺少对语用信息的重视，提出需要将语法信息、语义信息、语用信息都引入问答系统。陈千等（2018）突破了前人仅着眼于单句之间蕴含关系的局限，考察了多个句子与一个句子间的蕴含关系，指出这类蕴含关系的识别需要借助背景材料的语义信息，且覆盖多个片段，同时还强调了这种现象的普遍性和重要性。可以说，这项研究指出了在篇章中的同义表达不可忽略上下文的背景信息。而陈龙等（2019）发现了非字面义词的处理是语言深度理解中的一个棘手问题，从词典中发掘出了3524个非字面义二字词；如“主流”一词，该文认为它一个义项为“比喻事情发展的主要方面”，是其非字面义，而“河流的主要部分”，指的是具体的事物，是其字面义。但我们认为，任何词的义项的确定无不依赖语境，需要在具体的语境之下才能确定它此刻表示的是字面义还是非字面义。

倪盛俭（2013）曾指出当前文本蕴涵研究存在不足的根本原因是语言学角度的“研究严重缺乏，限于传统逻辑学和语言学的研究”；他认为语言外的知识不可排除在外，语言的理解有赖于语境，包括各类意象图式、脚本等。这方面在国外有研究者如Murray（2008）、Ofoghi &Yearwood（2010）等利用从框架网抽取的基于事件的特征或脚本等来识别文本蕴含。而目前国内汉语复述等同义表达的研究未见有这方面的突破。

总之，汉语当前已有的复述研究已取得不小的成果，但其中不足之处是：研究视角多限于脱离语境的静态抽象的词汇和句子，属于“无语境”的意义研究模式，忽略了需要考虑语境因素的篇章和话语中的复述现象。而任何语言单位只有在进入真实语用中，才有交际价值，且生成它临时的语用意义。因为意义必须依赖语境（黄希敏，2011）；且交际中任何词语、话语的生成和理解，语境都是一个影响词义的重要变量（王宁，2011）。何兆熊（1987）也认为“话语”的语用意义在句子语义基础之上，还存在于字面之外的“言外之意”。这是因为话语是实际交际中的词和句，必须从当下的情境来解答，它不像句子一样有固定的意义（Yule, 1996；Levinson，2000）。因而话语的理解往往需要一定的语用推理，从字面意义推断出其中隐含的话语含意（姜望琪，2014）。据此，张绍杰、杨忠（1993）认为有些话语甚至并不具有相同的命题内容，但其中隐含的语境意义、会话隐涵所传达的是同一个交际意图，如此构成“同义结构群”。

基于上述认识，本文集中关注在真实交际中为传达同一语用意义所采用的不同的表达方式。因此，本文将从汉语母语者的复述文本及其原文中提取出复述句，以语用学的视角考察这些进入使用状态的复述现象，揭示它们与游离于语境之外的语言单位的差异，补充现有研究的不足，为自然语言信息处理中对复述句研究提供基于语言事实的参考和依据。

2 篇章语用中复述句的获取

2.1 复述文本的来源

从现有公开的复述数据集来看，每个句子仅有一个或几个复述句，且多是基于传统的同义词替换或句式变换，往往强调“语义丰富”，鲜见有考虑语用因素而凸显语用变化的复述句。因此，尤为需要通过复述任务，驱动说话人或写作者在真实的交际情境中产出复述句，然后观察此类受语用因素干扰的复述句的特征。

为此，我们在汉语母语者的两个群体——广州某中学初中二年级和某高校本科三年级中分别随机抽取两个班，每班总人数都超过40人。在这4个班级中分别采用4篇不同的原文（下称“母文本”，字数均在1000字左右）进行复述测试。这4篇文章来自HSK六级考试中的写作测试——缩写[0]（详见表1）。这项“缩写”实质上是一项“读后脱稿笔头复述”，本文称其为“复述”。要求如下：（1）阅读下面这篇文章，时间为10分钟，阅读时不得抄写、记录。（2）10分钟后收回阅读材料。在35分钟内笔头复述原文，不少于400字。测试结束后收回复述文本，从每班中随机选取出40份文本（下称“子文本”），共计160份。然后将手写的原始文本转写为电子文档[1]。

2.2 复述文本的分析方法

要进行文本的比对，需要先将文本切分成一个个更小的单位，才能进行更精确的对应比对。为此，我们尝试了词语、句子、小句等为单位，最终选定以“小句”为切分和比对单位。然后确定了文本的分析步骤，具体如下：

步骤一：确定小句的切分标准

对于小句，在概念上我们采纳邢福义（1996）的界定。在形式上主要以逗号、句号等标点符号为形式标记。在操作上，主要参照杨毅、冯文贺（2018）的做法。

步骤二：切分子母文本的小句

确定了比对单位以后，按照上述切分标准，对子母文本逐一进行小句切分，结果见下表1。

母文本编号	母文本标题（自拟）	小句总数	子文本数量
01	乞丐搬砖	88	40份
02	胡萝卜、鸡蛋和咖啡	74	40份
03	两位司机	97	40份
04	老医生和年轻医生	76	40份
总共	160份		

Table 1: 复述母文本及子文本数量相关信息

步骤三：人工对齐子母文本的对应小句

将子文本中与母文本小句句意匹配的小句逐一对齐成近义句对。如下表：

母句编号	母文本02	子句编号	子文本02-1
0201	一天，女儿满腹牢骚地向父亲抱怨起生活的艰难，	0201-1	一天，女儿向她的父亲抱怨生活的困难，
0202	她说自己不知道该如何应付生活，	0202-1	她好像不知道怎样应付生活。
0203	好像一个问题刚刚解决，	0203-1	有时候刚解决了一个问题，

Table 2: 复述子母文本小句人工对齐比对示例[2]

根据母文本的编号对每一个子文本进行编号，如上据母文本02产出的第一个子文本编号为02-1，依次类推。小句的编号如第二个母文本的第一个小句编号为0201，对应第一个子文本的小句编号为0201-1，依次类推。

[0]真题来自《新汉语水平考试真题集》2012版（HSK六级），国家汉办／孔子学院总部编。

[1]本文仅关注文本的语义内容，故转写时对文中存在的语病做适当的修正，即根据我们母语者的语感，尽量还原原作者的意图。

2.3 复述句的类别及其数量分布情况

按照上述方法，我们对160个复述文本逐一进行以小句为单位的对齐和比对，从中提取出复述句对共6484对。然后，分析发现这些句对可以归纳为两大类（本文主要考察句对之间没有信息损耗的，其他类型句对将另文详述），具体如下表3:

复述句的类型	说明	频数	百分比
改换词语	替换语用中词义相似的词语，句式结构和句义一致	2352	36.27%
重铸整句	重组整句结构，句义保持	4132	63.72%
总计		6484	

Table 3: 篇章中复述句的类型总结

3 篇章中复述句的语用考察

以上我们从样本中提取到两大类共6484个复述句构成一个数据集。下面将以语用视角对这个数据中的样例进行分类分析。

3.1 改换词语

这类保持整句句式不变而替换局部个别词语的复述句，我们根据所替换的词语，将这类复述句分为4个小类。

第一，改换指称方式。在汉语中具有指称功能的语词主要包括名词、名词性成分、代词和零形式，称为“指称语”。进入篇章中的名词、名词性成分和代词往往具有具体的指称意义，但对同一个对象的指称表达，可以有不同的方式，这种现象廖秋忠（1992）称为“指同表达”。

子母文本的比对中发现句间存在一些表达形式不同，但在具体篇章中具有相同指称意义的指称语，其中人称指称语最多，例如:

0103	这个乞丐很可怜，	0103-3	他很可怜，
0302	向母亲乞讨。	0302-23	向我妈妈乞讨，
0206	父亲是一位著名的厨师。	0206-17	她父亲是一名著名的厨师，
0202	她说自己不知道该如何应付生活，	0202-38	她说她不知道怎么应付生活，

第二，替换语境下词义相同的词语。替换近义词是实现两个句子形成近义关系的常见手段，样本中也有大量这样的句对。但不同的是，有不少词语的替换，它们在脱离语境的情况下其近义关系往往不能成立。例如:

0234	喝了一口里面的咖啡。	0234-16	尝了一下（）。
0361	像今天一早，我就碰到您，	0361-22	就像今早我接到了你，
0156	……我有个秘密。”	0156-17	我有一个秘诀，

母句0234说女儿喝咖啡，前文是父亲让她看看咖啡豆煮过之后发生了什么变化，所以她尝试喝了一口。子句0234-22的“尝”非常精准地表达了“喝”在此处最确切的词义信息。第二例母句0161是出租车司机对上了车的乘客“我”说“碰到您”，根据交际者的身份关系，司机“碰到”乘客就是要把乘客“接到”某地。根据其社会关系和此时的行为活动，此时“碰到”即“接到”。

这即是词语在进入真实交际中，都会在其所在语境的制约下产生一个特定的“语用意义”，即“此人此时此地用此句是此意”（戴耀晶，2011）。而复述者在阅读理解的过程中通过认知加工，再现词语确切的语用意义。因而是该特定的语境促成了上述两词之间的近义关系。

第三，替换上下义词语。上下义关系是词汇系统中一种重要的语义聚合关系，但下面的这些词对是在所在语境下的上下位词，词义有一定的信息差额，形成包孕关系，如:

0319	我能干，	0319-3	我能搬，
0375	母亲说：“我们不能接受你的照顾。”	0375-4	母亲说：“我们也拒绝你的房子。”

在没有语境制约的情况下，如上“干”和“搬”等词对并不构成上下义关系，但在这一情景之下，说话人所说的“干”的具体所指即“搬”，复述者把该词的所指范围缩小、具体化。冉永平（2012）、叶慧君（2018）把这种现象称为“语用收缩”。这实际上也是一个推理过程，即寻求某一词汇或结构在特定条件下的精确意义（Levinson，2000）。

下面这一例与前面几种情况是一个相反的推导过程，请看：

0172	挤在他们旁边看看夕阳才走，	0172-30	我就挤在那边看看风景才回去，

第四，替换模糊量表达。有些句子中出现了精确的数量表达，但在其子句中往往被替换为约略笼统的含糊表达，如：

0120	不信你每天来坐12个小时看看，	0120-1	不信你每天坐十几个小时试试，
0217	20分钟后，父亲关掉了火，	0217-4	不久，父亲关掉了火。

第一例母句是出租车司机谈及自己的工作时间长达12小时，用的是精确描写法。而对应的子句用了“十几个小时”这样的概数虽流失了一些信息，但这样的表达方式甚至更能传达说话人此时想强调的工作时间长，与母句语用等同。而第二例文中父亲具体用了多长时间来煮并不重要，子句替换为“不久”同样能产生与母句等效的语用功能。

这种模糊语言现象是一个完整的语言理论中的一个组成部分，并且模糊语言对语境具有很强的依赖性（Channell，1994；Ruzait，2007）。冉永平（2012）则认为这种现象是话语中的“语用松散”，即一个词汇或结构的四周分布着围绕其语义原型而出现的可能选项，构成一个待选集合，不同的成员与语义原型的接近度不同。这恰好体现在我们的子文本中，即对于一个原词、原句，不同的子文本有不同的再现方式，这些方式就形成了一个集合，但成员之间以及与词语、结构都具有一定的近似性。

以上归纳了通过改换词语形成的复述句的4个子类。从样本中挖掘出来的语用中可以互相替换的词语甚至词集中，都充分说明了语境是一个不可忽略的重要因素，即语境可以促进或限制语用中词语近义关系的建构。这验证了庞杨、张绍杰（2012）、庞杨（2015，2016）的研究结论——词汇的同义关系除了依靠简单的语义联系，还需要通过语用推理机制在动态语境中调整和选择而构建形成。因此，纯粹依赖以往的“同义词词林”等仅着眼于抽象词义的知识资源还远远不够。

可见，基于语言事实，挖掘语用中词义的相似性（多词一义）和词义的相关性（一义多词），并对此类词对近义关系形成的机制，以及如何形式化以实现机器的识别、表征和计算，对知识库的建设乃至与语义相似性相关的实践应用都有所裨益。

3.2 重铸整句

绝大多数传统语法学、修辞学等所考察的“同义形式”和汉语“复述句”相关的研究都没有考虑语境因素。而我们从复述文本中提取到大量句对恰恰高度依赖语境。我们将此类复述句归纳为以下3个子类。

第一，推导言语行为意义。Austin（1962）指出,“言语行为”是人类言语活动的行为性质和行事意义,是字面语力和间接的施为语力。在实施言语行为的过程中，说话人通过其话语意义传达某一交际意图，完成某些功能，如拒绝、命令等，且这个用意是在字面意义的基础上结合语境推断出来的。

样本中出现一些在言语行为意义上构成一致的近义句对，如：

0452	不用担心啦，	0452-9	没什么大事，
0226	看看有什么变化。	0226-9	然后说出有什么不同。

劝阻句0452是医生（说话人）劝阻病人（听话人）“担心”，这是医生的用意，即以言行事；子句0452-9“没什么大事”则是以言指事。这两个话语都是要在听话人身上达到一个效果——让病人不要担心，促使他们放松、不担心，即“以言成事”（又称言后行为），这就是说者言语行为的意义，即隐含的用意。

上述这些句对之间并不具有相同的字面意义，但都传达同一个交际意图，达到了同一个交际目的。

此外，还有子文本将母句的言语行为进行抽象的，如：

0375	母亲说：“我们不能接受你的照顾。”	0375-30	母亲不接受他的照顾。
0309	对乞丐说：“你帮我把这堆砖搬到屋后去吧。”	0309-18	但是母亲还让他把砖搬到屋后去。

如上述前几例中分别用“不接受”和“让”抽象原直接引语句中的言语行为。

此外，评价也是一种言语行为，评价意义是说话人所传达的或褒或贬的意义（宗世海，2000）。在篇章中，作者通过评价事物或人物来表达某种主观倾向，这种倾向性也是一种言语行为意义。叶花（2006）认为评价意义也是话语中隐含的用意。

以下这些样例就是句对之间的评价倾向性一致，请看：

0185	原来喜欢他的不只我一个。	0185-15	原来不只是我认为司机很好，
0174	我突然意识到自己很幸运，	0174-14	我觉得这位司机非常好，

如母句0185“喜欢他”暗含着“我”对他持正面评价，也与0185-15“我认为司机很好”有相同的主观倾向性。

第二，通过语用充实。国内外越来越多学者（如Carston，2002；Wilson，2004；冉永平，2008等）发现词汇或结构的使用和理解的过程不是一个简单的信息编码－解码的过程，需要交际者根据特定的语境条件对它们进行不同程度的语用加工。

据此，冉永平（2012：79）提出了在语言运用中通过“语用充实”来确定和获取交际信息的过程，指的是听话人根据语境，“对它们（话语中的词汇）进行不同程度的语用加工，使其成为特定的语境化信息”，包括“语用收窄”和“语用扩充”两种类型。我们借用“语用充实”这个术语来论述子文本通过语用加工来再现原意的现象。

以上的实例多是依赖上文内容推导的，除此之外还有部分复述句需要借助下文信息推断获得，如：

0181	我决定跟这位司机要个电话，	0181-38	就决定跟他要一张名片，
0516	他想原来的厨师肯定不会继续干这份工作了。	0516-9	老板以为他第二天不会来工作了，

如上0181-38是从母文本后文“接过他名片的同时，他的手机铃声正好响起”获知他要了司机的“一张名片”。0110-23也是从后文得知他每天工作的具体时长是12个小时。

第三，推导修辞意义。Grice(1975)提出的会话中的合作原则及其4个准则，包括数量准则、质量准则、关联准则和方式准则；并指出隐喻、反语、夸张等此类现象都是有意违反会话合作准则，认为违反会话准则时就会产生“特殊会话含意”。这些喻意性结构的字面意义往往不是特定语境下说话人的交际意义；而是始于这个显性的字面意义推知隐含的信息（冉永平，2012）。徐盛桓（1996）、宗世海（2002）认为比喻、拟人等修辞就是含意的运用，这些修辞性语句的非字面义就是含意。

我们从样本中发现一些修辞性表达在子文本中被改写，例如：

0505	有一天,幸运之神终于眷顾了他,	0505-7	有一天他很幸运,
0190	心情就更要争气。	0190-35	但人更要争气。
		0190-14	我们都应该让自己更快乐,

总之，以上所罗列的复述句中，句对之间往往不具有相同逻辑语义真值，但在进入具体特定语境时，却能生成相同的会话隐涵或言外之意，传递同一个交际意图。这种现象在真实的口头话语、书面篇章中普遍存在，而我们交际者往往都能“心领神会”地理解，并准确地选用恰当的方式自如地表达，这是因为交际双方除了基本语言知识之外，还都有共知的背景知识和语境信息。

4 结语

本文的考察复述句都是进入具体篇章中为特定语用目的服务的语句，反过来语境又赋予了它们临时特定的语用意义。可以看到在改换词语的这一类复述句中，有相当一部分近义词高度依赖语境，而且它们并不总是具有相近的静态词义。也即在特定语境下词义的差异可能缩小甚至消失，而形成近义关系；也可能其差异得以凸显而限制了近义关系的建立。而传统语义学、词汇学等只限于对词汇的真值做静态观察和描写，而事实上进入使用状态中的词意是动态流变的，词汇本身的静态意义发生一定的伸缩，甚至变异。词语近义关系的成员词也会根据情境即时生成建构或即时消失。可见，交际语境共生是语用意义生成的根本途径，语用环境是解释进入使用状态的词语近义关系必不可少的一个重要因素。

而在重铸整句这类复述句中，很多情况下仅仅依赖抽象的句义也无法判断它们的关系，其近义关系需要在特定语境的制约下才能成立，即语境对话语近义关系的形成有促进或限制的作用。这印证了吕叔湘（2008）的说法——语言活动中出现的意义还包括环境给予的意义。可见，词语、话语作为语言的基本建筑材料，理解语篇首先是对词义、句义的理解，而语境是理解词义、句义不可忽略的因素。如果仅用语义学的意义观，则无法解释这些在真实交际中广泛存在的同义手段。

可见，在真实交际中形成的复述句，相比以往复述句、文本蕴含以及传统的“变换分析”、“同义形式”等更复杂多样，其主要特征可以归纳为3点：（1）需要基于句子本身静态抽象的语义和语法等语言学知识；（2）依赖上下文语用知识、语境信息和百科知识等非语言知识；（3）需要借助一定的语用推理和逻辑知识。

另一方面，从上述语用中的复述句分析可以看到，此类需要利用语用知识判断的复述句在真实交际中广泛存在且类型繁多，是实现机器准确理解语义，并进一步完成其它实践应用的关键环节。这就给我们一个更关键的启示：根据上文实例中发现的篇章语用中复述句的特征和类别，要实现计算机准确地识别此类复述句，相应地需要提供的知识库包括语义知识、同义词词林等语言学知识库、语境语用知识和百科知识以及推理知识；且其中语用知识极为关键，是必不可少的背景知识。而知识库的建设作为自然语言理解中一项基础而关键的任务，虽已取得不小的成果，但已有的知识库主要集中在语义知识上，语用知识库的基础研究、构建和实践应用还十分薄弱。这项巨大的系统工程中，包括知识获取的渠道、建构、表示和利用的难题还有待我们在日后的研究中逐一攻破。

当然，本文作为一项初步尝试，样本量、语篇类型以及构成的数据集仍十分有限。限于篇幅，也未能穷尽所有的类型并作详尽描写。但从本文有限的样本中仍能提取到相当数量的复述句，且有大量的复述句是难以基于前人总结的复述现象类型来解释的。这是与以往复述句研究最大的不同，也是未来复述研究中需引起关注的一个重要问题。

致谢

本文的撰写曾得到于东老师的指导，特此感谢。

参考文献

陈龙、饶琪、刘扬2019 汉语词的非字面义的表示与应用， 《中国科学:信息科学》第8期。

陈千、陈夏飞、郭鑫、王素格2018 面向阅读理解的多对一中文文本蕴含问题研究，《中文信息学报》第4期。

戴耀晶2011 句子语用意义的提取，《当代修辞学》第2期。

胡金铭、史晓东、苏劲松、陈毅东2013 引入复述技术的统计机器翻译研究综述，《智能系统学报》第3期。

黄希敏2011 英语词汇语用策略，世界图书出版社。

姜望琪2014 语用推理之我见，《现代外语》第3期。

金天华、姜姗、于东、赵美倩、刘璐2019 中文句法异构蕴含语块标注和边界识别研究，《中文信息学报》第2期。

李新良、袁毓林2013 面向计算的汉语动词蕴涵关系研究和型式库建设，《中国社会科学》第12期。

廖秋忠1986 现代汉语篇章中指同的表达，《中国语文》第2期。

刘松2014 基于全信息的问答系统研究，北京邮电大学博士学位论文。

刘明童、张玉洁、徐金安、陈钰枫2018 开放域上基于深度语义计算的复述模板获取方法，《中文信息学报》第2期。

马彬彬2019 基于神经网络的复述生成方法研究，北京交通大学硕士学位论文。

倪盛俭2013 文本蕴涵研究现状和发展趋势，《云南民族大学学报(哲学社会科学版)》第4期。

庞杨、张绍杰2012 词汇同义关系的认知关联解读，《外语学刊》第4期。

庞杨2015 词汇同义关系的语境依赖性与构建机制实验研究，《外语学刊》第4期。

庞杨2016 词汇同义关系的在线构建与认知相似性，《外语教学》第1期。

冉永平2008 词汇语用信息的临时性及语境构建，《外语教学》第6期。

冉永平2012 词汇语用探新，外语教学与研究出版社。

任函、冯文贺、刘茂福、万菁2017 基于语言现象的文本蕴涵识别，《中文信息学报》第1期。

王宁2017 论词的语言意义的特性，《北京师范大学学报(社会科学版)》第2期。

郗亚辉2016 产品评论挖掘中特征同义词的识别，《中文信息学报》第4期。

邢福义1966 《汉语语法学》，东北师范大学出版社。

徐盛桓1966 话语的含意性，《外语研究》第3期。

杨毅、冯文贺2018 汉语小句的俄语对应单位研究，《中文信息学报》第1期。

姚振宇2015 基于复述的机器翻译系统融合方法研究，哈尔滨工业大学硕士学位论文。

叶花2006 高考语文现代文阅读理解的语用学研究，暨南大学硕士学位论文。

叶慧君2018 汉语词义在线理解的词汇语用学研究，外语教学与研究出版社。

张绍杰、杨忠1993 论语用等同，《现代外语》第2期。

赵世奇、刘挺、李生2009 复述技术研究，《软件学报》第8期。

赵世奇2009 基于统计的复述获取与生成技术研究，哈尔滨工业大学博士学位论文。

宗世海2000 汉语话语中误解的类型及其因由，广东外语外贸大学博士学位论文。

宗世海2002 含意理论在对外汉语教学中的运用，《语言教学与研究》第3期。

Austin J.L. 1997. *How to do Things with Word*. Harvard: Harvard University Press.

Carston,R. 1981. Linguistic meaning, communicated meaning and cognitive pragmatics. *Mind and Language*, 2002,17/1&2: 127-148.

Channell, Joanna. 1994. *Vague Language*. Oxford: Oxford University Press.

Glickman, Oren, Eyal Shnarch and Ido Dagan. 2006. Lexical Reference: a Semantic MatchingSubtask. *Proceedings of the 2006 conference on Empirical Methods in Natural Language Processing*.

Grice,P. 1989. *Studies in the Way of Words*. Cambridge, MA: Harvard University Press.

Levinson. 2000. *Presumptive Meanings*. Massachusetts: MIT Press.

Liu Hongzhe, Bao Hong, Xu de. 2011. Concept Vector for Similarity Measurement based on Hierarchical Domain Structure. *Computing and Informatics*, (30):1001-1021.

R. Barzilay and K. R. Mc Keown. 2001. Extracting Paraphrases from a Parallel Corpus. *Proceedings of ACL*, 50-57.

Wilson, D 2004. Relevance, word meaning and communication: The past, present and future of lexical. *Modern Foreign Languages*, 103/1:1-13.

Yule, G. 1997. *Pragmatics*. Oxford: Oxford University Press.

面向汉语作为第二语言学习的个性化语法纠错

张生盛[1,3], 庞桂娜[2,3], 杨麟儿[2,3], 王辰成[4,3], 杜永萍[4], 杨尔弘[3], 黄雅平[1]
[1]北京交通大学，计算机与信息技术学院
[2]北京语言大学，信息科学学院
[3]北京语言大学，语言资源高精尖创新中心
[4]北京工业大学，信息学部

摘要

语法纠错任务旨在通过自然语言处理技术自动检测并纠正文本中的语序、拼写等语法错误。当前许多针对汉语的语法纠错方法已取得较好的效果，但往往忽略了学习者的个性化特征，如二语等级、母语背景等。因此，本文面向汉语作为第二语言的学习者，提出个性化语法纠错，对不同特征的学习者所犯的错误分别进行纠正，并构建了不同领域汉语学习者的数据集进行实验。实验结果表明，将语法纠错模型适应到学习者的各个领域后，性能得到明显提升。

关键词： 语法纠错 ； 个性化 ； 汉语学习者 ； 领域适应

Personalizing Grammatical Error Correction for Chinese as a Second Language

Shengsheng Zhang[1,3], Guina Pang[2,3], Liner Yang[2,3], Chencheng Wang[4,3], Yongping Du[4], Erhong Yang[3], Yaping Huang[1]
[1]Beijing Jiaotong University, School of Computer and Information Technology
[2]Beijing Language and Culture University, School of Information Science
[3]Beijing Language and Culture University,
Beijing Advanced Innovation Center for Language Resources
[4]Beijing University of Technology, Faculty of Information Technology

Abstract

The Grammatical Error Correction (GEC) task is to realize automatic error detection and correction of text through natural language processing technology, such as word order, spelling and other grammatical errors. Many existing Chinese GEC methods have achieved good results, but these methods have not taken into account the characteristics of learners, such as level, native language and so on. Therefore, this paper proposes to personalize the GEC model to the characteristics of Chinese as a Second Language (CSL) learners and correct the mistakes made by CSL learners with different characteristics. To verify our method, we construct domain adaptation datasets. Experiment results on the domain adaptation datasets demonstrate that the performance of the GEC model is greatly improved after adapting to various domains of CSL learners.

Keywords: Grammatical Error Correction , Personalizing , Chinese as a Second Language Learners , Domain Adaptation

1 引言

语法纠错(Grammatical Error Correction, GEC)任务旨在自动检测文本中存在的标点、语序等语法错误，识别错误的类型并对错误进行自动改正。语法纠错系统的输入是一个可能有语法错误的句子，输出是其相应的修改后的句子。如图1所示，第一行表示系统的输入，第二行表示系统的输出，其中加粗部分表示修改的内容。随着人工智能越来越深入地影响人们的日常生活，而自然语言处理作为语言学和计算机学科完美融合的一个学科，在人工智能领域扮演着重要的角色。语法纠错任务是自然语言处理领域的一个重要分支，不论是在实际生活还是科研领域都有着举足轻重的地位，吸引了大量的研究者。

输入：虽然不能完整的解决代沟的问题，但能减少代沟之宽度。
输出：虽然不能**彻底地**解决代沟的问题，但能**缩短**代沟之**距离**。

图 1. 语法纠错系统的输入输出示例

Lee and Seneff (2008)发现，二语学习者犯的语法错误经常受他们的母语因素的影响。例如，母语是日语且把英语作为第二语言的学习者常滥用冠词和介词。对母语是日语的二语学习者常犯的这类错误建模，可以有效地提高语言学习系统的性能。许多基于深度学习的方法 (Chollampatt and Ng, 2018; Junczys-Dowmunt et al., 2018)将语法纠错视为序列到序列(seq2seq)的任务，因此神经机器翻译(Neural Machine Translation, NMT)的方法被成功地运用到语法纠错中，通过将一个错误的句子翻译为正确的句子来实现纠错，并且在一般的领域获得了很好的性能。但是这些基于seq2seq的方法在特定的领域并不能获得鲁棒的性能，其主要原因是并未对特定的领域建模从而出现了领域漂移的现象。例如，Nadejde and Tetreault (2019)使用一般领域训练的模型在特定领域做测试，发现性能明显下降。进而他们将语法纠错模型适应到英语作为第二语言的学习者的母语和等级上，发现语法纠错系统的性能得到了明显的提升。但是，当前针对汉语的语法纠错方法 (Ren et al., 2018; Fu et al., 2018; Zhou et al., 2018; Zhao and Wang, 2020)都集中在一般的领域，并未对特定的领域建模。由于汉语语法与英语语法不同，汉语水平考试的等级与英语水平考试的等级评判标准不一致，二者之间没有直接的联系，故本文面向汉语作为第二语言的学习者，提出个性化语法纠错，通过迁移学习方法将一般的语法纠错系统适应到汉语学习者不同的领域，如汉语学习者的等级、母语等，并对不同等级、不同母语的汉语学习者犯的错误分别进行纠正。

为验证提出的方法是否合理，首先，本文选择汉语学习者的等级和母语作为领域适应的设置，并构建了不同领域汉语学习者的数据集。然后，本文将语法纠错任务视为翻译任务，通过将一个错误的句子翻译为正确的句子实现纠错，并选择基于Transformer增强架构的中文语法纠错模型 (王辰成et al., 2019)作为实验模型。最后，在不同领域的数据集上展开了实验，将一般的语法纠错系统适应到相应的领域，并对不同领域的汉语学习者所犯的错误分别进行纠错。实验结果表明，语法纠错模型适应到学习者的各个领域后，纠错性能得到显著的提升。其中，学习者等级领域适应模型、母语领域适应模型以及母语-等级领域适应模型分别比基线模型高出1.92、1.73、1.76个百分点。

本文的主要贡献如下：

1) 首次提出面向汉语作为第二语言学习的个性化语法纠错，对不同等级、不同母语的汉语学习者分别进行纠错；

2) 构建了不同领域汉语学习者的数据集，用来训练和测试语法纠错模型适应到汉语学习者不同领域后的性能；

3) 将语法纠错模型适应到汉语学习者不同的领域后，纠错性能得到显著提升，整体实验结果均超越基线模型，验证了提出的方法的合理性。

论文的整体结构：第一节是引言；第二节介绍了使用的语法纠错模型和领域适应的方法；第三节是详细的实验设置；第四节给出了实验细节和各个模型对样例的纠错结果；第五节是相关工作，主要介绍了与本文有关的工作；最后是论文的总结部分。

2 方法

在给定一个长度为M的错误句子$X = \{x_1, ..., x_M\}$和一个学习者的领域d后，基于神经机器翻译的语法纠错模型使用神经网络对输出句子$Y = \{y_1, ..., y_N\}$的条件概率建模，如公式1：

$$p(Y|X, d; \theta) = \prod_{t=1}^{N} p(y_t|y_{1:t-1}, x_{1:M}, d; \theta), \tag{1}$$

其中θ是模型的参数。根据Madotto et al. (2019)的工作，首先将模型的参数θ适应到学习者的领域d，然后使用错误的句子对输出句子的条件概率建模，如公式2：

$$p(Y|X; \theta_d) = \prod_{t=1}^{N} p(y_t|y_{1:t-1}, x_{1:M}; \theta_d), \tag{2}$$

其中θ_d是对学习者的领域d建模以后的参数。学习者的领域有多种定义的方式，如等级、母语、母语和等级的组合等。

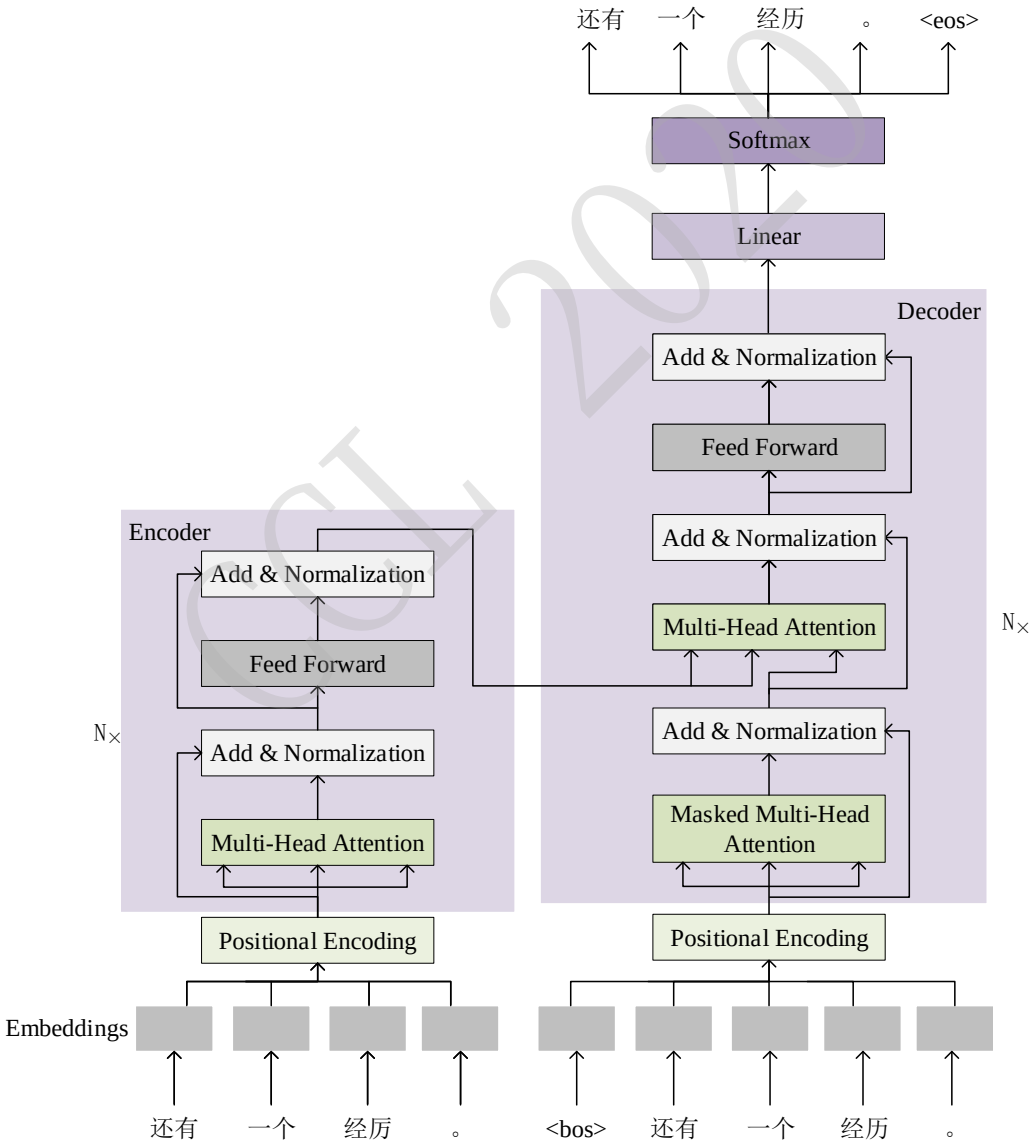


图 2. Transformer语法纠错模型

本文采用王辰成et al. (2019)实现的基于Transformer增强架构的中文语法纠错模型，该模型不仅可以捕获丰富的语义信息，还可以减少因为网络过深出现的梯度消失问题。

Transformer (Vaswani et al., 2017)模型是基于多头注意力机制的seq2seq生成模型，如图2所示，它由编码器(Encoder)和解码器(Decoder)组成。其中编码器由N个相同的模块构成，即图2左侧部分，每个模块由两个网络层构成，分别是多头自注意力机制和全连接的前馈网络，两者之间都使用了归一化和残差连接。解码器同样是由N个相同的模块构成，即图2右侧部分，与编码器不同的是解码器中包含使用编码器输出进行运算的多头注意力层。编码器的作用是将输入序列编码为高维隐含语义向量，解码器根据上一时间步的输出，解码隐含语义向量作为当前时间步的输出，每个时间步的输出对应序列中的一个元素，所有时间步的输出拼接在一起作为最终的输出序列。王辰成et al. (2019)将各个模块的输出动态地结合到一起，以此来增强模型对语义信息的表达能力。

图 3. 语法纠错系统领域适应框架

为实现个性化语法纠错，本文采用迁移学习方法，将一般领域的语法纠错模型适应到学习者的特征领域。采用迁移学习方法的主要原因是一般领域的数据较多，学习者特征领域的数据较少，一般领域的数据与学习者特征领域的数据可以共享模型的参数，更好地帮助模型适应到学习者的特征领域。具体做法是：首先使用一般领域的数据对语法纠错模型做预训练，再利用学习者特征领域的数据对模型进行微调，使其适应到相应的领域，整体的框架如图3所示。

3 实验设置

3.1 数据集

本文中首先在Lang-8[0]数据集上对模型进行预训练，然后使用HSK[1]作为领域适应的数据集。两个数据集均由汉语作为第二语言的学习者书写，并由母语是汉语的人进行了纠错。我们从两个数据集中抽取平行句对，去掉未修改的句对。使用jieba[2]分词工具对所有句子进行分词，并且运用字节对编码算法(Byte Pair Encoding, BPE)[3] (Sennrich et al., 2016)进一步限制词表大小，以缓解罕见词和未登录词(Out of Vocabulary, OOV)的问题。

原始句子	好像我的疲劳感也飞过去了。
分词后的句子	好像 我 的 疲劳感 也 飞过去 了 。
BPE拆分后的句子	好像 我 的 疲@@ 劳@@ 感 也 飞过去 了 。

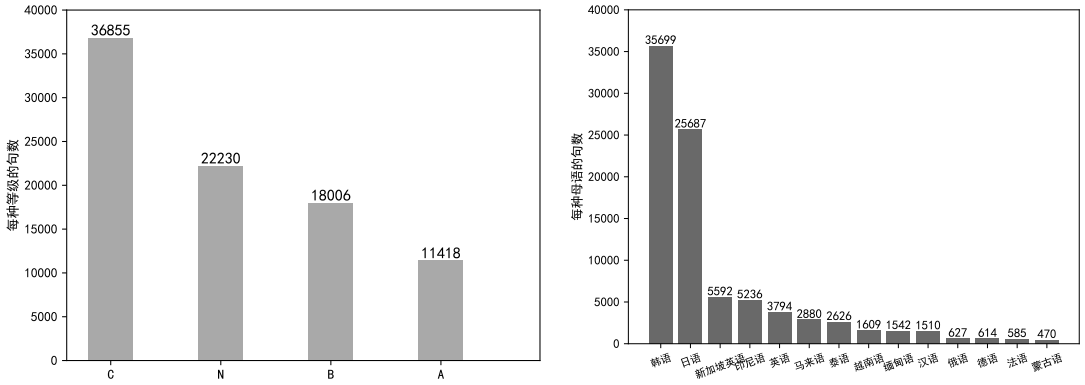
表 1. 一个句子经过jieba分词和BPE的示例

数据集	句子数目	原始句子的词语	修改后句子的词语
Lang-8	1,095,985	14,352,734	15,090,960
HSK	88,509	1,783,487	1,768,823

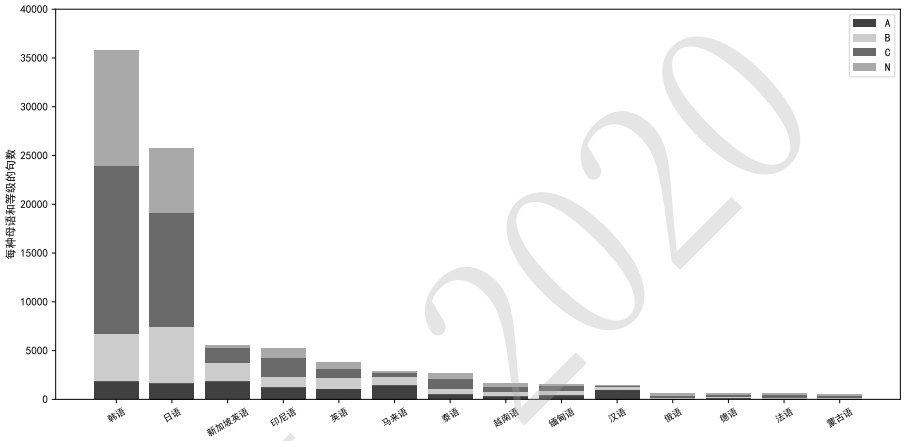
表 2. 数据集详情

[0] https://lang-8.com
[1] http://hsk.blcu.edu.cn/
[2] https://github.com/fxsjy/jieba
[3] https://github.com/rsennrich/subword-nmt

如表1所示，给出了一个句子经过分词和BPE以后的结果，其中‘@@’符号表示当前单元与下一个单元同属一个词语。经过处理后两个数据集的详细情况如表2所示。



(a) 汉语学习者等级在HSK中的分布情况　(b) 汉语学习者母语在HSK中的分布情况



(c) 汉语学习者等级和母语在HSK中的分布情况

图 4. HSK中各个等级、母语以及等级和母语的句子数分布

我们在HSK数据集上研究个性化语法纠错，该数据集由14种不同母语和4种不同等级的汉语学习者所写的考试作文组成。我们根据学习者的母语和等级信息抽取了平行句对，其中各个等级所包含的句子数如图4(a)所示，各个母语所包含的句子数如图4(b)所示，各个等级和母语的句子数如图4(c)所示。

3.2 超参数

在实验中，编码器词嵌入矩阵和解码器词嵌入矩阵维度为512 ，解码器的输入和输出词嵌入矩阵共享权重。编码器和解码器各包含6个模块，每个模块的多头注意力层有8个注意力头，前馈层的维度大小为2048。优化器使用Adam，动量设置为(0.9, 0.98)，warm-up为4000，学习率的更新策略为初始值是1×10^{-7}，在前4000步的训练中，线性增长到5×10^{-4}，之后逐步指数下降到训练结束，Dropout为0.3，柱搜索的大小为12，最大的token数为4000。在预训练阶段，我们根据Ren et al. (2018)的方法，随机从Lang-8中抽取5000个句对作为验证集，并选取验证集上最优的模型作为最终预训练的模型。在微调阶段，选取验证集上最优的前5个模型的权重并计算权重的平均值作为领域适应模型的权重。

3.3 基线模型

本文的基线模型有两个，分别为：

1) 无微调 (Nadejde and Tetreault, 2019)：该方法直接使用预训练的模型对各个领域测试集中的句子进行纠错；

2) 随机 (Nadejde and Tetreault, 2019)：该方法随机地从HSK数据集中抽取与各个领域相同数量的训练集和开发集，并使用这些数据对预训练的模型进行微调，然后对各个领域的测试集中的句子进行纠错。领域适应的各个模型超越这个基线模型可以帮助我们验证语法纠错模型性能的提升不仅是因为模型适应到HSK数据集上，还因为对各个领域成功地建模。

3.4 评价指标

正如王辰成et al. (2019)提到的应该重点关注模型对错误编辑的准确性而非编辑的数量，所以为了评价语法纠错模型的性能，我们采用MaxMatch(M^2)工具包计算$F_{0.5}$分数，根据$F_{0.5}$的大小判断模型的性能。计算$F_{0.5}$需要有$m2$格式的文件，它是根据原句和修改句来生成最佳的黄金编辑集合。因此，我们使用ERRANT[4] 工具包制作了各个测试集$m2$格式的文件。

4 实验结果

为了验证提出的方法是否合理，我们构建了各个领域适应的数据集，根据Nadejde and Tetreault (2019)的工作中的实验设置，我们将各个领域适应的数据集按照8:1:2的比例划分为训练集、开发集和测试集。

4.1 针对汉语学习者等级的语法纠错模型

HSK数据集将汉语学习者的水平分为A、B、C和N（N表示无）四个等级，我们为每个等级随机抽取11,000个平行句对；为随机基线模型在不考虑等级的条件下随机抽取8,000个平行句对作为训练集，1,000个平行句对作为开发集。

等级	无微调	随机	适应到等级
A	18.53	38.45	**42.04**
B	22.44	41.02	**43.19**
C	24.56	44.76	**44.91**
N	23.62	45.84	**47.63**
平均	22.29	42.52	**44.44**

表 3. 不同等级上的结果

实验结果如表3所示：我们的结果在等级A上超出随机基线模型3.59个百分点，从图4(a)可以看出，等级A的句子是HSK中数量最少的，却是提升最高的，这表明了个性化语法纠错是合理且有效的。在等级C上，相比随机基线模型我们的方法提升了0.15个百分点，从图4(a)可以看出，等级C的句子是HSK中数量最多的，所以在不考虑等级的情况下随机抽到的句对中等级C的数量也是最多的，这很好地解释了在等级C上的提升是最小的。从整体结果来看，我们的方法比随机基线模型高出1.92个百分点。

4.2 针对汉语学习者母语的语法纠错模型

母语	无微调	随机	适应到母语
韩语	23.84	44.25	**45.58**
日语	28.31	46.19	**46.47**
新加坡英语	14.74	30.25	**35.23**
印尼语	18.61	37.93	**38.28**
平均	21.38	39.66	**41.39**

表 4. 不同母语上的结果

HSK数据集由14种母语的汉语学习者书写，本实验选用其中数据最多的4种母语：韩语、日语、新加坡英语、印尼语。我们为这4种不同的母语分别随机抽取4,950个平行句对；为随机

[4] https://github.com/chrisjbryant/errant

基线模型在不考虑母语的条件下随机抽取3,600个平行句对作为训练集，450个平行句对作为开发集。

实验结果如表4所示：我们的方法在新加坡英语上比随机基线模型高出4.98个百分点，超越了其他母语。此外，在新加坡英语上，从无微调基线模型到随机基线模型的提升为15.51个百分点，低于其他母语。以上结果说明领域适应模型性能的提升不仅是因为将模型适应到HSK数据集上，还因为对学习者的领域进行了建模。从整体结果来看，我们的方法比随机基线模型超出1.73个百分点。

4.3 针对汉语学习者母语和等级的语法纠错模型

HSK数据集中包含4种等级和14种母语的组合，共有56种母语-等级组合的情况，本实验选用其中数据最多的5种母语-等级组合：韩语-C、韩语-N、日语-B、日语-C、日语-N。我们为这5种不同的母语-等级组合分别随机抽取4,950个平行句对；为韩语和日语每种语言随机抽取3,600个平行句对作为训练集，450个平行句对作为开发集；为B、C、N每种等级随机抽取3,600个平行句对作为训练集，450个平行句对作为开发集；为随机基线模型在不考虑等级、母语的条件下随机抽3,600个平行句对作为训练集，450个平行句对作为开发集。

	韩语-C	韩语-N	日语-B	日语-C	日语-N	平均
无微调	23.43	25.00	24.01	23.78	25.11	24.27
随机	44.67	44.64	43.13	44.71	46.37	44.70
等级	44.50	47.80	42.33	45.72	44.66	45.00
母语	44.92	46.69	**43.96**	45.27	46.83	45.53
母语-等级	**45.83**	**48.26**	43.34	**46.62**	**48.26**	**46.46**

表 5. 不同母语-等级上的结果

实验结果如表5所示：我们的方法在韩语-N上比随机基线模型提升了3.62个百分点，比韩语提升了1.57个百分点，超越了其他母语-等级，是母语-等级与随机、母语相比提升效果最明显的组合；在日语-N上比等级N提升了3.60个百分点，是母语-等级与等级相比提升效果最明显的组合。从整体结果来看，适应到母语-等级的模型比随机基线模型高出1.76个百分点，比适应到等级的模型高出1.46个百分点，比适应到母语的模型高出0.93个百分点。

4.4 实验结果分析

原始句子	第一当一个人在公共场所内抽烟当他抽完烟，如果他找不到丢烟蒂的地方，他可能会随手丢掉，
标准答案	第一，**一个人在公共场所内抽烟**，当他抽完烟**后**，如果他找不到丢烟蒂的地方，他可能会随手丢掉。
无微调	第一，**当一个人在公共场所内抽烟时，他抽完烟**，如果他找不到丢**心烦**的地方，他可能会随手丢掉，
随机	第一，**当一个人在公共场所内抽烟时，他抽完烟**，如果他找不到丢**抬水**的地方，他可能会随手丢掉。
适应到等级A	第一，**当一个人在公共场所内抽烟时，他抽完烟，如果他找不到丢的地方**，他可能会随手丢掉。

表 6. 不同模型的纠错结果

从表3可知，相比随机基线模型，领域适应的模型性能在等级A上的提升最明显。因此本文针对等级A的纠错样例进行分析，从而直观地反映将一般的语法纠错模型适应到相应领域后纠错能力的表现。如表6所示，给出了各个模型对一个有语法错误句子的修改结果，其中加粗部分表示修改内容。从表6观察发现相比随机基线模型，领域适应的模型修改更符合汉语用语习惯。

因此，将语法纠错模型适应到汉语学习者的特征领域以后，模型纠错的结果更接近汉语的习惯表达。

5 相关工作

传统的语法纠错方法可以分为两类：1) 基于规则的 (Bustamante and León, 1996)，这些方法只关注文本中的几种错误类型；2) 基于统计机器翻译的 (Brockett et al., 2006)，这些方法将语法纠错任务视为翻译任务，并使用统计器翻译方法进行纠错，极大地提升了语法纠错系统的性能。

随着深度学习的发展，许多序列到序列的方法成功地应用到语法纠错中，这些方法将语法纠错视为一般的seq2seq任务，即系统的输入是一个序列，输出也是一个序列。Yuan and Briscoe (2016)第一次将神经机器翻译模型应用到语法纠错任务中，他们使用双向的递归神经网络编码器和一个基于注意力的解码器对错误进行纠错，性能超越了基于统计机器翻译的纠错模型。Ji et al. (2017)使用嵌套注意力神经混合模型纠错，该模型通过合并单词和字符级别的信息来纠正两种类型的错误。Chollampatt and Ng (2018)使用一个多层卷积编码-解码器神经网络模型进行纠错，并结合语言模型对纠错的结果进行重排序。Junczys-Dowmunt et al. (2018)将语法纠错视作低资源的机器翻译任务，并使用Transformer作为纠错模型。

针对英语的语法纠错方法层出不穷，并且取得了很好的效果。但汉语语法纠错方兴未艾，自中国计算机协会举办的国际自然语言处理与中文计算会议(NLPCC)在2018年加入了汉语语法纠错评测任务后，出现了许多汉语语法纠错方法。Fu et al. (2018)采用简单到复杂的分阶段纠错方法，使用语言模型纠正简单的错误，字、词级的Transformer模型纠正复杂的错误。Zhou et al. (2018)使用多个模型纠错，分别是基于规则、统计和神经网络，通过模型组合的方式得到最终的纠错结果。Ren et al. (2018)使用基于卷积神经网络的seq2seq模型纠错，还采用了subword算法 (Sennrich et al., 2016)来缩小词表和缓解未登录词的问题。王辰成et al. (2019)提出了基于Transformer增强架构的中文语法纠错模型，该模型使用动态残差结构结合不同神经模块的输出来增强模型捕获语义信息的能力。Zhao and Wang (2020)采用动态掩码的方式提高模型的纠错性能，在训练步骤中动态地向原始的句子加入掩码来增加更多的平行句对。但以上这些汉语语法纠错方法均没有对汉语学习者的个性化特征进行建模。

周小兵et al. (2007)在对汉语作为第二语言的学习者的教学研究中发现，母语迁移是造成二语学习者语法偏误的一项主要原因。如有的汉语学习者会写“*我见面我的老师。”这样的错句，因为法语、韩语、日语、越南语等语言中“见面”可以带宾语，但是汉语中“见面”后面是不可以带宾语的。此外，Swan and Smith (2001)在对二语学习者的教学研究发现不同母语写作者会犯不同类型的错误，他们将其中的某些错误归因于语言之间的“转移” 或“干扰” ，即母语的“负迁移”。因此，已有许多针对二语学习者的研究，如Rozovskaya and Roth (2011)使用朴素贝叶斯对不同母语的英语学习者所犯的介词错误进行纠错；Mizumoto et al. (2011)发现当训练数据和测试数据用相同的母语时，语法纠错系统的表现会更好；Nadejde and Tetreault (2019)针对英语作为第二语言的学习者提出了个性化语法纠错，发现将语法纠错模型适应到学习者的不同特征时表现会更好。

6 总结

本文针对汉语作为第二语言的学习者，首次提出了个性化语法纠错。并将语法纠错任务视为翻译任务，使用基于Transformer增强架构的中文语法纠错模型对错误进行纠正。为了验证提出的方法的合理性，构建了不同领域的数据集，并使用迁移学习方法将语法纠错模型适应到学习者不同的领域，实现个性化语法纠错。各个领域测试集上的平均结果都超越了未做领域适应的基线模型，表明语法纠错系统在对学习者的特征建模以后可以有效地改进纠错的效果。

致谢

感谢北京语言大学语言资源高精尖创新中心项目(TYZ19005)和国家语委信息化项目(ZDI135-105)对本研究的支持。

参考文献

Chris Brockett, William B. Dolan, and Michael Gamon. 2006. Correcting ESL errors using phrasal SMT techniques. In *Proceedings of the 21st International Conference on Computational Linguistics*

and 44th Annual Meeting of the Association for Computational Linguistics, pages 249–256, Sydney, Australia, July. Association for Computational Linguistics.

Flora Ramírez Bustamante and Fernando Sánchez León. 1996. GramCheck: A grammar and style checker. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*.

Shamil Chollampatt and Hwee Tou Ng. 2018. A multilayer convolutional encoder-decoder neural network for grammatical error correction. *CoRR*, abs/1801.08831.

Kai Fu, Jin Huang, and Yitao Duan. 2018. Youdao's winning solution to the NLPCC-2018 task 2 challenge: A neural machine translation approach to chinese grammatical error correction. In Min Zhang, Vincent Ng, Dongyan Zhao, Sujian Li, and Hongying Zan, editors, *Natural Language Processing and Chinese Computing - 7th CCF International Conference, NLPCC 2018, Hohhot, China, August 26-30, 2018, Proceedings, Part I*, volume 11108 of *Lecture Notes in Computer Science*, pages 341–350. Springer.

Jianshu Ji, Qinlong Wang, Kristina Toutanova, Yongen Gong, Steven Truong, and Jianfeng Gao. 2017. A nested attention neural hybrid model for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 753–762, Vancouver, Canada, July. Association for Computational Linguistics.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Shubha Guha, and Kenneth Heafield. 2018. Approaching neural grammatical error correction as a low-resource machine translation task. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 595–606, New Orleans, Louisiana, June. Association for Computational Linguistics.

J. Lee and S. Seneff. 2008. An analysis of grammatical errors in non-native speech in english. In *2008 IEEE Spoken Language Technology Workshop*, pages 89–92.

Andrea Madotto, Zhaojiang Lin, Chien-Sheng Wu, and Pascale Fung. 2019. Personalizing dialogue agents via meta-learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5459, Florence, Italy, July. Association for Computational Linguistics.

Tomoya Mizumoto, Mamoru Komachi, Masaaki Nagata, and Yuji Matsumoto. 2011. Mining revision log of language learning SNS for automated Japanese error correction of second language learners. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 147–155, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.

Maria Nadejde and Joel Tetreault. 2019. Personalizing grammatical error correction: Adaptation to proficiency level and l1. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 27–33.

Hongkai Ren, Liner Yang, and Endong Xun. 2018. A sequence to sequence learning for chinese grammatical error correction. In Min Zhang, Vincent Ng, Dongyan Zhao, Sujian Li, and Hongying Zan, editors, *Natural Language Processing and Chinese Computing - 7th CCF International Conference, NLPCC 2018, Hohhot, China, August 26-30, 2018, Proceedings, Part II*, volume 11109 of *Lecture Notes in Computer Science*, pages 401–410. Springer.

Alla Rozovskaya and Dan Roth. 2011. Algorithm selection and model adaptation for ESL correction tasks. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 924–933, Portland, Oregon, USA, June. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August. Association for Computational Linguistics.

M. Swan and B. Smith. 2001. *Learner English: A Teacher's Guide to Interference and Other Problems*. Cambridge handbooks for language teachers. Cambridge University Press.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Zheng Yuan and Ted Briscoe. 2016. Grammatical error correction using neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–386, San Diego, California, June. Association for Computational Linguistics.

Zewei Zhao and Houfeng Wang. 2020. Maskgec: Improving neural grammatical error correction via dynamic masking. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 1226–1233. AAAI Press.

Junpei Zhou, Chen Li, Hengyou Liu, Zuyi Bao, Guangwei Xu, and Linlin Li. 2018. Chinese grammatical error correction using statistical and neural models. In Min Zhang, Vincent Ng, Dongyan Zhao, Sujian Li, and Hongying Zan, editors, *Natural Language Processing and Chinese Computing - 7th CCF International Conference, NLPCC 2018, Hohhot, China, August 26-30, 2018, Proceedings, Part II*, volume 11109 of *Lecture Notes in Computer Science*, pages 117–128. Springer.

周小兵, 朱其智, and 邓小宁. 2007. 外国人学汉语语法偏误研究. 北京语言文化大学出版社.

王辰成, 杨麟儿, 王莹莹, 杜永萍, and 杨尔弘. 2019. 基于Transformer增强架构的中文语法纠错方法. 中国计算语言学大会.

中文问句的形式分类和资源建设

黎江涛[1]**, 饶高琦**[1]
(1. 北京语言大学汉语国际教育研究院，北京100083)
eric_lijiangtao@163.com, raogaoqi@blcu.edu.cn

摘要

本文归纳了问句形式在问句语料筛选中的作用，探索了问句分类必需的形式特征，同时通过人工标注建设了中文问句分类语料库，并在此基础上进行了基于规则和统计的分类实验，通过多轮实验迭代优化特征组合形成特征规则集，为当前问答提供形式上的分类基础。实验中，基于优化特征规则集的有限状态自动机可实现宏平均F1值为0.94；统计机器学习中随机森林模型的分类效果较好，F1值宏平均达到0.98，表明问句形式分类具有相当可行性和准确性。

Formal classification and resource construction of Chinese question

Jiangtao Li[1]**, Gaoqi Rao**[1]
(1. Research Institute of International Chinese Language Education,
Beijing Language and Culture University, Beijing 100083, China)
eric_lijiangtao@163.com, raogaoqi@blcu.edu.cn

Abstract

This paper summarized the role of question forms in question corpus screening, explored the necessary formal features of question classification, and constructed a Chinese question classification corpus by manual tagging. On this basis, this paper have conducted classification experiments based on rules and statistics, and optimized feature combinations to form feature rule sets through multiple rounds of experiments, which provides a formal classification basis for current questions and answers. In the experiment, the finite state machine based on the optimized feature rule set can achieve a macro average F1-score of 0.94; The classification effect of random forest model is better, and the average F1-score reaches 0.98, which indicates that the classification of question forms is feasible and accurate.

1 引言

问句分类的效果直接影响问句理解。传统的中文问题分类主要是根据答案对象的类型划分，如询问人物、地点、时间、数量等，曹志娟（2005）还在此基础上增加疑问词短语分类、问题标准型、特征词分词来增强计算机识别问题的能力的方法，刘朝涛（2008）则进一步将疑问词模式与问题类型对应起来，进行了基于疑问句句型识别的问题理解研究。在这些分类任务中，问句的形式只是作为分类的辅助特征。

第十九届中国计算语言学大会论文集，第107页-第116页，海南，中国，2020年10月30日至11月1日。

实际上，一定的问句形式下的问句类别可以对应一定的问句功能，但这方面的理论在问句理解实践中并没有得到重视；相反，随着数据集的增加，问句覆盖的范围越广，复杂的问句形式特征被当作解决新问题的补丁不断地添进，使得问句分类标准越来越复杂。如果能在问题分类中先提供一个形式分类接口，再按照不同问句形式下对应的问句功能对问句做进一步分类，那么就能在形式上不遗漏任何问句，同时也能在分类过程中根据问句形式定位问句的具体功能。所以在现有问句分类研究基础上，提倡问句的形式分类具有深刻意义。

2 问句的性质

2.1 问句的范围

傅惠钧曾根据“疑”和“问”的组合划分出“有疑有问、有疑无问、无疑有问、无疑无问”四类。很明显“有疑有问”和“无疑无问”均可以明显地判断句子是否为问句，问题就集中到了“有疑无问”和“无疑有问”这两类句子上。

先说“有疑无问”。吕叔湘给出过例句“也许会下雨吧”,表示有传疑但不发问。这个例句后面既可以加上问号标记也可以不加上问号标记，邵静敏根据这种对比指出，两种情况表达的疑问程度是一致的，区别仅仅在于是否发问，即是否要求对方表示态度。所以由此可见，从问答理解的角度来看，回答的前提是存在发问，所以将没有发问意图句子排除在分析目标之外是合理的，这也符合问句提出的预期，即发问-解答。本文也将根据是否有发问意图来区分疑问问句和非疑问问句。

再说到“无疑而问”，学界对这类句子众说纷纭，普遍认同的一个观点是反问句（也叫反诘问句）可以作为“无疑而问”的典型代表，马氏文通中将这类句子的功能称为“传信”，与“传疑”相对。判断这一类句子必须要明确一点：“信疑”皆是从说话人的意图中推断出来的，而不是站在对话的全知视角或是听话人视角。如果“信疑”脱离了说话人意图，那么问句就可能会随着不同的回答而有不同的定性，在疑问句和反问句之间摇摆不定。例如，“谁欠你钱？”，说话人如若想表达“我不欠你钱”的意思该句则是反问句，但如果不考虑说话人的意图，仅考虑该问句的可回答性，也可以说“某某欠了钱”，但这明显已经脱离了说话人想表达的意图。所以“无疑而问”本质上是不含发问意图的句子。对于问句理解来说，如果是在问答系统中，“无疑而问”的问句显然不能成为分析的对象，因为句子本身不存在疑问点，也就无法对问题做出回答；但如果从人机对话的角度来说，“无疑而问”更偏向是一种套着疑问形式的表达方式，这样的句子往往承载着说话人的某些观点、意图，计算机要做的就是要在遵守语用交际原则的情况下回应这些句子，此时的“无疑而问”类句子无疑应该纳入该研究的分析对象。

而本文讨论分析的对象以含有说话者发问意图的问句为主，对不含发问意图的问句只做简单的功能探讨。

2.2 问句的分类

含有说话者发问意图的问句通常又叫疑问句，按照形式上的不同，它们又可以分为四类：是非问、特指问、选择问、正反问。

是非问。结构类似陈述句，一般用升调，句尾一般有“？”，句尾有时兼有语气助词“吗”显化疑问语调，也可以用“啊、哇”，但不可用“呢”。例如：“21世纪人类将要开发月球吗？”

特指问。用疑问代词代替未知部分，常用的疑问代词有“谁、什么、哪儿、怎么、多少”等，句尾有时用“呢”或“啊”，不用“吗”。例如：“这是哪里啊？”

选择问。有并列的若干分句，前后分句常用“是”“还是”相呼应，有时用语气助词“呢”或“啊”，但不用“吗”；另外，选择问中语词助词和连词可以兼有。例如：“是吃西餐还是吃中餐？”

正反问。通常包含否定词“不”“没有”，不采取复句的形式，在谓语中心或补语中用肯定和否定并列形式来提问。具体情况如下表所示：

形式	例句
V/Adj+不+V/Adj	你饿不饿？
V+不+V+X	你吃不吃饭？
V+X+不+V	你吃饭不吃？
V+X+不+V+X	你吃饭不吃饭？
V+否定词（不/不成/否）	你吃饭不？
V+补+V+否定词+补	这饭你吃得了吃不了？

表 1　正反问形式及例句

2.3　问句形式概述

问句形式是判断问句的依据，主要包括语音语调、标点形式、句法格式、特征词。语音语调主要指句子的句调，一般问句的句调均以声调为主。标点形式主要指问号，这是问句的主要形式标记。句法格式指不同问句类型由特定句法单位构成的格式，按照问句类别可以分为是非问句法格式、特指问句法格式、选择问句法格式和正反问句法格式。而特征词是指能够帮助判断问句类别的典型词语，比如特指问的疑问代词，选择问中的“还是”等。

根据承载问句的介质不同，可以从两个方面来说明问句形式的作用和特点。

1.语音问句识别中，本该使用标点停顿的地方用语音停顿替换，表达疑问的标点形式用相应的语音语调替换，因此主要是语音语调、句法格式和特征词等在语音问句识别中起作用。

2.文本问句识别中，标点完全代替语音信息起到停顿、疑问语气的作用，所以标点形式、句法格式和特征词在识别中占据主要地位，其中标点形式尤以问号“？”为主。

所以在问句判别的领域中，语音语调信息与标点信息形成对立，句法格式和特征词两者相互补充，甚至两者还互有交叉，一定情况下还可以相互转换。问号往往就是问句的标志。

3　问句形式在问句分类中的作用

问答系统一般由问题分类、查询扩展、搜索引擎、答案抽取以及答案排序选择多部分组成。问题分类是建构问答系统的重中之重。而对于问题分类而言，目标问句语料的筛选又是问题分类的前提条件。质量高的问句语料可以提高问题分类及后续工作的效率。

通常提取的问句对象都是文章中的对话内容，即引号内的问句，这样做有两个好处：一，可以保证问句提取的自然度，能够最大限度地模拟日常问答；二，为判定问句的意图提供了条件，可以通过问句的上下文来推测说话人的意图从而判别句子是“有疑而问”还是“无疑而问”。而文本问句的形式在上文已提到包括标点形式、句法格式、特征词三类，下面将围绕这三点说明问句形式在问句语料筛选中的作用。

3.1　标点形式

问号是问句的主要标志，根据问句中问号的多少可以把问句大致分为以下两类。

(1)问句中存在多个问号

一般包括两种情况：其一，问句是个连续问句群，例如：“你是谁？你来自哪里？”，此时问句能被分解为若干个单独的问句；其二，问句是选择问句的一种形式变体，如：“你要喝果汁？还是牛奶？”，此时每一个以问号成句的句子不能单独理解，必须将问句群看作一个整体，因为从语义上来说，单独的问句语义并不完整，只有问句群才能够表达完整的意义。

连续问句往往不能成为问句分类分析的典型语料，但它作为问句的组合形式一种，能拆解成若干个问句来理解。而选择问句的形式变体实际上是标点的一种误用，在形式上与连续问句相同，但它在问句语料中也占据一定数量，应算作问句分类分析中的典型语料，否则会使选择问在自然语言中的比例不能得到正确的反映。

(2)问句中只存在一个问号

又可根据问句内部是否存在标点分为两类：一类是组合问句群，另一类是常规问句。汉语中的连续问句可以用逗号连接，以问号煞尾。此时句子并不是单一的问句，而是一个组合式的问句群，不能成为问句分类分析的典型语料。例如：“我是谁，来自哪里，又将会去何处？”

3.2 句法格式

问句中存在一些包含特殊句法格式的句子，这类句子如若按照形式去分析，其问句理解的复杂程度相较其余典型问句要大得多，可细分为以下几类。

(1)“W+呢”类

“W+呢”类又可细分为“NP+呢？”和“VP+呢？”两类。

“NP+呢”在形式上没有明显的问句形式特征，但可以根据其前行句在深层语义上对其进行不同的扩展。例如：

清少爷，你这一向好啊？—好，您老人家呢？（曹286）

“您老人家呢？”可以作“您老人家好不好”、“您老人家怎么样”、“您老人家好吗？”等三种语义理解，且这三种理解分别属于正反问句、特指问句、是非问句。所以可以看出，理解这类问句在语义上需要借助语用信息，在形式上做进一步分类也容易出现分歧。

“VP+呢？”，邵敬敏（1997）将这类问句分成了三种类型：

甲（要是）VP呢

乙（要是）VP，怎么办呢

丙（要是）VP呢、（要是）VP，怎么办呢

形式上来看，“VP+呢”类问句中，甲句型最简洁，乙句型最完整，丙句型为兼备甲乙句型的特点，三种类型都能表达相同的语法意义。另外从功能上来看，“VP+呢”类问句既能表示假设也能表示询问，但无论作何种功能，这类问句的理解同样需要语用信息，且问句往往以甲句型出现。当然，如果考虑到根据深层语义补足原有形式的话，这类问句应是特指问，即根据完整句型乙推出。所以，在问句语料的筛选中，这类问句往往因为其功能的复杂性排除在典型问句的筛选范围之外。

(2)省略疑问成分的问句

一些问句还存在一些缺省疑问成分，但在一定语境下仍旧可以表达疑问。例如两人初次见面时，一方可以用“您是？”提问，意为“您是哪位/您是谁”；对对方的变化感到疑问，可以用“您这是？”提问，意为“您这是怎么了？”。这类句子在省略了疑问词的情况下，以是非问句的形式的存在，但如果根据深层语义补足原有形式，这类句子大多属于特指问，且要理解句子省略了何种疑问词也需要结合语用信息才能说明。所以，在问句语料的筛选中，这类问句往往排除在典型问句的筛选范围之外。

(3)回声问句

回声问是“对话的问题”，具有更多的交际价值，但对于问题本身来说它需要依托于一定的语境才能理解它的含义或补全它的完整问句形式。所以，在问句语料的筛选中，这类问句往往排除在典型问句的筛选范围之外。如下例。

鲁侍萍　　老爷那种绸衬衣不是一共有五件？您要哪一件？

周朴园　　要哪一件？（曹63）

3.3 特征词

不同的问句类型有自己的特征词，这些特征词是判定句子类别的标志。如果特征词出现了错误，就可能影响问句的分类，进而影响问句的理解。主要表现为疑问代词，例如：“在中国有好多人在看摇滚”、“浮云是神马意思”。前者的“好多”带有地域方言色彩，应属疑问词，对应标准式“多少”；后者的“神马”是网络词汇，属于疑问词“什么”一种语言变体。如果在问句理解中，不能对这些形式的问句加以区分，容易在语法结构和语义的分析上造成偏差，最后影响问句的理解。由此可知，在问句语料的筛选中，还需要主要特征词的错写对语料筛选的影响。

所以，标点形式、句法格式、特征词在问答系统的任务中具有举足轻重的作用，规范的问句形式和正确信息同等重要，规范的问句形式是保障问句语料正确性、完整性的基础。

3.4 问句特征选取与特征集构建

根据语言学对是非问、特指问、选择问和正反问的定义，可以进一步将句法格式和特征词细化为疑问格式、语气词、语气副词以及疑问代词四大类，这四大类在具体语料中又可以细分为七个小类：语气词“呢”、语气词“吗”类、疑问代词、语气副词、是非问疑问格式、正反问疑问格式以及选择问疑问格式。注意到在是非问句中，一些句子的显性问句标记过少，不含七小类特征中的任一特征，如是非问“他走了？”，所以为避免无特征匹配是非问句的情况，我们将

增加一类补充特征，即当问句不存在疑问代词、正反问疑问格式和选择问疑问格式任一特征时，默认该句有补充特征，否则没有。

4 问句语料库建设

4.1 数据标注

为测试问句形式对语料的筛选的有效性，同时也为问句数据做进一步的分类，我们从一批小说语料中选取了2400个问句并将这些句子分成三组，每组800句，交由6位语言学专业的研究生两两标注，问句的分类标准主要参照上文的问句定义。是非问、特指问、选择问和正反问分别以数字1、2、3、4表示。一个完整标注的问句如下所示，问句前的数字代表问句的类别。

1：还有其他异常情况吗？（问句标注示例）

经统计，三组在没有对抽取句子进行形式上的筛选之前，一致率分别为：0.855，0.82，0.845，平均一致率达0.84；而经过对抽取的句子按照常规问句形式的筛选，剔除句意理解与语用信息相关的句子后，一致率分别为：0.965，0.943，0.894，平均一致率达到0.934。可见，问句形式有助于提高问句标注的一致率。同时，以上实验也表明，根据问句的语言学特征来判定问句种类并不是一件过于复杂的任务，在此基础上可以继续扩大问句标注规模。

4.2 问句分布情况

经标注及筛选后，我们得到1679句问句。在此基础上，我们还标注了一批形式上较为规整，不依赖语境且可以自足分析的百度知道问句数据集，共2621句。各数据集的问句分布如下所示：

	小说	百度问答	总和
是非	651（38.8%）	527（20.1%）	1178（27.4%）
特指	749（44.6%）	1857（70.9%）	2606（60.6%）
选择	23（1.4%）	40（1.5%）	63（1.5%）
正反	256（15.2%）	197（7.5%）	453（10.5%）
总和	1679	2621	4300

表 2　问句数据分类分布情况

从上表可以看出，特指问在问句中数量与占比均为最高，其次是是非问、正反问以及选择问，这一定程度上也反映了这四类问句在自然语言中的大致分布情况。

此外，在不同数据集上，四类问句的分布也稍有差异。在小说问句中，是非问与特指问占比相当，特指问略高于是非问；而在百度问答问句中，特指问占比超过70%，远远超过是非问的20.1%，这一定程度上呈现了小说问句与百度问答问句的特点，两者既有联系又有区别。百度知道问句是属于百科问答式问句，对概念的提问、事件发生的原因等问句比例较大，致使包含疑问代词的问句较多，也就造成了特指问句在百度问答数据集上分布较多。而小说问句中并没有这种明显的倾向性，使得是非问句与特质问分布较为均匀，同时小说问句的语境也更接近于日常生活场景的问句使用情况。

本资源将向学术界开放使用[0]

4.3 问句特征在语料的分布情况

上文我们整理出了问句的八个小类特征，分别用F1–F8来表示，在语料库中，这些形式特征的计量统计如下：

[0]链接：https://pan.baidu.com/s/1R9se1GPUcQcPLpkzZaGaiw　　提取码：ppux

特征	类别	数量/占比	说明
F1	语气词	253/5.88%	是否有语气词“呢”
F2	语气词	802/18.65%	是否有语气词“吗、么、嘛、吧”
F3	疑问代词	2790/64.88%	什么、如何、哪、哪里、几、谁、啥、为啥、何、何不、为何、为什么、怎么、咋、干吗、多X
F4	疑问格式	1292/30.05%	是非问：能愿动词+语气词
F5	疑问格式	66/1.53%	选择问：X还是X
F6	疑问格式	479/11.14%	正反问：X不X、X不、X没有、X不成
F7	语气副词	96/2.23%	莫非、莫不是、难道、难不成、到底、何必、何须、何妨、何曾、何尝、何不、何苦、究竟、岂
F8	补充特征	1053/24.49%	F3、F5、F6的补充特征

表 3　问句特征分布

表3中，各特征多寡是和不同类型问句占比有关的，部分特征分布情况甚至可以直接反应问句整体的分布情况。如特征F3、F4、F5、F6的占比与四类问句在数据集中的分布情况相当，反映出特指问和是非问在问句中占比较大，选择问相较正反问数量更少。另一方面，疑问格式与疑问代词特征的占比相加大于100%，说明问句分类的结果不是仅由疑问格式决定的，至少存在一个问句包含多个疑问格式或疑问代词的情况，问句分类的复杂性也体现于此。

5　基于问句形式的自动分类

5.1　基于统计机器学习的多特征分类

从问句特征到问句种类的识别实际上是一个从特征到分类的问题。其过程就是把每个问句中能匹配的问句特征转化为可量化的特征向量，最终将特征向量映射到该问句所对应的类别。根据表3的问句特征我们对语料中的问句进行向量化处理，含有指定特征即将特征所在维度的向量值记为1，反之记为0；是非问、特指问、选择问、正反问分别用1、2、3、4表示。如表4。

例：《亮剑》中李云龙求婚对白是第几集啊？								
F1	F2	F3	F4	F5	F6	F7	F8	CLASS
0	0	1	0	0	0	0	0	2

表 4　特征转换示例

在获得多维度向量和其对应的分类标签后就已经进入了根据特征分布进行问句分类的任务。根据以往分类任务经验，本文拟用支持向量机、逻辑回归分类器、贝叶斯分类器、K近邻、决策树以及随机森林等六种机器学习方法来验证问句特征对问句的分类效果。

此外，不同特征数量的选择对问句分类的结果也会有影响。F1至F8等特征近似于从语言学角度对问句形式进行列举，但哪些特征组合能够使得问句分类效果最佳需要进一步实验证明，所以本文将对F1至F8等8个特征做排列组合，共计225种组合结果。

我们再将人工标注的1679句小说问句作为训练语料，后续标注的2621句百度知道问句作为测试语料，将机器学习方法与特征组合结果结合后，下文将从多角度来分析模型的分类效果。

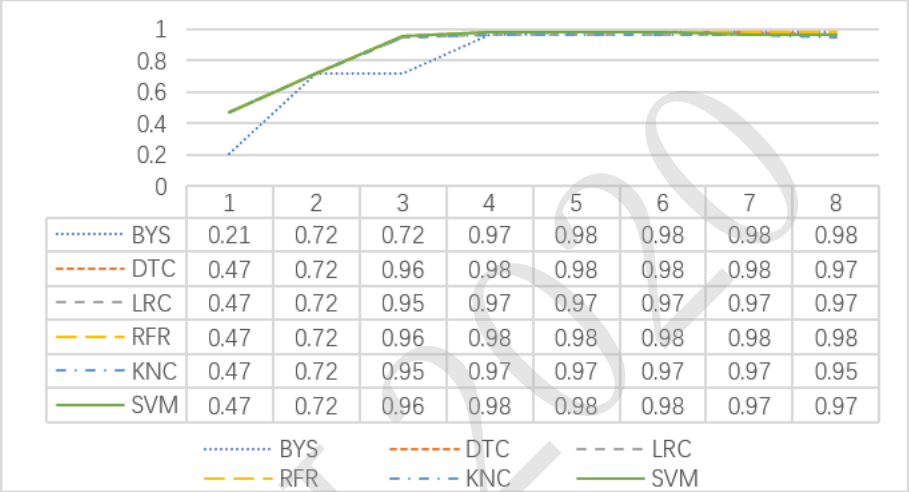
5.2　基于形式特征集的有限状态自动机

不同问句特征对于问句分类判定的贡献不同。根据表3我们可以把特征的覆盖率作为问句特征对问句分类的贡献程度，便有如下排序：语气词=疑问格式¿疑问代词¿其他。那么基于此，

我们可以让贡献大的问句特征优先参与问句判定，而问句特征无法覆盖的问句可以归入形式最多样的是非问，这样问句分类就是在一个有限规则内进行，只要输入一个问句，必定可以输出问句所属的类别。这样就完成了基于形式特征集的有限状态自动机构建准备。

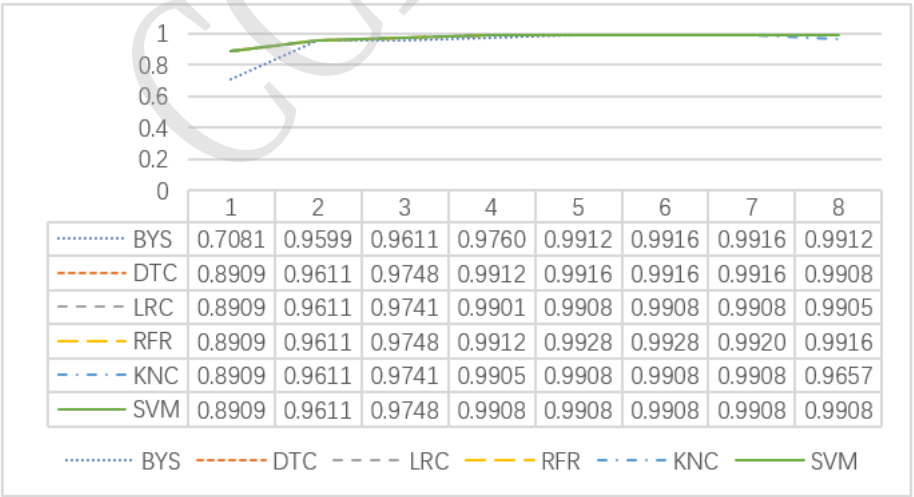
5.3 实验结果

由于问句类别包含四类，我们主要从宏观的角度来分析模型随特征数量变化的情况,即通过不同模型分类的F1值宏平均和微平均分析问句分类整体的优劣(如下两图)。考虑到在某一特征数量下，存在不同特征组合影响分类结果准确性的情况，我们只选取某一特征数量下最好的结果作为比较对象。



	1	2	3	4	5	6	7	8
BYS	0.21	0.72	0.72	0.97	0.98	0.98	0.98	0.98
DTC	0.47	0.72	0.96	0.98	0.98	0.98	0.98	0.97
LRC	0.47	0.72	0.95	0.97	0.97	0.97	0.97	0.97
RFR	0.47	0.72	0.96	0.98	0.98	0.98	0.98	0.98
KNC	0.47	0.72	0.95	0.97	0.97	0.97	0.97	0.95
SVM	0.47	0.72	0.96	0.98	0.98	0.98	0.97	0.97

图 1　各模型F1值在特征数量上的宏平均



	1	2	3	4	5	6	7	8
BYS	0.7081	0.9599	0.9611	0.9760	0.9912	0.9916	0.9916	0.9912
DTC	0.8909	0.9611	0.9748	0.9912	0.9916	0.9916	0.9916	0.9908
LRC	0.8909	0.9611	0.9741	0.9901	0.9908	0.9908	0.9908	0.9905
RFR	0.8909	0.9611	0.9748	0.9912	0.9928	0.9928	0.9920	0.9916
KNC	0.8909	0.9611	0.9741	0.9905	0.9908	0.9908	0.9908	0.9657
SVM	0.8909	0.9611	0.9748	0.9908	0.9908	0.9908	0.9908	0.9908

图 2　各模型F1值在特征数量上的微平均

通过F1值宏平均以及微平均的筛选，得出随机森林模型在特征数量为5时，分类模型的F1值宏平均和微平均达到最高值，分别是0.99和0.98。此时选取的特征分别是F2、F3、F4、F5、F6，即语气词“吗、么、嘛”，是非问疑问格式，选择问疑问格式，正反问疑问格式以及疑问代词。随后，我们将百度知道的2621条问句作为实验对象，采用有限状态自动机分类和随机森林模型分类的效果如表5所示:

分类方法		精确率	召回率	F1 值
有限状态自动机	是非	0.96	0.87	0.91
	特指	0.95	0.99	0.97
	选择	0.93	0.95	0.94
	正反	0.99	0.88	0.93
	F1 值宏平均	0.94		
	F1 值微平均	0.96		
随机森林	是非	0.99	0.99	0.99
	特指	1.00	0.99	1.00
	选择	0.97	0.95	0.96
	正反	0.96	0.99	0.98
	F1 值宏平均	0.98		
	F1 值微平均	0.99		

表 5　随机森林模型和有限状态自动机分类结果

从模型整体效果来看，随机森林的F1值宏平均和微平均相较有限状态自动机的分类结果提高了0.04和0.03。这一方面说明了有限状态自动机分类的方法对问句分类也有较好的效果，通过特定的问句规则可以有效覆盖大多数问句，但这种方法往往会出现召回率偏低的情况，无法处理一些组合特征；另一方面也说明了随机森林模型在进行问句分类过程中具有更好的分类效果。

从各个问句小类的分类结果来看，特指问的F1值在两种分类方法中均为最佳，但在是非问、正反问中，有限状态自动机的F1值却偏差随机森林颇多，体现了是非问句、正反问句的判定受形式特征的多样性明显，单一的问句特征不足以覆盖大多数此类问句；而对于正反问句来说，有限状态自动机的方法在精确率上高于随机森林模型，说明正反问的问句形式特征对正反问的判定具有较强的作用，但在召回率上低于随机森林模型，与是非问情况相同，也体现了正反问形式特征的多样性。

5.4　随机森林模型错例分析

按照错判的类别分为以下典型几类：

例句：有谁能帮忙解释一下，吴尊拍这张照片的这时候在干吗?

上述句子是特指问句，却被错判为是非问句。究其原因是“干吗”作为疑问代词，词中含有“吗”字，使得模型误以为含有是非问特征词，加之语气词“吗”属于强形式特征，模型会倾向于将问句分为是非问。

例句：听说有位明星自杀了真的假的?

上述句子是选择问句，却被错判为是非问句。这是由于选择问句的形式不能覆盖原问句形式所致。选择问句中最典型的疑问格式是“X还是X”，但也存在选择并列的情况，如上句。并列的成分可以是谓词性成分也可以是体词性成分，但不论何种，并列成分在结构上总存在一定的相似性。也正由于这个原因，这类问句形式上难以量化，本实验的模型无法对此类问句的识别效果有限。

例句：韩庚什么的，没上09央视春晚吧?

上述句子是是非问句，却被错判为特指问句。这是由于原是非问句缺少明显的是非问形式特征，但却存在次强形式特征疑问代词，使得模型倾向于将原问句判断为特指问。从另一个角度来说，上述问句的疑问代词“什么”并不是疑问点，而是表示虚指，要正确对此类问句分类必须分清句中的疑问代词是否表示疑问。

例句：可最近心情又是不好，吃药都没作用啦，难道说还是抑郁症?

上述句子是是非问句，却被错判为选择问句。这是由于句中出现了选择问的强形式特征，但“还是”前后连接的并不是选择的对象。结合前文中选择问句错判的例句，可以得出对于选择问问句，精确率较其他分类低，是由于连词“还是”作为选择问的典型特征易与状中结构“还是”混淆，召回率低则是选择问存在不易归纳的问句形式所致。

例句：如何判断经营者决策是否正确?

例句：怎么看哈士奇纯不纯?

上述句子是特指问句，却被错判为正反问句。这是由于句中同时存在正反问的强形式特征和特指问的次强形式特征所致，正反问强形式特征对问句分类的直接增益更大，所以原句分为正反问句。实际上，上句中的“经营者决策是否正确”和“哈士奇纯不纯”并不是原问句的疑问焦点，“经营者决策是否正确”等价于“经营者决策的正确性”，“哈士奇纯不纯”等价于“哈士奇的纯度”，要解决这一个问题，需要介入问句焦点信息的识别工作。

6 结论

本文详细分析了问句形式在问句语料筛选和问句分类中的作用，并以此筛选、标注了4300句问句，构建了目前最大的中文问句分类语料库；此外，还借鉴语言学上的问句形式特征，利用多种机器学习方法构建问句分类模型。根据问句的类别进行分布情况统计，得出特指问在目标数据集中数量与占比均为最高，其次是是非问、正反问以及选择问，这一定程度上也反映了这四类问句在自然语言中的大致分布情况。

问句形式自动分类实验表明，当形式特征集为语气词“吗、吧、么、嘛”、是非问疑问格式、疑问代词、选择疑问格式、正反疑问格式时，对于问句分类具有较高的准确度，表明句尾语气词不仅区分问句，也是问句内类型分类的最有力特征。最终在随机森林模型分类下F1值宏平均达到0.98，F1值微平均达到0.99，特指问分类的F1值最高可达1，是非问分类的F1值达到0.99，正反问分类的F1值达到0.98，选择问分类的F1值达到0.96。本研究当中的问句数据一定程度上可以反映自然语言中问句句类的分布情况，对具体领域中的问句分布研究有一定的参考价值。

同时，可以看出，问句的形式分类本身是一个特征较为明确，规则性较强的问题，使用规则系统也可以获得不差的效果。因此我们认为，在为问句分类时可以增加一个问句形式分类的接口，一方面问句形式自动分类的精度有一定的保障，另一方面可以在这个问句形式分类接口可以集中处理所有问句形式的问题，为问题进一步分类提供分类基础。

文中也存在以下不足：第一，在资源建设方面，本文采用的数据集规模仍需要扩大来进一步考察问句形式特征的效果，届时大规模的数据集可以给深度学习提供充足的泛化空间，将深度学习的方法用于问句分类，以此来与现有分类效果做比较；第二，本文的着重研究的是问句形式对含有说话人发问意图的疑问句语料的筛选和分类问题，而问句形式对于反问句的分析有何作用尚进一步分析。不过已知的是，反问句与疑问句在问句形式上差异不大，只存在有无发问意图的区别，所以通过问句形式来识别反问句效果可能不理想。但可以通过研究一般陈述句变为反问句所需要的问句形式条件，建构反问句形式的意图表达机制，来完成反问句自动生成，从而达到机器表达具有“拟人性”的目的。

参考文献

曹志娟,李祖枢,刘朝涛. 2005. 自动问答系统中的问题理解研究. 计算机科学,2005(11):158–160+230.

傅惠钧. 2008. 关于疑问句的性质与范围. 浙江师范大报(社会科学版),2008(5):77–82.

范继淹. 1982. 是非问句的句法形式. 中国语文,1982(6):426–434.

郭婷婷. 2005. 现代汉语疑问句的信息结构与功能类型. 武汉大学.

黄伯荣. 2017. 现代汉语. 北京:高等教育出版社, 101–107.

刘朝涛. 2010. 中文问答系统中的句型理论及其应用研究. 重庆大学.

刘朝涛,李祖枢. 2008. 基于疑问句句型识别的问题理解研究. 计算机科学,35(12):151–153+189.

吕叔湘. 1985. 疑问•否定•肯定. 中国语文,1985(4).

李宇明. 1997. 疑问标记的复用及标记功能的衰变. 中国语文,1997(02):97–103.

陆俭明. 1982. 由“非疑问形式+呢”造成的疑问句. 中国语文,1982(6).

牛彦清,陈俊杰,段利国,张巍. 2012. 中文问句分类特征的研究. 计算机应用与软件,29(3):108–111.

邵敬敏. 1996. 现代汉语疑问句研究. 上海:华东师范大学出版社.

文勖,张宇,刘挺,马金山. 2006. 基于句法结构分析的中文问题分类. 中文信息学报,2006(02):33–39.

袁毓林. 1994. 正反问句及相关的类型学参考. 北京:北京语言学院出版社.

镇丽华,王小林,杨思春. 2015. 自动问答系统中问句分类研究综述. 安徽工业大学学报(自然科学版),32(01):48–54+66.

朱德熙. 1982. 语法讲义. 北京:商务印书馆, 202–205.

基于组块分析的汉语块依存语法

钱青青
北京语言大学/ 北京海淀学院路15号
qianqingqing1996l@foxmail.com

王诚文
北京语言大学/ 北京海淀学院路15号
chengwen_wang15@126.com

摘要

基于词单元的经典依存语法在面向中文的句子分析中遇到诸多汉语特性引起的困难。为此，本文提出汉语的块依存语法，以组块为研究对象，以谓词为核心，在句内和句间寻找谓词所支配的组块，构建句群级别的句法分析框架。这一操作不仅仅是提升叶子节点的语言单位，而且还针对汉语语义特点进行了分析方式和分析规则上的创新，能够较好地解决微观层次的逻辑结构知识，并为中观论元知识和宏观篇章知识打好铺垫。本文主要介绍了块依存语法理念、表示、分析方法及特点，并简要介绍了块依存树库的构建情况。截至目前为止，树库规模为187万字符（超过4万复句、10万小句），其中包含67%新闻文本和32%百科文本。

关键词： 块依存语法；依存语法；组块；谓词

Chinese Chunk-Based Dependency Grammar

Qian Qingqing
BLCU / 15th Xueyuan Road, Beijing
qianqingqing1996l@foxmail.com

Wang Chengwen
BLCU / 15th Xueyuan Road, Beijing
chengwen_wang15@126.com

Abstract

Classical dependency grammar based on word unit encounters many difficulties in Chinese oriented sentence analysis. Therefore, this paper proposes a Chinese Chunk-Based Dependency Grammar, which takes predicates as the core and chunks as the research object. It seeks the chunks controlled by predicates within and between sentences, and constructs a syntactic analysis framework at the level of sentence group. This operation not only improves the language unit of leaf nodes, but also innovates the analysis methods and rules according to the semantic characteristics of Chinese. It can solve the logical structure knowledge at the micro level and lay a good foundation for the meso argument knowledge and macro textual knowledge. This paper mainly introduces the concept, representation, analysis method and characteristics of Chinese Chunk-Based Dependency Grammar, and briefly introduces the construction of Tree-Bank. Up to now, the size of the tree database is 1.87 million characters (over 40,000 complex sentences and over 100,000 single sentences), including 67% news texts and 32% encyclopedia texts.

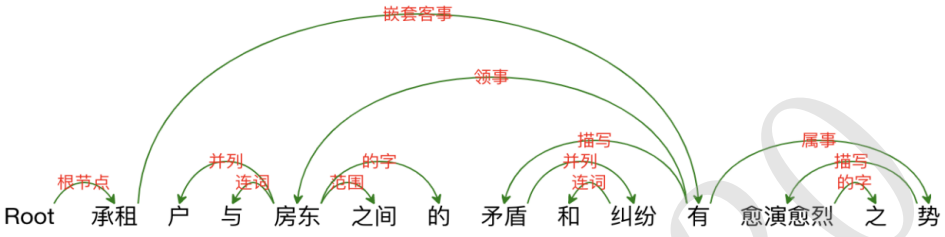
Keywords: Chunk-Based Dependency Grammar , Dependency Grammar , chunk , predicate

1 引言

句法分析是自然语言处理领域中重要的基础研究问题之一，依据句法结构的差异性，可分为短语结构和依存结构。其中依存句法以能够适应汉语灵活语序特征且将句子分析为更加扁平的结构以降低分析、标注、储存难度的优势，近年来获得了更为广泛的应用，在问答系统、知识图谱、信息抽取等任务上发挥着重要作用。

在句法分析中，明确分析的单元是最基础、最根本的要求。传统依存句法分析大多以词作为最小单元，但分词及词性标注可能带来错误级联；汉语实际语篇中，词的词性、词义较为灵活，存在大量的活用、增加语境义的现象，传统依存句法分析较难适应该特性；汉语具有意合特征，同样的语义内容可由语序不同单元表达，过于关注“词-词”关系，使句子依存结构更为繁琐；词与词之间的关系复杂、多变，依存关系类划分的太细，降低了标注的可操作性，带来数据稀疏问题，也会影响到分析器的适应面和鲁棒性。

1)承租户与房东之间的矛盾和纠纷有愈演愈烈之势。



2)我直觉地认为鲁迅是非常中国的人物。

在例1中，主语相对复杂，此处就将主语内部词“承租户”切分开，把“承租”当成了全句的核心，从而也导致了整句依存结构的错误，依存分析时容易陷入复杂“词-词”关系分析的困境而产生错误；在语序方面，若交换“承租户”“房东”或“矛盾”和“纠纷”的语序，甚至将整个主语倒装，变为“矛盾和纠纷，在承租户与房东之间的”，句子的语义都不会产生较大的变化，但分析结构却会因此改变，这是不必要的。而例2中“中国”意为“具有中国品质的”，但此处在分析时仍然将“中国”和“人物”定义为“领事”关系，认为“中国”是一个实体，这是由于无法识别其中活用的信息而导致的。

除了基于词的依存句法分析本身存在的问题，汉语的特殊性也为句法分析带来了困难。

中文多小句、流水句，经过分析，我们发现汉语中至少有25%的小句存在成分缺失的现象[1]。而当前的中文树库中大多利用逗号、句号等标点划分分析边界，容易导致分析单位缺少成分、信息丢失，当流水句中后续小句的主语缺失时，还可能产生歧义：空主语既可能跟先行小句的主语(A)同指照应，又可能跟先行小句的宾语(B)等其他成分同指照应。修饰词（如否定词等）的辖域问题也会导致歧义的产生。

3)她不像她母亲,认为做家务的男人都是没有出息的。

4)他有票，我没有。

5)１９９１年，女足世界杯首次举行，有12支队伍参赛。

在例3中，句子呈现为两个小句，“她不像她母亲”和“认为做家务的男人都是没有出息的”。这个句子形成的图结构是分离的，后一小句的主语既可能是前一小句的主语“她”，也可能是前一小句的宾语“他母亲”，显然主语的不同会导致语义的差别，若割裂地看这个句子，会产生歧义。除了主语缺失之外，例4、5分别为宾语缺失、修饰语缺失。主宾语缺失的问题，已有学者从“篇章回指”“指代消解”等角度进行分析，如陈平（1984）、徐赳赳（1992）等，但仅限于实体之间的指代关系，忽视了提供大量情态信息的修饰语的缺失问题。宋柔（2017）关注到了除

[1]具体分析请见《汉语块依存语法与树库构建

实体之外缺省补全的重要性，他将汉语的句子界定为自足的广义话题结构，把小句界定为基于广义话题结构的话题自足句，利用流水模型生成这两类汉语篇章结构单位，为自然语言处理篇章分析单位提出了新的角度，从汉语篇章微观话题结构的角度为流水句提供了佐证和启示。但汉语中标点句并非只缺省句首的话题成分，句中或句尾的状语、宾语、补语等的缺省也值得关注；按照广义话题结构所生成的句子仅仅提示其话题-说明结构，与句子更深层次的句法语义分析之间缺少衔接，大多还是停留在拆分复杂结构，生成“能说”的自足句层面。

6)他把衣服抖了抖，然后穿上。

7)没有人民民主专政，就不可能保卫和建设社会主义。

话头理论的目的是寻找缺省的话头并生成话头自足句，但生成的话头自足句可能由于句法不通、语义不明等导致“不成句”。如例6中的第二个小句，生成自足句应当为“他把衣服然后穿上”，这是由于话头结构是线性分析的，强调“话头”和“说明”的语序，遇到语言中一些比较灵活的现象时，就会产生不成句的问题；此外，“话头-说明”的关系情况多样，可能是句法上的主谓关系，也可能是语义上的衔接关系，就使得在标注时存在两可情况，也可能与篇章级别的分析产生混杂，如例7中的“话头-说明”关系一般认为是复句中的条件关系。

指向不明确也会使句子分析不准确，下面这几个例子结构相似，但句子中名词性短语、修饰短语受哪些动词的支配却不尽相同。

8)老师让小张来办公室一趟。

9)我们洗衣服挺累。

10)我劝他手术好几天了。

针对以上的问题，我们提出汉语的块依存语法，以组块为研究对象，以谓词为核心，在句内和句间寻找谓词所支配的组块。分析时，利用汉语中的组块和组块间的依存关系，将成分缺失和指向不明的问题转化为小句内组块依存问题和小句间的组块缺省问题。补全缺失的成分，为后续任务提供准确的分析单元，消除由于指向不明确而导致的歧义。

2 组块及其类别

由于汉语句法的特殊性，“块”具有很好的现实意义。“块（Chunk）”概念最早由Abney（1991）提出，他认为句法分析可以分为三个阶段来进行，以达到简化句法分析任务的目的。即：对块进行识别、分析块的内部结构、分析块之间的关系。本阶段的工作，主要为第三步。

我们将组块定义为：由连续词语或语素整合而成的序列，表现为同一句子层级中充当句法成分的各个连续单元，例如下面这个句子被分为4个组块。

11)这句话| 只| 是| 一个例子。

组块按照其功能，可分为句法结构层面和非句法结构层面两部分（图1），其中句法结构层面的组块指在句子内部与谓词存在向心关系的组块，按照与谓词的关系，又可层层下分，而非句法结构层面的组块指通常在篇章层面用于衔接或表示语气作用的组块。

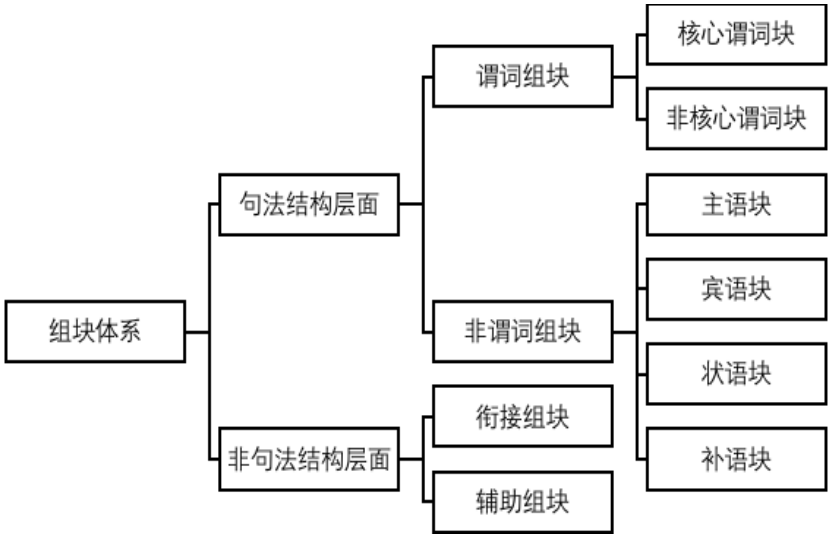


图 1: 组块体系

(1)谓词组块 谓词组块即由核心述语构成的组块，能够支配句中的非谓词块，是所在句子层级的核心，由最内部的小括号“()”表示。谓词组块主要由动词性、形容词性的词或短语[2]来充当，在一些特殊句中也会有空谓词组块的存在。句子中最顶层的谓词组块（即整个句子的核心）是核心谓词组块，出现在修饰语[3]、谓词性主宾语中的谓词组块为非核心谓词组块。

12)他(狼吞虎咽地(**吃完了**))饭。

13)这个人**()**黄头发。

14)我(现在(**承认**))){你((**做**)得比我好)}。

以上划线部分均为核心谓词组块，其中例13由补充的空述语充当。例14的核心谓词组块“承认”是整个句子的核心，而非核心谓词组块“做”是宾语“你做得比我好”中的核心。

(2)非谓词组块

非谓词组块指在结构上依存于谓词组块的组块，主要有主语块、宾语块、状语块、补语块几类。

a)主语块

主语块即结构中的主语，包括主谓谓语句中的大小主语。按照其内部是否还嵌套有谓词组块可将其分为体词性主语块和谓词性主语块。主语块在结构上依存于谓词组块。以下几例中的黑体部分为主语块:

15)**他**((说话)很快)。

16)**电脑**{**我**(可(是))门外汉}。

17){**(很(丰富))(却不(精细))**}(也不(是))我们说的优秀。

b)宾语块

宾语块即结构中的宾语，按照其内部是否还嵌套有谓词组块可将其分为体词性宾语块和谓词性宾语块。宾语块在结构上依存于谓词组块，谓词性宾语用“{ }”表示，双宾之间用“||”隔开。以下几例中的黑体部分为宾语块:

18)[在他壮年时，]他(爬上过)**珠峰**。

19)我(现在(承认))){**你((做)得比我好**)}。

20)(感谢)你(告诉)**我||这个好消息**。

c)状语块

状语块指述语中位于谓词组块前部与其紧邻和被其他成分或标点隔离的组块，对核心语块起到修饰作用，受谓词组块支配。以下几例中的黑体部分为状语块:

21)(**一年内**(新增))培育科技型企业||3465家。

22)[**别把孩子的教育，**](全(寄))希望[于教育机构上]。

[2]一般由V+着了过、V+单音节补语、两个连续的单音节V组成，字典中收录成语、常用俗语等也作为谓词组块。

[3]出现在修饰语中的非核心谓词组块将在下一步工作中进行处理

d)补语块

补语块指在句中充当补语的组块，一般位于谓词组块后部，可与谓词组块紧邻或被其他成分或标点隔离，对谓词组块起到修饰作用，受谓词组块支配。以下几例中的黑体部分为补语块：

23)她(哭着)((跑)**出来**)。

24)[别把孩子的教育，](全(寄))希望[**于教育机构上**]。

(3)衔接组块

衔接语块由连词、话语标记、插入语等组成，在句中主要发挥衔接功能，属于篇章成分。用尖括号“<>”表示，以下黑体部分为衔接语块：

25)她(非常不想(去))，<**因为**>(今天(下))雨。

(4)辅助组块

辅助组块由辅助语构成，句法上与句中其它各个成分之间没有结构上的关系，在句中主要承载表达语气的功能，用“<<>>”表示。以下各例中黑体部分为辅助语块。

26)他(走了)<<**吗**>>?

27)<<**嗯**>>，<<**好的**>>，我(知道了)。

3 块依存语法

3.1 块依存语法的表示

块依存语法主要分析非篇章成分的组块，即基于句法结构层面的6类组块，通过分析对象的选择，可将构建自足小句的过程与篇章关系的界限划分清楚。衔接组块用于表示句间的衔接关系，辅助组块则承载了表达语气的功能，均不应与句内的成分混杂。在分析句子内部成分时，我们认为核心谓词组块是句子的核心，各类非谓词块均受核心谓词组块的支配并依存于核心谓词组块之上，若某非谓词块和谓词组块之间存在依存关系，则称该非谓词块为谓词组块的从属成分，谓词组块为该非谓词块的依存对象。除了一些特殊的独词句，一般认为句子中都存在一个或多个核心，非谓词块至少依存于一个谓词组块之上。谓词组块作为句内各语块的依存对象，其左右上下各有四个点位，分别表示其主语位（1号位）、修饰语位（2号位）、宾语位（3号位）、述语位（4号位），各非谓词块按照其类别分别依存于谓词组块的四个节点上，依存线条从谓词组块的四个节点指向其从属成分。

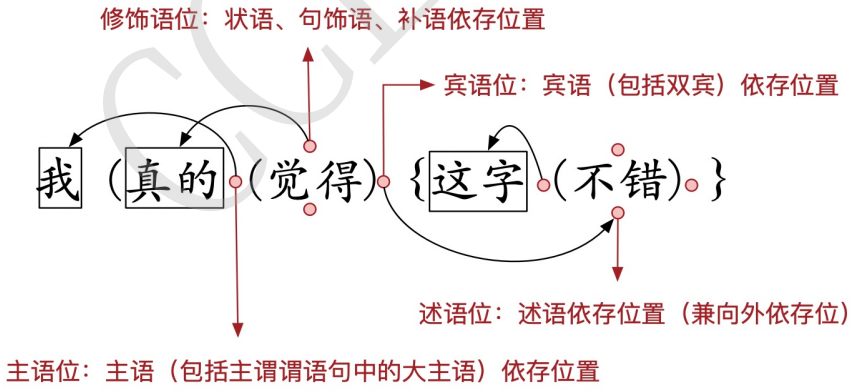


图 2: 块依存标注图示

主语，包括主谓谓语句中的大小主语依存于谓词组块的1号位；

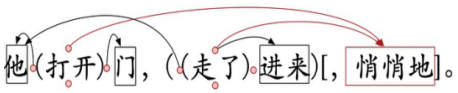
状语、补语依存于谓词组块的2号位；

宾语，包括双宾语中的远近宾语依存于谓词组块的3号位；

述语省略时从4号位置与相关述语连接，当某谓词组块依存于其他谓词组块时从4号位向外依存。

不同于Robinson（1970）所提出的四条依存分析方法的公理，块依存语法分析中，允许非谓词块、非核心谓词组块有一个或多个依存对象，允许谓词组块有多个从属成分，且允许线条交叉、跨句。中文中存在较多的非投影结构（闻媛，2018），允许线条交叉、组块多依存对象，能够使分析结果更清晰、准确。

28)他(打开)门，((走了)进来)[，悄悄地]。



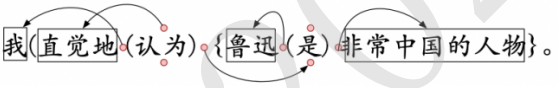
此例中，前一小句缺少了修饰成分“悄悄地”，后小句缺少了主语“他”，必然导致分析不完整。在块依存语法中，允许线条跨句、交叉，找到两个小句中核心谓词的所有从属成分，即可将两个小句补充完整。

3.2 块依存语法的分析方法

在下述两例中，“承租户与房东之间的矛盾和纠纷”“非常中国的人物”均为一个组块，语义具有相对的稳定性，更符合语言的认知规律。以组块为研究对象，能够减少分词碎片，降低活用、语境义等带来的分析错误；同时，避免纠结于“词-词”之间的关系，使得依存关系得到了精简，更关注于句子的整体结构，进一步降低存储和分析的复杂性，加强鲁棒性。在此基础上进行分析，能够在保证浅层结构正确的情况下为更深层次的分析打下基础。

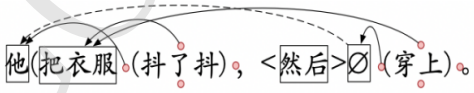
29)承租户与房东之间的矛盾和纠纷有愈演愈烈之势。

30)我直觉地认为鲁迅是非常中国的人物。



其次，通过跨句找回依存块，能够补全句子成分。组块缺省指在线性的结构标注中由于承前蒙后省略或小句分割等情况导致核心谓词组块在该小句内缺省了从属成分，在这样的情况下需要将句子在上下文中进行分析并在其四个节点处补全缺省的从属成分。

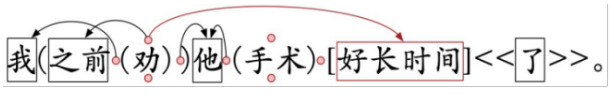
31)他(把衣服(抖了抖)，<然后>(穿上)。



在这个例子中，小句“穿上”缺省主语块和状语块，我们在这里以缺省的主语块为例，将其用“”表示，因此依存于“穿上”的主语块是“”，而“”是前一小句“他”的省略。因而为了寻回缺省的组块，使后一个小句成分完整，我们认为前一个小句的主语块“他”除了依存于所属小句的核心谓词组块“抖了抖，也依存于后一个小句的核心谓词组块“穿上”。在补全了缺省的组块之后，我们还可以将前后两个小句拆分为：“他(把衣服(抖了抖)”和“他(把衣服(穿上))”，这样，二者在这一个简单的上下文中，就没有缺省的从属成分了。篇章层面的组块“然后”并没有依存的对象，也就不进入自足句构建的过程，仅用于表示两个小句之间的顺承关系。以上的补全过程，是在排除了篇章层面的组块之后、以结构为指导的、句法层面的补全，能够与下阶段分析句间关系相衔接，且更具有理据性——能够成为另一个小句的一部分是因为它受到其中动词的支配。

针对依存对象不明确的问题，则通过寻找谓词的依存块，更好地明确句意。我们看以下这个例子：

32)我(之前(劝))他(手术)[好长时间]<<了>>。

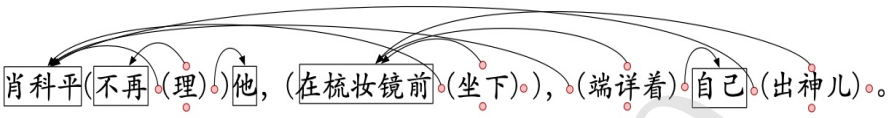


对于这样的句子，一般的处理原则是“默认左归”或者“默认右归”，采取“左归”方法时，认为“他”是“劝”的宾语，但和“手术”之间没有关系，“好长时间”是“手术”的修饰语。如果按照这样分析，这个句子的意思可能就变成了：我之前劝他，我手术好长时间了。但显然，这句话并非这个意思。因此我们判断其依存对象，认为“他”既是劝的从属对象，也是“手术”的从属对象，而“好长时间”则是“劝”的从属对象。这样，能够对这一类句子达到更好的分析效果。对兼语句、连谓句等特殊句式，也能做到很好的区分和分析。

按照缺省的组块类型，我们将组块缺省分为非谓词块缺省和谓词组块缺省。以下各举几例。**（1）主语块缺省**

主语块缺省即句子或小句中的谓词成分因省略或标点等原因缺少从属的主语块。事实上，有相当一部分的主语块缺省是由于语音上的停顿、语篇成分的插入造成的，在书面上表现为标点、衔接语、辅助语等。当忽略这些成分时，我们可以发现这类小句可与前后带有主语块的小句形成复谓或并列结构，从而找回主语块。

33)肖科平(不再(理))他，(在梳妆镜前(坐下))，(端详着)自己(出神儿)。



此句中，“坐下”“ 端详着”“ 出神儿”缺省了主语，“端详着”“出神儿”还缺省了状语，找回后，我们可以将其补充为完整的三个小句：

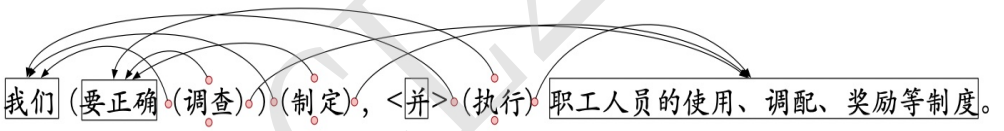
34)肖科平(不再(理))他，

35)肖科平(在梳妆镜前(坐下))，

36)肖科平(在梳妆镜前(端详着))自己(出神儿)。

（2）宾语块缺省

37)我们(要正确(调查))(制定)，<并>(执行)职工人员的使用、调配、奖励等制度。



宾语块缺省即句子或小句中的谓词成分因省略或标点等原因缺少从属的宾语块。在这个例子中，两个小句都缺省了一些成分，其中前一小句中的两个核心谓词缺省了宾语块，后一个小句的核心谓词组块“执行”缺省了主语、状语。进行分析后，我们可将两个小句补全为：

38)我们(要正确(调查))(制定)职工人员的使用、调配、奖励等制度，

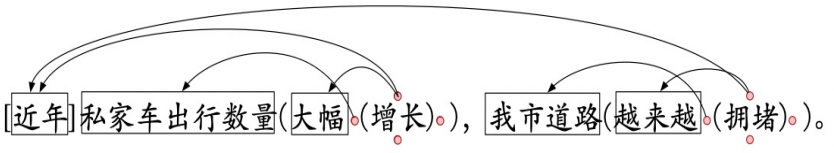
39)我们(要正确(执行))职工人员的使用、调配、奖励等制度。

此句中的“并”属于衔接组块，用于提示篇章中上下文的衔接关系，是我们下一步工作所需要关注的对象。

（3）状语块缺省

状语块中承载了大量的时地信息、情态信息，然而位于句首的状语在分句的时候，易随第一个小句进行切分，而第二个小句就因此缺少了这个状语。如下例中，我们可以将“近年”重新依存至“拥堵”，将后一小句的时间信息补充完整。

40)[近年]私家车出行数量(大幅(增长))，我市道路(越来越(拥堵))。



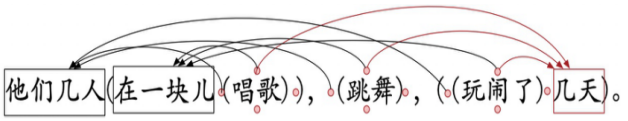
拆分后的完整小句为：

41)[近年]私家车出行数量(大幅(增长))，

42)[近年]我市道路(越来越(拥堵))。

（4）补语块缺省

43)他们几人(在一块儿(唱歌))，(跳舞)，((玩闹了)几天)。

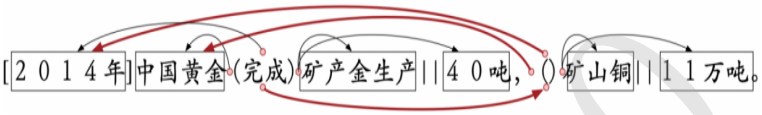


状语块缺省即句子或小句中的谓词成分因省略或标点等原因缺少从属的状语块。在上例中，补全“几天”作为“唱歌”“跳舞”的补语之后，为其增加了时间信息，句意更完整了。

（5）谓词组块缺省

谓词组块缺省是我们认为的一类特殊缺省情况。指由于省略前文中已出现过相同的核心谓词组块而造成的缺省。在这样的情况下，需要将缺省的核心谓词组块依存到原有核心谓词组块上。通过这种方法，我们可以补全原本缺省的谓词，使得句意更加清晰。

45)[2014年]中国黄金(完成)矿产金生产||40吨，()矿山铜||11万吨。



经过分析之后，生成的完整小句为：

46)[2014年]中国黄金(完成)矿产金生产||40吨，

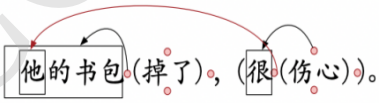
47)[2014年]中国黄金(完成)矿山铜||11万吨。

3.3 组块分割与小块依存

一般进行块依存分析时，非谓词块以整体的形式充当谓词组块的从属成分，但在某些特殊情况下，存在小块依存的现象。小块依存指在一个组块内部划分更小组块，进行依存关系构建。在小块依存中，谓词组块的从属成分并非是一个完整的组块，而是某个组块的一部分。小块依存现象在体词性的主宾语组块以及状语、补语组块中较为多见。

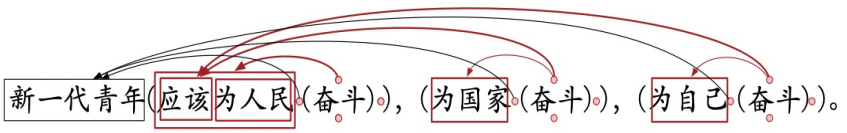
体词性主宾语块的小块依存多出现在定语和中心语之间存在从属或整体部分关系的情况下。下例中第二个小句通过块依存方法可找回主语并补全。

48)他的书包(掉了)，(很(伤心))。



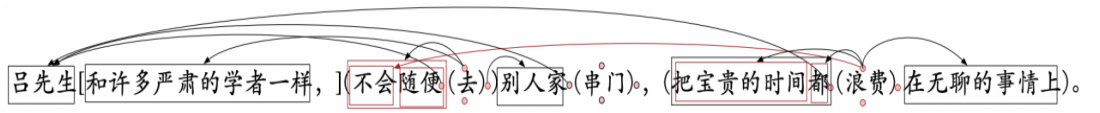
有部分状语或补语组块内部不同的部分从属于不同的一个谓词组块，此时也存在小块依存的现象。如下例中，若不分割组块，则“应该为人民”仅修饰第一个“奋斗”，将状语分割并重新分析其依存关系之后，能够更明确三个核心谓词“奋斗”的状语，在补全主语之后，即可形成3个完整的小句。

49) 新一代青年(应该为人民(奋斗))，(为国家(奋斗))，(为自己(奋斗))。



50)吕先生[和许多严肃的学者一样，](不会随便(去))别人家(串门)，(把宝贵的时间都(浪费)在无聊的事情上)。

否定性词语对于确定文本中的事件到底发生与否和是非评价有决定性的影响，尤其是否定词的辖域到底管到哪儿也决定了信息抽取的准确性。上例中，若无小块分割及跨小句的依存，则后一小句的语义与正确语义截然相反。而正确的语义应为：



51)吕先生和许多严肃的学者一样，不会随便去别人家串门，
52)吕先生和许多严肃的学者一样，不会把宝贵的时间都浪费在无聊的事情上。

4 块依存树库构建

目前，我们正在展开基于块依存语法的树库构建，经标注实践验证，该理论体系及表示方法能够覆盖绝大部分的语言现象，详细构建方法、过程及数据分析请见另文讨论[4]，以下简要进行介绍。基于块依存理论，以数据标注规范作为指导，以两两对比标注的模式，在基于浏览器的在线标注系统中，我们标注了百科和新闻领域文本，构建了汉语块依存树库。截至目前为止，树库规模为187万字符其中包含67%新闻文本和32%百科文本（仍在扩展中）。其中，新闻文本来源于新浪2006年新闻、新华社2012-2018 间新闻，百科文本来源于百度百科，分属自动化控制系统、电子学与计算机、轻工、大气与海洋及水文科学、航空航天、经济学等领域。各类别文本信息见表1。

	文件数	字符数	复句数（ROOT）	单句数（IP）
新闻	1461	1266466	29471	73973
百科	738	608389	16702	32289
总计	2199	1874855	46173	106262

表 1: 不同来源文本标注统计

当前树库中共包含299763个谓词词符，13425个谓词词形。其中约有1877个谓词（token）无从属成分，其余谓词均至少支配一个从属成分，依据目前定义的6类谓词和组块之间的依存关系，统计结果见表2。

依存关系	核心谓词（type）	核心谓词（token）	依存块数量（token）	谓词平均可支配组块数
NP-SBJ	11705	96199	101877	1.059
NP-OBJ	6757	72938	73850	1.013
VP-OBJ	1151	11901	14357	1.206
NULL-MOD	10267	73309	104980	1.432
VP-SBJ	765	2297	2805	1.221
VP-EMP	12	13	17	1.308

表 2: 各类依存块依存情况统计

从统计结果上看，在出现的一万三千多个谓词中，进行缺省补全后约有87%的谓词可支配名词性主语块，其次为修饰语块，为76%左右。这表明汉语中谓词支配主语和修饰语的普遍性，在明确动词具备支配该类组块能力的情况下，进行缺省补全是有必要的。另外，谓词支配修饰组块的能力最强，树库中平均一个谓词可支配1.432个修饰语块或小块。修饰语块中携带了大量的情态、时间、地点等各类语义信息，但内部结构相对复杂，存在框式结构、介宾短语等内部成分，因此进行小块切分，能够便于后续的语义角色分析、情态结构分析等工作。单个谓词支配名词性主语和宾语的组块数量相对较少，但仍略大于1，则表明语言中主谓谓语句及双宾

[4]具体分析请见《汉语块依存语法与树库构建》

语的现象占少数，后续工作中分析单主语和单宾语与谓词间的语义关系应作为重点，而相对于双宾句，主谓谓语句优先级更高。

5 块依存语法的特点

块依存语法是一种结合了组块分析、依存语法的语言分析方法。按照块依存语法所生成的句子，与宋柔所提到的“自足句”有相似之处，但也更进一步关注句子内部。

块依存语法以组块为研究对象，能够避免纠结于“词-词”之间的依存关系，关注句子的整体结构，进一步降低存储和分析的复杂性，也能够达到减少分词碎片、加强鲁棒性的目的；关注句法结构层面的组块，能够厘清句内-句间的界限，为篇章关系分析打下基础；以谓词为核心，在上下文中找到其支配对象，能够在句子层面补全缺省成分的同时明确内部成分的指向、句子结构。此外，块依存语法不仅关注常出现在句首的主语、状语成分，也关注经常出现在句中或句末的宾语、补语等，使生成的句子更加完整。

我们还注意到，以谓词为分析对象使得句法分析根据灵活。块依存语法分析能够以块依存图的形式对句子进行展现。整个句子以空节点为根，指向句中的核心谓词，核心谓词又有各个线条指向其支配成分。在篇章关系分析中，无论是寻找句间关系还是直接分析谓词间关系，都能够以更准确的分析单元为着力点。

袁毓林（2002）曾将信息抽取所需的语义知识分为三类，分别为宏观篇章知识、中观论元结构知识、微观层次的逻辑结构知识。块依存语法能够解决微观层次的逻辑结构知识，并为中观论元知识和宏观篇章知识打好铺垫。事实上，核心谓词的支配成分除了特殊的空述语之外，均可与汉语中的论元结构相挂钩，其余的状语成分、补语成分也可提示情态信息，此时的谓词论元、情态成分等均已齐全，仅需进行分类即可。在宏观层面，已明确的篇章分析单位，结合其余的辅助组块、衔接组块，为分析篇章的逻辑语义关系带来便利。

6 结论

本文创造性地提出了汉语块依存语法，并介绍了其标注体系和目前构建的树库规模。块依存语法在句内和句间寻找缺省的组块，补全缺省成分，以此为基础，也能够更深入地进行篇章层面的“小句-¿句间-¿篇章”关系探索。块依存语法与具体的语境、语用环境相结合，能够较好地解决当前中文自然语言处理中存在的分析对象不明确、依存对象不清晰、成分缺失等问题，更好地服务于事理图谱、知识图谱、问答系统、信息抽取各项任务。

参考文献

Steven P. Abney. 1991. *Parsing By Chunks. Principle-Based Parsing.*,257-278. Springer Netherlands.

Robinson, J.J. 1970. *Dependency Structures and Transformation Rules.*,1970,46(2) [J]Language.

Zhou Ming. 2000. *A Block-Based Robust Dependency Parser for Unrestricted Chinese Text.* The second Chinese Language processing workshop attached to ACL 2000, HongKong.

陈平. 汉语零形回指的话语分析[J]. 中国语文, 1987,(5):363-378.

陈亿,周强,宇航. 分层次的汉语功能块描述库构建分析[J]. 中文信息学报.2008(03):24-31+43.

郭艳华,周昌乐.. 面向多领域多来源文本的汉语依存句法树库构建[J]. 中文信息学报.2019. 38-46.

李素建. 汉语组块计算的若干研究[D]. 中国科学院研究生院（计算技术研究所, 2002.

刘伟权,王明会,钟义信. 建立现代汉语依存关系的层次体系[J]. 中文信息学报,1996(02):32-46.

卢露,矫红岩,李梦,荀恩东. 基于篇章的汉语句法结构树库构建[J/OL].自动化学报:1-12[2020-08-18].http://kns.cnki.net/kcms/detail/11.2109.TP.20200521.1558.007.html.

宋柔. 汉语篇章广义话题结构的流水模型[J]. 中国语文, 2013(06):483-494+575.

宋柔,葛诗利,尚英,卢达威. 面向文本信息处理的汉语句子和小句[J]. 中文信息学报, 2017,31(02):18-24+35.

徐赳赳. 现代汉语篇章回指研究[M]. 中国社会科学出版社,北京,1992.

闻媛,宋丽,吴泰中,李斌,周俊生,曲维光. 基于中文AMR语料库的非投影结构研究[J]. 中文信息学报, 2018,32(12):31-40.

尹鹏. 基于SVM的中文组块间依存关系分析[D].大连理工大学,2006.

袁毓林. 流水句中否定的辖域及其警示标志[J].世界汉语教学,2000(03):22-33.

袁毓林. 信息抽取的语义知识资源研究[15][J].中文信息学报,2002(05):8-14.

周明,黄昌宁. 面向语料库标注的汉语依存体系的探讨[J]. 中文信息学报, 1994(03):35-52.

周强. 汉语基本块描述体系[J]. 中文信息学报,2007(03):21-27.

周强,孙茂松,黄昌宁. 汉语句子的组块分析体系[J]. 计算机学报,1999(11):1158-1165.

新支话题的句法成分和语义角色研究

卢达威
中国人民大学文学院/ 北京市海淀区中关村大街59号
wedalu@163.com

摘要

话题的延续和转换是篇章中重要的语用功能。本文从句首话题共享的角度对话题延续和转换进行了分类，分为句首话题延续、句中子话题延续、完全话题转换、兼语话题转换、新支话题转换五种，进而对话题转换的特殊情况——新支话题展开研究。基于33万字的广义话题结构语料库，本文对新支话题的句法成分、语义角色进行了统计和分析。句法成分方面，宾语从句或补语从句主语、主谓谓语句小主语、状语起始句的主语、句末宾语、连谓句非句末宾语、兼语句兼语、介词宾语甚至状语等都能成为新支话题，从而引出新支句，其中句末宾语做新支话题的情况最多，但未发现间接宾语作为新支话题的情况；语义角色方面，大部分主体论元（施事、感事、经事、主事）和客体论元（受事、系事、结果、对象、与事），及少数凭借论元（工具、方式，材料）和环境论元（处所、终点、路径）能成为新支话题引出新支句。其中，系事和受事成为新支话题情况最显著，施事、结果和对象次之。本文的研究揭示了句法、语义对话题转换这一语用现象的一种可能的约束途径。这将有助于人和计算机更深入地理解汉语篇章的话题转换机制，以期将这种语用现象逐步落实到语义直至句法的形式中，最终实现计算机对话题转换的自动分析。

关键词： 话题延续；话题转换；新支话题；句法成分；语义角色

A Study of Syntactic Constituent and Semantic Role of New Branch Topic

LU Dawei
School of Liberal Arts, Renmin University of China
wedalu@163.com

Abstract

Topic Continuity and Topic Shifting are important pragmatic functions in discourse. From the perspective of topic sharing at the beginning of a punctuation clause(P-Clause), this paper classifies the Topic Continuity and Topic Shifting into five categories: *Topic Continuity at the beginning of a P-Clause*, *Sub-topic Continuity among a P-Clause*, *Complete Topic Shifting*, *Pivotal Sentence Topic Shifting* and *New Branch Topic Shifting*, where we do a further research on special cases of topic shifting. Based on a corpus of Generalized Topic Structure with 330,000 characters, we make a statistical analysis of the syntactic constituents and semantic roles of the New Branch topic.

本研究得到国家社会科学基金青年项目“汉语话题延续与转换机制及其计算模型研究”（18CYY030）的支持。

As for syntactic constituents, the subject of object clause or complement clause, the small subject of subject-predicate predicate sentence, the subject of adverbial starting sentence, the object at the end of a sentence, the object of a sentences with serial verbs, the object of a pivotal sentence, the prepositional object, and even a adverbial can serve as a New Branch topic, which leads to a New Branch clause. Among them, the object at the end of a sentence is the most frequently used as a *New Branch Topic*, but no indirect object is found as a *New Branch Topic*. As for semantic role, most of the subject arguments (*agent*, *sentiment*, *experiencer*, *theme*) and object arguments (*patient*, *relative*, *result*, *target*, *dative*) are found. As for the supporting argument (*instrument*, *manner*, *material*) and environment argument (*location*, *goal*, *path*), a few of them can function as New Branch topics and lead to New Branch clauses. Among them, *relative* and *patient* are the most typical New Branch topics, followed by *agent*, *result* and *object*. Our study reveals a possible path for syntactic and semantic constraints on *Topic Shifting*. This study can serve people and computers for deeper understanding of the topic shifting mechanism of Chinese discourse, promoting to implement this pragmatic phenomenon into the form of semantics and syntax, consequentially accomplishing automatic analysis of Topic Shifting by computers.

Keywords: topic continuity , topic shifting , New Branch topic , syntactic constituent , semantic role

1 新支话题和话题共享

“新支话题”是汉语小句复合体相关研究中（宋柔，2008，2013，2018）提出的概念[0] 。一段语篇中，当某个标点句[1] 缺少话题，需要共享上文非句首的语言片段作为话题，但不共享该片段之前的部分，则该被共享的语言片段称为“新支话题”，以“新支话题”为话题的标点句称为“新支句”。例如：

例（1）

①顾炎武一抬头，
②　　见到壁上挂着**一幅高约五尺、宽约丈许的大画**，//新支话题
③　　‖　　绘的是一大片山水，//新支句
④　　‖　　笔势纵横，//新支句
⑤　　‖　　气象雄伟，//新支句
⑥　　不禁喝了声采，

例（1）由6个标点句组成，根据“广义话题结构”标记方法（宋柔，2013），我们对每个标点句按照其话题—说明情况，用换行缩进的形式进行排列。这样就展示了各成标点句之间的话题共享关系。其中，①句不缺话题，后面5句都缺话题。②句和⑥句共享①句的“顾炎武”为话题，整体构成一个围绕“顾炎武”的话题结构。③至⑤句共享第1句的“一幅……大画”作话题，但不共享“一幅……大画”之前的部分。所以②句中的“一幅……大画”称为新支话题，③至⑤句是对新支话题的说明，是新支句。

例（1）中②至⑤句语用上还可以看作零形回指。零形回指有多种情况，这种句首的零形回指比较特别，而新支句又是句首零形回指的特例。

[0]宋柔（2013）关于“小句复合体”的研究，原称为“广义话题结构”。其中的“话题”是指字面上出现的某个词语，该词语是话语的出发点，其下文（少数情况为上文）是这个词语所引发的说明。这是对“话题”这一概念更广泛、更朴素却更容易形式化的定义，不仅包含传统意义上语用方面的名词性的“话题”，还有谓词性成分、状性成分，甚至连词等，故称为“广义话题”，后为避免争执，改称“话头”。本文所说的新支话题的成分，虽然也是“广义话题（话头）”的一种，但主要为名词性成分，更接近于传统意义上狭义的语用方上的“话题”。故本文仍使用“话题”这一术语。

[1]篇章语法的研究通常以小句为单位，但小句的定义标准有一定模糊性，常纠结于大量的零句、从句、状语成分等是否小句。自然语言信息处理领域在处理小句时，往往采取标点符号来直接切割，宋柔（2008，2013）明确定义为“标点句”，即指以逗号、分号、句号、叹号、问号、直接引语的引号以及这种引号前的冒号所分隔出的词语串。再以标点句为基础，讨论标点句之间的话题关系或逻辑关系。本文采取这一方法，以标点句的作为篇章单位，在不至混淆的情况下，本文所说的“句”一般指标点句。

从篇章的组织结构看，例（1）各句是通过句首的零形回指，组织成一个话题结构的。我们把这种标点句句首的零形回指称为**话题共享**。在汉语中，这种话题共享不是临时的现象，而是作为一种篇章组织机制而普遍存在的。卢达威等（2014）发现，若将汉语篇章按照标点句切分，58.3%标点句的句首缺少话题。标点句正是通过话题共享，组织成话题结构，进而组成篇章的。

从话题的延续和转换的角度看，篇章通过不同的话题共享方式实现话题的延续与转换，从而推进篇章发展。如例（1）总体延续的话题是“顾炎武”，但中间插入了对新支话题“一幅……大画”的说明，语用上形成了一种临时的话题转换。本文拟从话题的延续和转换的角度讨论新支话题的特性，分析什么样的语言片段能够被后续说明，成为新支话题。

2 话题延续与转换的类型

话题延续有多个层次，从宏观到微观，包括主题的延续（thematic continuity）、动作的延续（action continuity）和话题/参与者的延续（topics/participants continuity）（Givón，1983）。即使微观的话题/参与者延续，也有不同的延续方式或回指方式，形式从简到繁包括零形回指、代词回指、同形名词、指示词+名词、描写性定语+名词等（方梅，2005）。本文关注最微观，形式最简单的话题延续或转换，即仅讨论由话题共享（即句首零形回指）而形成的话题延续与转换，并认为新支话题是话题转换的一种特殊情况。为行文方便，下文所说的“话题”都指由句首零形回指而形成的共享话题。

基于共享话题的视角，话题的延续和转换可以分为5种情况。有些情况前人已有研究，但角度、范围和说法可能有所不同，故相应的分类中我们尽量用前人文献中的例子，以便对应。

（一）句首话题延续。句首的话题继续被后续的标点句共享。例如：

例（2）

①**那个人**也意识到跑不脱，

②　　只好扔掉麻袋（选自陈平，1987）

例（2）中①句话题完整，②句缺少话题，共享①句的“那个人”。这类话题延续陈平（1987）称为平行推进。

（二）子话题的话题延续。虽然直接谈论的是子话题，但原话题被保留。实际上仍在说明原话题，只是缩小到关于子话题的那个视角。子话题和原话题之间通常为部件、属性、隶属等相关关系。这个语义关系决定了语用关系的话题延续。例如：

例（3）

①**这个女孩眼睛**很大，

②　　很漂亮（选自曹逢甫，2005，P104）

例（3）中①句不缺话题，是主谓谓语句。②句有两种解读，一种解读为，“很漂亮”直接说明“这个女孩”，即看作句首话题延续。另一种解读为，②句共享①句的小主语“眼睛”为话题，构成子话题延续。与第1节描述的“新支话题”不同的是，虽然②句直接说明的是“眼睛很漂亮”，但“眼睛”是“这个女孩”身体的一部分，②句间接以“这个女孩”为外层话题，实现了外层话题的延续。直接证据是把①句“眼睛”之前的部分与②句连起来，所构成的句子“这个女孩眼睛很漂亮”是合格通顺的。曹逢甫（2005）的看法与此相似，他把第一种解读看作是以“这个女孩”为主题的主题链；第二种解读看作是以“眼睛”为主题的小主题链内嵌于“这个女孩”为主题的主题链中，相当于文本所说的子话题的话题延续。

（三）完全话题转换。即标点句话题自足，不需要从上下文共享话题。例如：

例（4）

①红海早过了，

②船在印度洋面上开驶着。（选自钱钟书，《围城》）

例（4）①句的话题是“红海”，②句的话题是“船”，虽然在一个句号句内，但两个标点句都话题自足，相互不形成话题—说明关系，就话题延续方面看，我们认为属于完全话题转换。完全话题转换后，下文不能再以延续前面的话题。如该例中，下文不能以话题共享的形式对“红海”进行说明，除非再重复一遍话题或以其他方式回指。对于篇章中其他形式的回指，包括代词回指、同形名词、指示词+名词、描写性定语+名词等形式，由于不是话题共享，与共享话题的延续与转换不是一个层面的问题，我们暂时也归在完全话题转换中，日后再进一步分类和讨论。

（四）兼语句话题转换。例如：

例（5）

①上面有**个干部模样的人**，

②　　托着一个袖珍半导体收音机（选自陈平，1987）

例（5）①句是主动宾结构的句子，②句以①句的宾语“（这）个干部模样的人”作为话题。这类话题与子话题延续的不同之处在于，②句语义上与①句的“上面”等词语无关，没有间接延续①句外层话题。然而，②句在补全缺失成分的时候，把①句话题前的部分连起来也是成句的，即“上面有个干部模样的人托着一个袖珍半导体收音机”。也就是说，虽然语义上②句与①句的外层话题没有太多关系，但通过句法仍保持了一定的联系。屈承熹（2006）称这类情况为话题套叠，陈平（1987）称该例为层继推进。

（五）新支话题转换。例如：

例（6）

①这把**五人**吓坏了，

②　‖　　跟办事员讲了许多好话。（选自钱钟书，《围城》）

例（6）和例（1）一样属于新支话题的转换。②句共享的话题是①句的介词宾语“五人”，“五人”成为了新支话题，②句是新支句。新支话题转换不同于其他几类情况的地方是，②句无论是句法上还是语义上，都不能与①句新支话题“五人”之前的部分形成更大的关联。若仿照第二类和第四类的方法，把“五人”前面的部分与②句连起来，即“*这把五人跟办事员讲了许多好话。”是不通的。即相当于发生了话题的转换，从原话题“这”转到新支话题“五人”。为了直观可读，标记时，在新支话题的左下方插入一个双竖线标记，表示新支句同新支话题的左边成分是隔离的。

需要强调的是，这种新支话题的话题转换是局部的和临时性的，因为后文可以以零形式继续延续原有的话题，如例（1）。我们预测，这类新支话题和新支句是有一定限制条件的。另一方面，在语料整理中发现，这种新支话题不一定出现在动词宾语位置，例（6）就是介词宾语作为新支话题引出新支句的例子。对此，后文还将进一步分析。

综合起来，话题延续和转换的分类汇总如表1。表1列出4个特征：原话题是否被新标点句直接说明、原话题是否被新标点句间接说明、原话题与新标点句能否连成句、新话题是否在原句内，并使用这4个特征比较了话题延续和转换的5种类型的异同。

话题延续转换方式类别		原话题被新标点句直接说明	原话题被新标点句间接说明	原话题与新标点句能连成句	新话题在原句内
话题延续	句首话题延续	＋	－	＋	
	子话题延续	－	＋	＋	
话题转换	完全话题转换	－	－	－	－
	兼语句话题转换	－	－	＋	＋
	新支话题转换	－	－	－	＋

Table 1: 话题延续转换方式的分类

前人研究大多注意到了前四种情况，对新支话题情况及其形式特点，特别是原话题与新标点句连起来不能成句的特点，没有足够的重视。而这种不能成句，正是判断新支话题的直接标准。

虽然统计中新支话题数量不多，但新支话题的研究很有意义。对于完全的话题转换，研究其成因需要在篇章全局范围内考察，难度较大。对于新支话题，虽然新话题和原话题没有直接话题—说明的关系，但新话题出自于原话题—说明的局部语境，研究新支话题的成因可以局限于这个局部语境，复杂度大大减小，有助于揭示话题延续和转换现象的本质。本文以句法成分和语义角色作为研究新支话题的切入点，考察新支话题充当什么样的句法成分时容易引出新支句，以及语义上存在什么样的关联。

本文使用了一个自建的广义话题结构标注的语料库，该语料库包括《围城》全文及其他小说、新闻、工作报告、说明文、法律法规等不同体裁的汉语篇章共约33万字，标点句35800多句。其中，发现新支话题501例，新支句777句（一个新支话题可以被多个标点句说明，故新支

句数目多于新支话题）。基于该新支话题和新支句的语料，本文从句法和语义层面对新支话题和新支句进行统计与分析，以期作为新支话题研究的起点。

3 新支话题的语法功能与句法成分分析

语法功能方面，我们在语料库标注中发现，新支话题绝大多数是体词性短语，且多数指称具体的实体，如例（1）和例（6）；也有极少情况是谓词性短语，例如：

例（7）

①全国人大常委会...对全国人大制定的法律进行**部分补充和修改**，

②　　‖　　(但是)不得同该法律的基本原则相抵触。

例中①句的“部分补充和修改”原是谓词性短语，成为②句的新支话题后，强迫变成了指称性，指“补充和修改的内容”。但这种情况极为少，且其指称的事件也是具体事件，容易从陈述性变为指称性。

句法成分方面比较复杂，以下我们全面将考察新支话题所充当的句法成分，并对已发现的不同句法成分举例分析。以下例子如无特殊说明，都出自钱钟书的小说《围城》。

（一）主语

从句主语，主谓谓语句小主语，状性成分起始句主语都有可能成为新支话题的主语。

1）从句主语

成为新支话题的从句主语包括宾语从句主语和补语从句主语。后续的标点句只对从句主语说明，不能够与主句构成话题—说明关系。此类共有29例，占5.8%。其中，宾语从句28例，补语从句1例，例如：

例（8）

他一见唐小姐，

便知道**她**今天非常矜持，

‖　毫无平时的笑容，

‖　出来时手里拿个大纸包。

（*说明：从本例起，我们主要关注新支话题（加粗部分），如无特殊需要，不再在标点句前标示序号。）

本例中，新支话题“她”是“知道”的宾语从句的主语。后续两个标点句“毫无平时的笑容”和“出来时手里拿个大纸包”说的都是“她”，即“唐小姐”，且不能与“她”之前的部分“他便知道”构成连成句法语义贯通的句子。

例（9）

这暖烘烘的味道熏得**方鸿渐**要泛胃，

‖　又不好意思抽烟解秽。

“方鸿渐”是“熏得”的补语从句主语，标点句“又不好意思抽烟解秽”是对“方鸿渐”的进一步说明，与主句无关。

2）主谓谓语句小主语

以主谓谓语句小主语作为新支话题的有38例，占7.6%，例如：

例（10）

今天苏小姐起身**我**都不知道，

‖　睡得像木头。

本例由多重主谓谓语结构组成，“我”是内层的主谓结构“我都不知道”的小主语，被标点句“睡得像木头”共享，但后句不能与“今天苏小姐起身”构成话题—说明关系。“我”成为新支话题。

3）状性成分起始句主语

虽然句首的状性成分不是传统的“话题”，但语料中的情况是，有些句首状性成分能被后续标点句共享，有些不行。这些句首状性成分不能为后续的标点句所共享时，该句的主语就成为了新支话题。这类情况共有16例，占3.2%。

例（11）

无论如何**他**决不会一翻脸就走的；

‖　来得困难，

‖　去也没有那么容易。

主语“他”被后续的标点句“来得困难”以及“去也没那么容易”所谈论，但副词性成分“无论如何”不能够被共享，“他”成为了新支话题。

（二）动词宾语

动词宾语充任的新支话题在新支话题中最普遍，共有364例，占所有新支话题的72.7%。按宾语出现的位置，又可进一步分为：句末动词宾语、兼语句兼语、连谓句非句末宾语。

1）句末动词宾语

该类新支话题最多，有318例，占63.5%。例如：

例（12）

李梅亭拉开**抽屉**，

‖ 里面是排得整齐的白卡片

本例首句是主动宾句，“抽屉”是“拉开”的宾语，位于句末。后句进一步说明“抽屉”的内容，不能与“李梅亭”等形成话题—说明关系，“抽屉”成为新支话题。

例（13）

他打了两下**门**，

‖ 没人来开。

本例首句是主动宾句，“门”是“打”的宾语。后句是对“门”进行说明，与前面都“他打了两下”都没有关系，“门”成为新支话题。

2）兼语句中的兼语

兼语句的兼语成分既是述宾短语的宾语，又是主谓短语的主语。兼语成分作为下一个标点句的新支话题有37例，占7.4%。例如：

例（14）

生产队派**他**干什么他就干什么

‖ 从不计较工分报酬。

兼语“他”是新支话题，在首句中既作“派”的宾语，又作“干什么”的主语。第二句对“他”进行说明，与“生产队”没有直接话题—说明关系。

3）连谓句非句末宾语

连谓句有若干个动宾结构，出现在句中间的动词宾语也可能成为新支话题。该类有9例，占1.8%，例如：

例（15）

我买了**一打新手帕**上船，

‖ 给船上洗衣服的人丢了一半。

首句是连谓句，“一打新手帕”是动词“买”的宾语，位于句中，第二句是对“一打新手帕”的进一步说明，与“我”无关。

（三）介词宾语

介词的宾语也可以成为新支话题，已发现的介词包括“把、被、给、对、在”等，共有39例，占7.8%。如上一节的例（6），又如：

例（16）

你这话给**我父亲**听见，

‖ 该说“孺子可教”了。

“我父亲”是介词“给”的宾语，被第二个标点句共享，但“你这话”不能与后续标点句构成话题—说明关系。

（四）状语

状语成为新支话题只有1例，占0.2%，但这例比较特殊，是名词作状语。

例（17）

只听得阿丑半楼梯就**尖声**嚷痛，

‖ 厉而长像特别快车经过小站不停时的汽笛，

(跟着)号啕大哭。例中“尖声”是名词，后续标点句是对“尖声”的比喻，与前面的话题“阿丑”无关。而一般的状语是副词或形容词，暂没有发现这类词语充当新支话题的情况。

（五）嵌套较深的句法成分成为新支话题

有些新支话题较为特殊，从句法成分上看，结构层次很深。比如宾语的定语成分，甚至是定语的一部分作为新支话题。但比例不大，有14例，占2.8%，例如：

例（18）
丈夫是**女人**的职业，
‖　　没有丈夫就等于失业
例（19）
我这话说在**你**耳朵里，
‖　　不要有了新亲，
把旧亲忘个干净！
例（18）的新支话题“女人”是系动词“是”的宾语的定语，例（19）的新支话题“你”嵌套得更深：介词“在”的宾语“你耳朵里”是名词—方位词短语，“你”是的其中名词短语的表领属的定语。

（六）小结：新支话题句法成分的特点

综上所述，语料库中新支话题句法成分统计情况详见表2。

新支话题句法位置			数量	比例
主语（83 例，16.6%）	从句主语	宾语从句主语	28	5.6%
		补语从句主语	1	0.2%
	主谓谓语句小主语		38	7.6%
	状性成分起始句主语		16	3.2%
动词宾语（364 例，72.7%）	句末动词宾语		318	63.5%
	兼语句兼语		37	7.4%
	连谓句非句末动词宾语		9	1.8%
介词宾语			39	7.8%
状语			1	0.2%
嵌套更深的成分			14	2.8%
总计			501	100%

Table 2: 新支话题句法成分统计

从以上众多例子可以看出，新支句无论从语义上还是句法形式上，与定语从句都有一定的相似之处，主要是对新支话题进行补充描述，所以形成了一种临时的话题转换。从新支话题的句法成分分类可以看出，位于句末的动词宾语作为新支话题的情况最显著，占63.5%，其他都不超过8%。这是因为，句末的动词宾语是新信息最显著的地方，容易产生进一步说明的需求，比其他地方更容易变成新支话题。位于其后的是介词宾语（7.8%），一种可能的解释是介词保留了一定动词的性质，依然能引出新的事物进一步描述；兼语句的兼语（7.4%）类似。再次是主谓谓语句小主语（7.6%）和从句主语（5.6%），它们本身就是陈述的主体，形成新支句原因是由于新支话题前面的部分不能作为共享话题延续到后句，从而形成了局部的话题转换，在机制上与动词宾语、介词宾语引出的新支句有所不同。总的来说，我们能看出，新支话题所处的句法位置总的来说是多样的，既可以在句中，也可以在句末；既可以是主语，也可能是动词的宾语，还可以是介词的宾语，甚至状语；既可以是单句的某些成分，也可以是从句的某些成分，还可以是嵌套在某一句子成分中的一部分。可见，在篇章的叙述过程中，对事物进一步说明的需求是很强烈的，只要语境确实需要，句法位置限制不大。

但是，我们也发现，就目前的语料看，也存在一些句法成分没有充当新支话题的情况，比如间接宾语，而且人为要造出间接宾语作为新支话题的句子也不容易，例如：
例（20）
?小明送了**小红**一支玫瑰花，
‖　　开心了一天。
例中似乎很难将“开心了一天”的话题理解为“小红”，或者说“开心了一天的”话题是“小明”还

是“小红”至少是有歧义的。如果要对“小红”进行说明，最自然的方法应该是将“小红”重复一遍，即“小明送了小红一支玫瑰花，小红开心了一天。”究其原因，我们认为，间接宾语要求一个能带双宾语的三元动词；而动词的三个论元都围绕同一个事件，且间接宾语和主语一般都指人；此时，如果后续标点句描述的是人而且以零形式回指某个论元，那么主语和间接宾语将存在竞争（如上例的“小明”和“小红”），而主语在句法和认知上位置都比较突显，导致主语更容易被后续标点句说明，间接宾语较难成为新支话题。

需要注意的是，以上的比例统计，只是对已发现的新支话题的句法成分进行分类统计，而不是这些句法成分成为新支话题的比例。就某个句法成分来说，不作为新支话题的情况必然比成为新支话题的可能性大得多，这从新支话题本身的数量和占比就可以看出来。所以，新支话题依然是一种特殊的话题转换现象。

4 新支话题的语义角色分析

由于新支句是对新支话题的进一步描述，那么新支话题在原句中的语义角色，能够在一定程度上反映该成分需要被进一步描述的语义动因。

在501例新支话题中，大部分新支话题位于动词谓语句中，但有情况两种例外。

一是新支话题的所在句（/从句）是形容词谓语等的非动词谓语句，这类有11例，占2.2%，其语义角色通常是主事，比如例（8），又如：

例（21）

我看她年轻得很，

‖ 是不是在念书?

“她年轻得很”是形容词谓语句做“看”的宾语从句，“她”是从句主语，成为新支话题，语义角色是主事。后句与主句无关。

二是新支话题不是句中的直接论元，而是宾语的定语等句法成分嵌套较深的情况，这类新支话题我们暂不讨论它的语义角色问题，这类有14例，占2.8%。

其余476例新支话题的所在句（/从句）都是动词谓语句。以下，我们讨论这些新支话题与所在句（/从句）中主要动词的语义角色关系。语义角色体系参考袁毓林（2008，2013）的论元结构理论，分为必有论元和非必有论元；必有论元包括主体论元（施事、感事、经事、致事、主事）和客体论元（受事、系事、结果、对象、与事）；非必有论元包括凭借论元（工具、材料、方式、原因、目的）和环境论元（时间、处所、源点、终点、路径、范围、量幅），共22种。对于动词论元的语义角色界定，我们使用袁毓林教授开发的《北京大学现代汉语动词句法语义功能信息词典》，对新支话题进行语义角色标注。

以下，我们统计和分析各例新支话题的语义角色。

（一）主体论元

主体论元是新支话题时，句法上可以充当从句主语、主谓谓语句小主语、状性成分起始句主语等，少数是介词（被、给等）宾语。它们的论元角色有施事、感事、主事、经事，但没有出现致事成为新支话题。主体论元作为新支话题有111例，占22.2%。

1）施事

该类新支话题句法上通常是从句主语或者主谓谓语句小主语等，这类新支话题共有52例，占10.4%。施事作为动作的发出者，进一步描述是很自然的，成为新支话题主要是该词语前面的部分与后续标点句不能连成句的问题，导致了临时的话题转换。如例（11）的新支话题“他”（从句主语）是“翻脸”的施事，又如：

例（22）

大大小小的事全是**她**一手办理，

‖ 外表斯文柔弱，

‖ 全看不出来！

新支话题“她”是“办理”的施事。不过后续是对“她”的描述，与原句“她”办事不相关，形成了新支话题。

2）感事

感事类新支话题情况与施事类似，但对应的动词通常的感知认知方面的。这类语义角色的新支话题共有13例，占2.6%。除了从句主语，还有部分是“给、被”等介词的宾语，如也成为感事的新支话题。如例（9）的新支话题“方鸿渐”是补语从句中“泛胃”的感事，例（10）中新支

话题“我”是小谓语核心动词“知道”的感事，例（16）的新支话题“我父亲”（“给”的宾语）是“听见”的感事。

3）经事

由于论元语义角色为经事的动词较少，经事类的新支话题也较少，只有2例，占0.4%。主要是各类从句主语，例如：

例（23）

鸿渐早像**箭猪**碰见仇敌，

‖ 毛根根竖直，

本例宾语从句主语“箭猪”是“碰见”的经事。

4）主事

主事类新支话题数量较多，有45例，占9.0%。一来是论元的语义角色为主事的动词较多；二来除了主动宾句的主语外，存现句（“处所+动词+主事”）的宾语也属于主事。存现句的主事不仅因为是句法上的宾语，更在于存现句在语义上引出了一个新的事物，具有强烈的对该事物进一步说明的倾向，常常成为新支话题。如例（1）的“一幅……大画”，又如：

例（24）

紧靠讲台的记录席上是**一个女学生**，

‖ 新烫头发的浪纹板得像漆出来的。

本例“一个女学生”是存现句的主事，后面对她的说明，与其处所“记录席”无关，故形成新支话题。

通常来说，主体论元继续被后续标点句说明是很自然的，作为主语时，大多是一种话题延续而不成为新支话题。它们之所以成为新支话题，是因为句首的其他成分只对本句起作用，不能被后续的标点句共享。除此之外，存现句的宾语是主事，也是主体论元。

（二）客体论元

客体论元成为新支话题的情况是比较普遍的，大部分动词宾语和介词宾语属于此类，共有310例，占61.9%。

1）受事

受事经常在动词宾语位置成为新支话题，如例（12）中新支话题“抽屉”是动词“拉开”的受事，例（13）中新支话题“门”是动词“打”的受事。有时是“把、将、给”等介词的宾语成为新支话题，如例（6）的“五人”是“吓”的受事。受事类的新支话题有84例，占16.8%。

2）系事

系事类的新支话题最多，有130例，占25.9%。系事类新支话题的动词主要是表等同的“是”或表比喻的“像”等，这类动词的客体论元常常因与主体论元相同或相似性而被引入到篇章中，成为语篇中的新事物，很容易有进一步说明的需求。例如：

例（25）

那张是**七月初的《沪报》**，

‖ 教育消息栏里印着两张小照，

“七月初的《沪报》”是关系动词“是”的系事。这里“是”表等同，如果说后句说明的是“那张”也未尝不可，但是看作对“《沪报》”的进一步说明，可能更加贴切。

3）结果

结果类新支话题有47例，占9.4%。语料中发现，结果论元非常容易成为新支话题，当动词创造出一个事物之后，较容易有对这个新事物有进一步说明的需求，例如：

例（26）

阿丑写了**“大”字和“方”字**，

‖ 像一根根火柴搭起来的。

本例“‘大’”字和‘方’字”是“写”的结果，后句对这一结果进一步说明。

4）对象

对象类新支话题有37例，占7.4%。引出对象论元的动词通常是感知或认知类的动词，如“看”“听”“知道”“喜欢”，以及介词“对”的宾语等。通过主体论元的观察、感受，向语篇引入了新对象，也有一定的进一步说明的倾向。

例（27）

①鸿渐偷看**苏小姐**的脸，

②　　‖　　光洁得像月光泼上去就会滑下来，
③　　‖　　眼睛里也闪活着月亮，
④　　‖　　嘴唇上月华洗不淡的红色变为滋润的深暗。

本例由“偷看”引出了新事物，形成新支话题。其中，新支话题有两重。②句“光洁得……滑下来”指向“苏小姐的脸”，③至④句说的是“苏小姐”。都与“鸿渐”无关，形成多重的新支话题。用换行缩进方式表示的话题结构，把这种话题共享的层次直观地展示了出来。

5）与事

与事类新支话题有12例，占2.4%。语料中与事当新支话题的情况都多数是动词宾语，少数是介词宾语，最常见的间接宾语的与事没有成为新支话题。例如：

例（28）

我前天碰见**周厚卿的儿子**，
　　‖　　从前跟老大念过书，
　　‖　　年纪十七八岁……

本例“碰见”的客体论元是与事，后面几句都是围绕“周厚卿的儿子”的说明，与“我”无关，与事成为新支话题。

总的来说，客体论元成为新支话题是最常见的情况。

（三）凭借论元

极少数凭借论元的论旨能够成为新支话题，语料库中只发现1例方式论元成为新支话题的情况，占0.2%。这例就是例（17）“尖声嚷痛”的状语“尖声”成为新支话题，“尖声”的“嚷痛”的方式。

其他凭借论元（工具、材料、原因、目的等）暂未发现成为新支话题的情况。可见，凭借论元要作为新支话题引出新支句比较困难。

（四）环境论元

环境论元中，发现处所论元和终点论元作为新支话题的情况，共14例，占2.8%，语料库中没有发现时间、源点、路径、范围论元作为新支话题的情况。

1）处所

处所类的新支话题共6例，占1.2%，例如：

例（29）

辛楣等睡在**一个统间**里，
　　‖　　没有床铺

本例的“一个统间”是“睡”的处所论元，后句是对“统间”的说明，与“辛楣”无关。

2）终点

终点类的新支话题共8例，占1.6%，例如：

例（30）

明天上午他们到了**界化陇**，
　　‖　　是江西和湖南的交界。

本例“界化陇”是“到”的终点论元，后句是对“界化陇”，与“他们”等话题无关。

这些环境论元作为新支话题的例子，基本上都位于句末，且句中的核心动词本身与处所、范围等相关，实际上这些环境论元对于主要动词来说，是核心论元。

（五）具有多种语义角色的新支话题

有些新支话题在句中身兼多种语义角色，对不同动词承担不同语义角色，在句法上主要是兼语句的兼语，或连谓句某个动词的宾语，这类新支话题有39例，占7.8%。

例如，例（14）中新支话题“他”是动词“派”的受事又是动词“干”的施事。又如：

例（31）

他拣出bf 一叠纸给鸿渐看，
　‖　　是英文丁组学生的公呈。

本例中“一叠纸”是“拣出”的受事，又是“看”的对象。

（六）小结：新支话题语义角色的特点

综上所述，语料库中新支话题语义角色统计情况详见表3。

从语料库实例的统计和分析，我们发现，语义角色中客体论元最容易成为新支话题，共占61.9%，其次是主体论元，环境论元不太容易，凭借论元非常困难。其中，系事（25.9%）成

论元角色			数量	比例
必有论元（422 例，84.2%）	主体论元（112 例，22.4%）	施事	52	10.4%
		感事	13	2.6%
		经事	2	0.4%
		致事	0	0.0%
		主事	45	9.0%
	客体论元（310 例，61.9%）	受事	84	16.8%
		与事	12	2.4%
		结果	47	9.4%
		对象	37	7.4%
		系事	130	25.9%
非必有论元（15 例，3.0%）	凭借论元（1 例，0.2%）	工具	0	0.0%
		方式	1	0.2%
		材料	0	0.0%
		原因	0	0.0%
		目的	0	0.0%
	环境论元（14 例，2.8%）	时间	0	0.0%
		处所	6	1.2%
		源点	0	0.0%
		终点	8	1.6%
		路径	0	0.4%
		范围	0	0.0%
多重语义角色的新支话题			39	7.8%
非动词谓语句的新支话题			11	2.2%
非论元的新支话题			14	2.8%
总计			501	100.0%

Table 3: 新支话题语义角色统计

为新支话题比例最高，因为系事论元通常是“是”“像”等关系动词句子中的宾语，而“是字句”等多为静态描写，此时，引出新支句作进一步描写的倾向较强。其次是受事（16.8%）和施事（10.4%），这两种语义角色是与行为最直接相关的语义角色，对行为的发出者或者接受者有进一步说明的需求，且通常他们的句法位置较高，较容易被进一步说明。对象（7.4%）和感事（2.6%）论元的情况看与受事和施事相似，只是论元搭配的是感知、认知类动词，这类词语数量不多。然而，结果（9.4%）论元做新支话题的情况则较为特殊。带有结果论元的动词总体数量不多，但引出新支句数量较多，这是由于结果论元常常是由动词创制的新事物，通常第一次引入篇章，故对新事物容易形成进一步说明的需求。主事（9.0%）论元主要由于存现句的主事论元常常成为新支话题。其他语义角色成为新支话题的情况都较少，不足5%。还有一些语义角色，我们在语料中没有发现它们成为新支话题的例子。这些语义角色可能分为两类情况。一类是虽然受语料规模所限暂时没有发现该论元成为新支话题的例子，但可以自造例子。例如工具论元。

例（32）

我喜欢用这把刀切菜，

　　||　　特别锋利。

例中“这把刀”是工具论元，后句进一步说明“这把刀”的特点。另一类则在语料库中没有发现，且难以自造，如表原因、目的论元。

例（33）

他因为《海瑞罢官》在文革中受到冲击，那出戏被禁演了。

例中“受到冲击”是核心动词短语，“《海瑞罢官》”是介词“因为”的宾语，是原因论元，但难以成为新支话题。如果想以“《海瑞罢官》”为话题进一步说明，就需要以某种形式重复话题而不能零形式，如例中的“那出戏”。再如：

例（34）

两人为了儿子暂时离婚，儿子归了父亲。

本例的核心动词是“离婚”。介词宾语“儿子”既不是“离婚”的主体，也不是“离婚”的客体，而是“离婚”的目的，但难以成为新支话题。当“儿子”需要作为话题时，就得重复出现。还有一种由介词“除了”引出的宾语，不在通常所说的动词论元范围内，我们暂时称作排除论元。这种论元也难以充任新支话题，如：

例（35）

我们班除了几个女生其余都在，那几个女生回家了。

“几个女生”属于核心动词“在”的排除对象，这种被排除者的语义角色不能充当新支话题。如果需要成为话题，就得重复出现。我们认为，这些论元在整个句子中往往是一种背景信息，当前景信息出现后，背景信息很难再次被激活而作进一步说明。而已发现的充当新支话题的语义角色，往往在语义上与动词直接相关，是以动词为核心的事件中不可或缺的一部分。

5 实质语义对新支话题的作用

当我们进一步考察一些嵌套较深的新支话题，发现新支话题的引出不是取决于语句的表层语义，而是取决于语句的实质意义。由于修辞或其他语用原因，这种深层的语义有时跟表层语义不尽相同。

如例（18）中新支话题“女人”是关系动词“是”的宾语“女人的职业”的定语。但“职业”是“女人”的一个从属属性。“丈夫是女人的职业”也就是“女人的职业是丈夫”，或者干脆不要“的”，就是“女人职业是丈夫”。语义上“女人”是联系动词“是”的主体论元，因此后续标点句仍能以“女人”作为话题。

又如例（19），从字面上看，新支话题“你”句法位置嵌套得很深，不是“说”的客体论元（“你耳朵里”是“说”的处所论元）。但是这句话的表达的实质意义是“我警告你”，“你”是“警告”的受事。因此，“你”成为了新支话题，“耳朵”只是一种修辞手法。再如：

例（36）

你小心别讨了**你那位朋友**的厌，

‖ 一脚踢你出来，

从句法上看，动词宾语的定语“你那位朋友”成为新支话题，而且是离合词“讨厌”的插入成分，层次嵌套较深，“讨厌”的论元分析也很困难。但句子的实质语义是“你别招你那位朋友讨厌”，“你那位朋友”是致使动词“招”的对象，又是“讨厌”的感事。上述例子可见，形成新支话题的必有论元不苑囿于字面词语的必有论元，而是看说话人真正想表达的实质语义的必有论元。用实质语义方法分析新支话题的成因，只是初步尝试，还没有形成明确的概念和系统的方法，对实质语义形成新支话题的规律还有待进一步研究。

6 结语

本文介绍了新支话题和新支句的概念，将标点句句首的零形回指称为话题共享，根据话题共享情况，对篇章微观话题的延续和转换进行了分类，分为句首话题延续、句中子话题延续、完全话题转换、兼语话题转换、新支话题转换五种，认为新支话题是一种特殊的、临时的话题转换。

为进一步研究新支话题的特点，本文考察了33万字的语料，发现了501个新支话题，分析和统计了其句法成分和语义角色的分布情况。我们发现，句法方面，新支话题在句中的句法位置可以是：宾语从句或补语从句主语、主谓谓语句小主语、状语起始句的主语、句末宾语、连谓句非句末宾语、兼语句兼语、介词宾语、状语等。它们都能引出新句子进行说明。其中，63.5%是句末宾语做新支话题，其他都不超过8%，超过5%的情况有：介词宾语（7.8%）、

主谓谓语句小主语（7.6%）、兼语句兼语（7.4%）和宾语从句的主语（5.6%）。这是因为新支句的作用一般用于补充说明或进一步描述新支话题。句末宾语是新信息最显著的位置，因此被进一步描述的机会最大。值得注意的是，介词宾语做新支话题的频率也不低，“把、被、给、对、在”都可能引出新支话题，共39例，反映了介词仍保留动词的功能，介词宾语也具有引出新信息的特点。但间接宾语无法成为新支话题。语义方面，可以做新支话题的语义角色包括：大部分主体论元（施事、感事、经事、主事），所有客体论元（受事、系事、结果、对象），少数凭借论元（方式）、部分环境论元（处所、终点）。其中，最常见的语义角色是系事（25.9%）、受事（16.8%）、施事（10.4%）、结果（9.4%）、主事（9.0%）和对象（7.4%），其他都不超过3%（即不超过15句）。系事常用于描述新事物，结果则是创造出新事物，这些新事物被进一步描述可能性大，因此需要用新支句对其进一步说明。施事、受事、感受、对象等都是动词的最直接参与者，也存在被进一步说明的可能性。但是表原因、目的的论元和表排除义（“除了”）的论元都难以成为新支话题，他们在句中都是背景的信息，难以被进一步描述。

对于句法上嵌套得较深、结构上难以划入动词论元的新支话题，我们尝试着用说话人表达的实质语义来解释，说明在语义理解的层面，这些论元实际处于表层的位置。

总的来说，新支话题作为一种临时的话题转换，是出于对引入语篇的事物有进一步说明的语义需求。对新支话题进行说明的新支句，由于不能与外层话题形成话题延续，或多或少有一种补充说明或者插入说明的意思。但这种说明是有句法和语义限制的，并非句中任意句法成分或语义角色都能够被进一步说明，本文的研究发现，只有处于当前陈述的事件中最密切相关的前景信息，才可能被进一步说明，发生临时的话题转换。

话题延续和转换是一种语用现象，我们期望通过句法和语义的约束研究，将这种语用现象逐步落实到语义直至句法的形式上，从而为语言的科学化、形式化研究起到推进作用。本文目前还只是面向人的分析和操作，将来应逐步发现人工分析操作的形式依据，以便交由计算机去处理。当然，要达到这个目标还有很长的路要走。

参考文献

Givón, T. 1983. *Topic continuity in discourse: A quantitative cross-language study*, volume 3. John Benjamins Publishing, Amsterdam, The Netherlands.

曹逢甫. 2005. 汉语的句子与子句结构. 北京: 北京语言大学出版社.

陈平. 1987. 汉语零形回指的话语分析. 中国语文, (5):363-378.

方梅. 2005. 篇章语法与汉语篇章语法研究. 中国社会科学, (6): 165-172.

卢达威, 宋柔, and 尚英. 2014. 从广义话题结构考察汉语篇章话题认知复杂度. 中文信息学报, 28(5): 112-124.

屈承熹. 2006. 汉语篇章语法. 北京: 北京语言大学出版社.

宋柔. 2008. 现代汉语跨标点句句法关系的性质研究. 世界汉语教学, (2): 26-44.

宋柔. 2013. 汉语篇章广义话题结构的流水模型. 中国语文, (6): 483-494.

宋柔. 2018. 汉英短句复合体的形式结构. 揭春雨, 刘美君*(*编*)*. 实证和语料库语言学前沿. 北京: 中国社会科学出版社.

袁毓林. 2008. 基于认知的汉语计算语言学研究. 北京: 北京大学出版社.

袁毓林. 2013. 基于生成词库论和论元结构理论的语义知识体系研究. 中文信息学报, 27(6): 23-31.

眼动记录与主旨结构标注的关联性分析研究

单昊聪，周强
清华大学信息技术研究院
语音和语言技术中心，北京，100084
2390326004@qq.com，zq-lxd@mail.tsinghua.edu.cn

摘要

给定包含主旨概括句的汉语句群，针对该句群的内部结构标注是基于语言学的分析结果，而阅读句群时的眼动轨迹则蕴含着人的心理认知，两者的信息融合和内在关联性分析是该文主要工作。该文使用基于径向基函数支持向量机和递归特征消除的分类模型，根据标点小句片段对应的眼动指标数据预测该片段是否为包含主旨内容的关键信息，达到了0.76的准确率，并通过分析关键片段上眼动数据的分布特点，提取出对句群主旨概括信息区分度较好的眼动指标。

关键词： 眼动指标；文本结构标注；支持向量机；径向基函数；特征递归消除

Research on the correlation between eye movement feature and thematic structure label

Haocong Shan, Qiang Zhou
Center for Speech and Language Technology,
Research Institute of Information Technology,
Tsinghua University, Beijing, 100084, China.
2390326004@qq.com, zq-lxd@mail.tsinghua.edu.cn

Abstract

Given a Chinese sentence group that contains a theme sentence, the internal structure label of the sentence group is based on the results of linguistic analysis, while the eye movement trace of reading sentence group contains human psychological cognition. The main work of this paper is the information fusion and internal relevance analysis of structure label and eye movement. In this paper, a classification model based on radial basis function support vector machine and recursive feature elimination is used to predict whether the punctuation clause segment is the key information containing the thematic content according to the corresponding eye movement feature data. The accuracy of 0.76 is achieved. By analyzing the distribution characteristics of eye movement data on the key segment, eye movement features with good distinction for the thematic information of the sentence group are extracted.

Keywords: Eye Movement Feature , Text Structure Label , Support Vector Machine , Radial Basis Function , Feature Recursion Elimination

1 引言

人类使用眼睛运动来选择性地获取视觉信息再加工处理是一个很重要的认知过程。在人类阅读相关的心理学研究中，挖掘眼动数据所蕴含的信息是重要的方向。一方面，眼动数据包含着读者阅读行为的认知信息。例如不同教育背景读者的阅读能力的差异性可以反应在眼动模式中，更详细的研究显示不同读者的阅读策略等信息也可以依据眼动数据来判别 (Ablinger et al., 2014)。另一方面，阅读任务的不同也影响着眼动模式。例如无意识阅读和有目的阅读的眼动指标分布有着较明显的差异 (Reichle et al., 2010)。针对有目的阅读而言，不同阅读任务所对应的眼动模式也是不同的。例如在文本校对和文本理解两个不同阅读任务中，眼动数据所反应的注意力分配是不同的 (Kaakinen and Hyönä, 2010)。在文本主旨相关的任务中，有融合眼动数据的多主旨文本标题提取任务研究 (Hyönä and Lorch, 2004)，该研究主要考察了标题句的眼动数据，并详细分析了眼动指标的特性，但是该研究所选取的文本缺少具有语言学支撑的结构标注，同时缺少衡量眼动指标的重要性分布。也有单主旨文本摘要提取的眼动数据研究 (Xie et al., 2019)，但是该研究仅仅专注于眼动轨迹，没有完整的眼动指标分析，缺乏认知心理学依据。

在分析阅读的眼动研究中，主要包含注视和眼跳两种基本的眼动现象 (闫国利et al., 2013)，第一类是与眼睛何时移动有关的时间维度眼动指标，例如遍注视时间，其反映了读者处理特定信息所需要的时间；另一类指标是与眼睛移动位置有关的空间维度的眼动指标，具体包括眼跳距离，注视位置，注视顺序等，其中注视位置反映了读者在特定时间内所处理的信息内容在上下文中的位置，注视顺序则反映了读者处理信息的次序。眼动数据在更深层次上还反映了读者在词汇和句子加工理解过程中的复杂心理认知阶段。Rayner (1979)在研究中使用了针对不同词汇的“首次加工有多次注视的首次注视时间”这一指标，较好地反映了读者对于词汇的早期加工情况，同样地第二次注视时间也被看做是一个较好反应复合词早期加工的指标 (Pollatsek et al., 1986)。第二遍阅读时间和回视时间等则反映了读者对于复杂句子中词语的后期处理情况。后期处理的指标往往反映了被试在兴趣区（包含字、词或句子等的区域）遇到困难后的再处理和分析过程。所以时空维度的眼动指标数据可以反应人在阅读语篇时候的心理认知过程。这样的心理认知过程蕴含着人对于语篇中词语的辨识、句子的解读和整个语篇的理解。

该文旨在融合基于语言学的主旨结构标注和基于认知心理学的眼动数据指标信息，建立模型来探索两者的关联性，并考察眼动指标的显著性分布，提取出在主旨判断中重要的指标并做心理学分析，以填补目前的研究空白。该文主要面临三个问题。首先是眼动数据来源于人对文本的阅读，选择合适的文本供被试阅读是很重要的实验前提，需要从语言学的角度将文本做处理，以能够配合眼动数据做可计算的数据集。其次是如何根据数据集选择合适的计算模型，在保证计算模型的精度情况下，需要挖掘出眼动指标数据对于语篇分析的指导意义，提取出较为重要的眼动指标组，从认知角度给予语篇意义分析一定的帮助。最后是如何设计实验，需要考虑在研究历史中眼动数据的特点合理地设计实验以保证实验的科学性。

对于文本的选择问题，该文的文本研究对象为句群 (吴为章, 2000)。较广泛意义上的语篇而言，句群在语义上有逻辑联系，在语法上有结构关系，一般都会有一个明晰的中心意思即主旨。主旨是句群全部内容表达的基本观点，是其叙写议论的基本描述目标。按照其归纳形式不同，大致可以分为显性表示和隐性表示两类。在显性表示类中，句群主旨直接落实在句群的主题句中，可以通过对主题句内容的直接引用或适当改写得到。在隐性表示类中，句群中不存在明显的主题句，但大多存在包含句群关键信息的核心句，通过对这些核心句子内容的概括提炼，可以生成合适的主旨内容描述句。在连贯话语中，句群是相对独立的句法语义结合体，可以从语流中切分出来。在语篇结构分析标注研究中，它起着承上启下的作用。按照其中描述内容不同，大致可以分出记叙、议论、说明等功能类别，它们具有不同的结构描述模式。对于这样的文本，其含有的信息价值较高，眼动数据的可研究性也更为丰富，并且能够和文本摘要任务很好的契合，因为能够依据语言学研究对文本做主旨结构标注，以产生可以为模型训练的数据。所以此时就确定了研究的主要范围为汉语句群片段的眼动数据和主旨标注数据之间的信息融合和关联性。针对阅读汉语句群片段，一方面通过被试阅读该汉语句群，记录下眼动数据，另一方面通过基于语言学的句群主旨结构分析，可以得到人工标注数据，这两部分的数据结构细节在第2节数据介绍中做详细的说明。

对于模型的选择问题。在获得了数据集后，需要针对此数据集合理地选择模型做训练和预

测。随着统计机器学习方法和神经网络的发展 (Lauzon, 2012)，以及数据量越来越大的真实世界数据现象，机器学习方法越来越广泛地应用到实际数据分析中。虽然神经网络的精确率和性能要比统计学习方法更好，但是由于统计学习方法有着更为深厚和坚实的数学理论基础，所以在可解释性 (Molnar, 2020)方面统计学习方法更胜一筹。该文研究的重要方向是眼动数据指标的描述特点，分析其对于文本主旨标注的关联性，在保证模型精度的同时，还需要对关联性有较好的解释，所以对于模型而言选择了统计机器学习方法中较为经典的支持向量机 (Suykens and Vandewalle, 1999)模型，对更为详细的模型设计在第3节给出。

对于实验的设计问题。在获取了数据集并确定了模型后，需要考虑影响实验的因素来合理地设计。首先，对于人的阅读认知过程而言，不同的人有着不同的阅读理解水平，并且该阅读水平也反映在了眼动数据的模式上 (Hyönä et al., 2002)。所以为了减少眼动数据的误差，需要考虑不同背景的被试对于结果的影响，并提取出较优的被试集合来达到更高的准确率。其次，考虑到在人工文本主旨标注过程中，有一个环节是要求总结出句群的主旨信息并添加到标注数据表格中，这一环节可以看作是带主旨总结目的的阅读过程。普通的自由阅读和带主旨总结目的阅读的心理学认知是不同的 (Kaakinen and Hyönä, 2010)，其差异性也会反应在眼动数据中，所以需要进行对比实验，即自由阅读和带主旨总结任务阅读两个阶段的眼动实验。通过对比来分析指标的差异性。实验阶段的详细信息和实验结果在第4节给出。

2 数据介绍

该文的原生数据分为两部分，一部分是微博和百科类句群的人工标注数据，另一部分是对于每个语篇的眼动数据。在获取了两部分数据后，将处理后的数据做匹配得到可训练的数据集。

2.1 微博和百科类人工标注数据

对于微博和百科类的语篇而言，对其进行结构主旨标注 (周强, 2019)来分析挖掘其中主旨信息的不同表现模式。

首先根据其中的各个结句点号（句号、问号、叹号、省略号等）自动切分出各个句子（Sentence），在这个过程中，需要注意标号对（双引号、括号等）内部的结句点号的句子分隔作用。然后在各个句子中进一步依据其中的分隔点号（主要是逗号、分号和冒号）自动切分出各个标点小句片段（Punctuated Clause Segment, PCS），至此PCS层小句产生。

下面表 1给出一个具体的微博短讯标注实例对此进行解释说明：

句子序号	句子和小句	主旨描述
1	前天晚上，武昌警方调集数百名警力， 对武汉大学校园周边治安环境进行整治。	武昌警方在武大校园 周边查验8名涉嫌吸毒男子。
1-1	前天晚上，	
1-2	武昌警方调集数百名警力，	
1-3	对武汉大学校园周边治安环境进行整治。	
2	在记者跟随下，一队民警从武大近邻的军悦 假日酒店内揪出8名涉嫌吸食麻果的男子。	
2-1	在记者跟随下，	
2-2	一队民警从武大近邻的军悦假日酒店 内揪出8名涉嫌吸食麻果的男子。	
3	他们都坚决否认吸毒，民警当面查验， 证实其中6人吸了毒。	
3-1	他们都坚决否认吸毒，	
3-2	民警当面查验，	
3-3	证实其中6人吸了毒。	

表 1: PCS层切分实例：微博短讯（武汉晚报-20121020-1600）

在表 1中，可以看到该武汉晚报微博短讯里包含有三个句子，每个句子中又包含有多个PCS层小句片段，例如在句子序号为1的句子中，包含有1-1，1-2和1-3三个PCS层小句。主

旨描述列信息则是从句子形成的句群中抽取关键信息总结得到，此时完成了对整个语篇基于PCS层细粒度的划分。然后需要进一步确定PCS层小句为关键或非关键信息。关键信息的标注则可由结构主旨标注相关信息自动得出。

句子序号	句子结构序列	小句结构序列	关键信息
1	ES		
1-1		TIM	0
1-2		FE-1	1
1-3		FE-1	1
2	ES		
2-1		BE-2	0
2-2		FE	1
3	OS		
3-1		BE	0
3-2		FE-1	0
3-3		FE-1	0

表 2: 实例所对应的结构标注信息

在表 2中，展示了与表 1中PCS层一一对应的主旨结构标注信息，关键信息列中的标注信息需要直接用于实验。其中标注为1代表关键信息，标注为0代表非关键信息。而关键信息列的标注信息由句子结构序列和小句结构序列的标注信息而得到。例如句子1和2中，句子结构序列标注为ES（Event Sentence）事件句，表示其包含句群的部分关键信息。进一步考察其中的小句结构序列标注，可发现FE代表前景小句，包含了句子的主要事件描述内容，所以相应的PCS小句的关键信息标注为1。而句子3的句子结构序列标注为OS（Other Sentence），不包含句群关键信息，只是对关键句的补充解释或者阐释说明，所以其后的相应PCS的关键信息标注为0。

利用上述方法对1172个微博短讯和1300个汉语新闻百科句群的人工标注数据进行自动提取，得到可用于实验的标注数据，以匹配眼动数据做为实验数据集。

2.2 语篇眼动数据

眼动实验一共有125位被试者，每位被试者要求分两个阶段使用眼动仪阅读2.1中的微博和百科语篇。一个阶段是正常自由阅读语篇，简称*article* 阶段；另一个阶段需要被试者在阅读完毕后总结语篇的主旨，简称为*gist*阶段。同时每个语篇需要至少三位以上的被试做实验，可以获得至少3份不同的被试记录数据，以为后面的数据处理减少误差做准备。

实验通过眼动仪产生注视眼动记录报告，所有不同被试的相关数据都会保存在一个EXCEL表中，可以根据*Eyelink*用户手册描述自动导出不同被试的相关数据报告 (Research, 2010)。报告中较为重要的是兴趣区数据和眼动指标数据。该文兴趣区设置为句群中的PCS层次小句片段，所以实验中的兴趣区即为PCS层小句。针对眼动指标数据，由于存在冗余的指标，需要对部分指标做选取和舍弃。这涉及特征工程中的特征选择问题。在特征工程的数据预处理中，对于实际意义差距很大的特征数据一般采用独热编码 (Jiang et al., 2016)。而对于眼动注视数据而言，其特征大部分是与时间尺度相关的变量，其实际意义相近或相同，例如第一次注视时间等；另一部分是次数相关的指标，例如注视该兴趣区次数，此情形下采用*z-scores*标准化数据 (Abdi, 2007)较为合理。

同时需要保证每个特征的数据在标准化后的数值具有相同的度量标准，那么对于个别眼动特征而言，例如开始时间、结束时间，其数值为绝对时间点而非可以衡量时间长短的区间值，所以此类特征也要舍去。另外在实验中，被试和仪器会有短暂的数据未记录的现象，多数表现为数据为空值，如果按照上述的特征选择标准，结合特征的实际意义，这些缺少值可以直接赋予0。例如若注视时间为0，则代表被试未注视该兴趣区；若注视次数为0，则同样代表被试为注视该兴趣区。对于分类模型而言，这也是减少误差的有效方法。

另一方面眼动特征主要有注视和眼跳两个方面。注视数据报告中每一行表示注视点的各种对应指标值，按照注视点的发生时间顺序排列。眼跳数据报告每行则表示一个眼跳的各种对应指标值，按照每个眼跳的发生时间顺序排列。注视数据和眼跳数据实际上属于眼动指标中时间

和空间两个不同维度的信息。该文以注视相关的眼动指标为考察重点，在未来的实验中会加入眼跳相关的数据分析。

经过上述过程就可以在PCS层次建立起眼动兴趣区注视相关指标。而同时句群数据的结构主旨标注也是基于PCS层次，这样两者就形成对应关系，以进行后续的信息融合和关联性研究。至此可以得到16个筛选出的眼动指标，相应的实际意义如表 3所示：

指标名称	含义
IA_DWELL_TIME	所有注视点总注视时间
IA_FIXATION_COUNT	该兴趣区的所有注视点个数
IA_RUN_COUNT	注视该兴趣区次数（注视点落入后离开算注视一次）
IA_LAST_FIXATION_RUN	该兴趣区阅读遍数
IA_FIRST_RUN_DWELL_TIME	第一遍进入兴趣区到离开的所有注视点时间之和
IA_FIRST_RUN_FIXATION_COUNT	第一遍阅读这个兴趣区的注视点个数
IA_FIRST_FIXATION_DURATION	第一个注视点的注视时间
IA_SECOND_RUN_DWELL_TIME	从第二次进入兴趣区到离开所有注视点时间之和
IA_SECOND_RUN_FIXATION_COUNT	第二遍阅读这个兴趣区的注视点个数
IA_SECOND_FIXATION_DURATION	第二个注视点的注视时间
IA_THIRD_RUN_DWELL_TIME	从第三次进入兴趣区到离开所有注视点时间之和
IA_THIRD_RUN_FIXATION_COUNT	第三遍阅读这个兴趣区的注视点个数
IA_THIRD_FIXATION_DURATION	第三个注视点的注视时间
IA_LAST_RUN_DWELL_TIME	最后一次进入兴趣区的总时间
IA_LAST_RUN_FIXATION_COUNT	最后一次进入兴趣区的注视次数
IA_LAST_FIXATION_DURATION	最后一个注视点的注视时间

表 3: 眼动指标说明

3 模型设计

该文基于被试眼动数据判断所对应的PCS层小句是否为关键信息句，来挖掘不同的眼动指标的重要性排序以及相应的心理学依据。根据第2节的实验数据描述可以清楚地看到需要训练一个二值分类器，依据此分类器对相应特征的重要性做排序。

另外，还需要考虑一个很重要的数据特征，即数据是线性可分还是非线性可分。在2.2的表 3中可以看到，眼动指标比较复杂。这些指标大致分为四个主要类型：第一类指标为注视点总特征，例如所有注视点总注视时间以及个数；第二类指标为第一遍进入兴趣区相关的注视时间和注视点个数；第三类指标为第二、三遍阅读兴趣区的注视点信息；第四类为最后一次进入兴趣区的总时间。实际上这些指标之间可能有着较为复杂的信息交叉。例如所有注视点的总注视时间和所有注视点的个数可能会成正相关。因为在普通阅读中，可能存在着注视点个数越多，被试看的时间越久的情况，实际上该文计算了这两个指标的相关系数 (Lee Rodgers and Nicewander, 1988)为0.35，所以其并不是完全的正相关。可能出现注视点个数较少，但是每个注视点的时间较久导致总注视时间大的情况。尽管如此这两个指标之间的信息还是存在一些交叉，其他几类的指标仍然可能存在着类似的情况。所以在实验的过程中需要考虑数据是否是线性可分的，如果非线性可分，那么应该有线性和非线性的结果比较，所以此模型应该能同时处理线性可分数据和非线性可分数据。

另外对于特征的重要性衡量也是需要考察的很重要的方面。因为在认知心理学中，对于眼动指标的认知方向的研究较为丰富。如果通过眼动数据来判断该眼动数据对应的小句片段是否为关键信息，那么也应当分析该眼动数据是如何起作用的。例如眼动数据中哪一部分指标起着重要作用，哪一部分指标的作用较小，以此获得不同眼动指标的显著性分布。依据该分布可以提取出对于主旨结构信息比较重要的指标，并结合认知心理学做分析。所以模型需要具备衡量指标重要性的功能。

综上可知模型需要根据PCS层次上的眼动指标数据来判断该片段是否关键，无论这些数据是线性可分或非线性可分，同时还需要对这些眼动指标如何作用的过程有着较为清晰的展现，能够可视化指标的重要性排序以及显著性分布。结合这些模型需求和数据特点，该文选择支持

向量机SVM(Support Vector Machine)和递归特征消除RFE(Recursive Feature Elimination)相结合的模型：SVM-RFE。因为支持向量机本身是一个数学理论基础较为完善的模型，同时对于近似线性可分的数据有较好的处理结果，并且对于非线性可分数据而言，可以通过核函数等技巧来解决。另外支持向量机结合递归特征排序可以对特征的显著性做分析。

实际上，SVM-RFE算法是一个经典的特征排序方法，适用于数据是线性可分或者近似线性可分的情况。它使用SVM线性核，并且通过RFE包装方法来将特征做筛选和排序。筛选排序的依据就是SVM训练过程的参数w，w反映了所有特征的权重。该权重在空间中反映了每个维度的特征对结果的影响情况，所以可利用该权重的大小对特征重要性做排序。对于SVM-RFE算法 (Guyon et al., 2002)本身，其正确性和可靠性已经在分类患者和读者等实验中有了很好的验证：特征权重的大小一定程度地反映了该特征对于模型预测结果的重要程度，对于结果没有较大影响的特征可以做筛减。

至此，该文确定采用SVM-RFE算法做训练和分类。在实验的结果中发现，对于注视眼动指标而言，线性SVM不能很好地区分出关键句和非关键句。在下文第4节结果分析里以及上述的数据分析中也可以推测眼动指标数据并非是线性可分的，所以线性分类模型不适用于注视眼动类数据。

对于非线性可分数据而言，SVM中的非线性核，例如径向基核函数RBF(Radial Basis Function) (Musavi et al., 1992)，可以提供一种处理此类型数据的方法。该核技巧的思想是将原数据的有限特征映射到更高维度的特征空间中，使得数据在更高维度的特征空间中能线性可分。

但是该文需要研究不同眼动特征对于主旨判断的认知意义，而非线性RBF核函数的主要缺陷是没有线性核函数中的特征权重w。该方法专注于训练后的精确度结果，对于映射后的高纬度特征空间的函数不能够显式地表达，其背后原因是将RBF核函数做泰勒展开后，得到的是无穷维的特征映射函数 (李航, 2019)。所以该方法不能做特征筛选和排序。

针对这个缺陷，很多研究提供了改进办法。例如有的研究使用高斯核函数，专注于改进核函数的参数选择难题，并结合RFE方法来抽取出非线性特征 (Xue et al., 2018)。有的研究引入对角矩阵，将特征参数放置于矩阵对角线中，其余的矩阵位置归零，并通过迭代的方式将映射后的特征不断削减，最终得到有限个特征的排序 (Mangasarian and Kou, 2007)。这些研究进展大多不需要考察指标的实际意义，只需要专注于准确率的提升。但是对于该文而言，指标的重要性分布以及其在心理认知中的含义是考察的重点。

该文使用了对于RBF核函数做变形展开的处理思想 (Liu et al., 2011)，将核函数的表达形式改写为内积形式，同时改写SVM的目标优化函数，将核函数和目标函数结合成包含内积形式核函数的SVM目标优化函数。然后对此函数做麦克劳林展开，最后对隐函数求偏导，可以得到一个计算特征权重的表达式。此类表达式的处理类似于线性核函数中的参数w，所以非线性RBF核函数的特征权重可以计算。另外由于非线性RBF在训练的过程中有超参数，该文在训练过程中加入了交叉验证方法以得到最佳的参数值。

4 实验结果分析

实验方法是将所有的PCS层小句对应的眼动数据以及相应的关键标注信息作为输入，用3节中的算法处理训练模型来判断该小句是否为关键信息，同时衡量眼动指标的显著性分布。实验主要考虑三个变量，首先是线性和非线性SVM对于实验结果的影响，其次是不同被试以及不同阅读任务的影响，后两个变量对于认知心理学判断有着重要的意义。

在认知心理学中，不同受教育程度的被试有着不同的阅读认知水平，不同被试的受教育程度影响着对文章意义把握的准确率 (Lou et al., 2017)，这些信息蕴含在眼动数据中。所以需要考虑不同被试对于实验准确率的影响。另一个重要的变量是不同的阅读阶段，在2节中可看到被试参与两个不同的阅读阶段，自由阅读简称为*article*阶段，和总结主旨简称为*gist*阶段，需要考察这两个阶段的阅读特性和眼动特征。

该文根据这三个因素设计了三个实验。实验一侧重于线性和非线性SVM-RFE的比较分析；实验二建立在实验一的结果之上，在确定使用非线性SVM-RFE模型后，考察不同被试的影响，选取一组质量最好的被试作为数据集；实验三建立在实验二的结果之上，在选定了质量较好的被试作为数据集后，着重*article*和*gist*阶段的结果分析。

4.1 实验一结果分析

对于实验一而言，训练了线性和非线性SVM两种模型，其中线性SVM中惩罚参数c为1，非线性RBF核函数中$gamma$为0.1，均采用交叉验证方法得到。两种模型的结果如下所示：

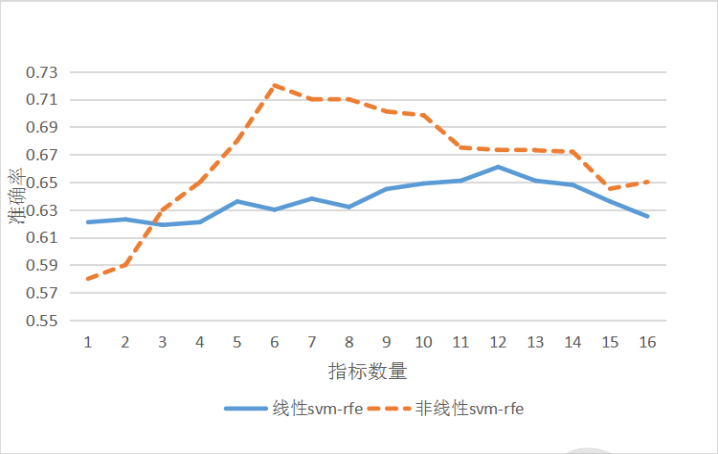


图 1: 线性和非线性比较

在图 1中可以看到对于线性SVM而言，不同的指标特征数量对准确率的影响很小，总体在0.65左右波动，表明线性SVM并不能够很好地将数据分割开。数据在特征空间很大程度是非线性可分的，可能因为眼动指标之间较为复杂，指标和指标之间的有一定的关联性，并不是所有的指标都具有非互性。

而对于非线性SVM-RFE而言，一方面可以看到其准确率随着指标的选择数有明显的变化，能够反应出不同指标对于准确率的影响程度，另一方面，在指标数通过RFE筛减到6的时候，准确率达到了最高，说明前6个指标的综合作用效果最为显著。这也验证了非线性SVM-RFE对于眼动数据指标分类的可行性。

4.2 实验二结果分析

在实验一的基础上，可以得到非线性优于线性分类器的结论，此时可以确定实验二的基本模型为非线性SVM-RFE分类器。实验二需要考虑不同被试对于实验的影响，因为在选择被试的时候并不能够具体确定其阅读水平。

该文按照125位被试切分数据集，每个被试形成一个独立的数据集，该数据集内包含若干个语篇的眼动数据。将每位被试的数据集分别用非线性SVM-RFE分类器训练。得益于SVM方法对小样本数据训练的可靠性，虽然切分后的数据量小于切分前，但是结果依然稳定。被试准确率分布情况如图 2所示：

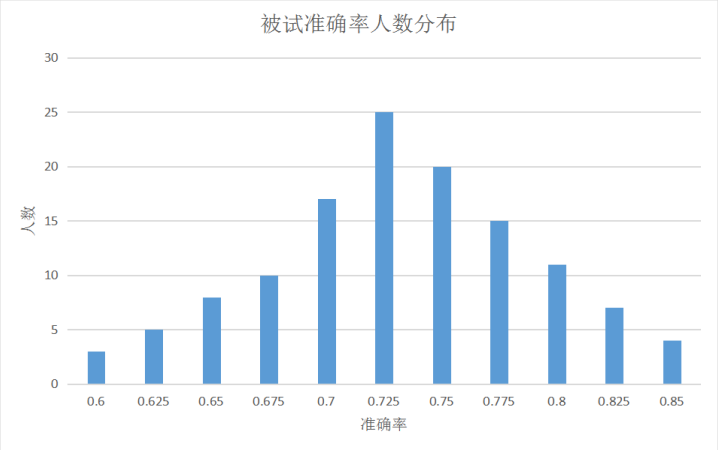


图 2: 被试准确率人数分布图

可以看到不同被试的准确率有着一定的差异性。整体而言，所有被试的准确率分布近似于正态分布，在0.725-0.75区间内的被试人数最多。通过该结果可以得到被试准确率的排序，准确率较低的被试可能会成为实验的误差。考虑到每篇语篇都有至少三位被试参与了实验，所以可以依据被试的准确率排名从高往低筛选语篇眼动数据，能够保证每篇语篇的眼动数据都是最优，最终可以得到一个整体眼动数据最优的数据集。

4.3 实验三结果分析

在实验一和二中，确定了非线性RBF核方法和基于区分被试的最优数据集。实验三就使用该最优数据集训练基于非线性RBF核的SVM—RFE模型。要注意的是在实验三中，数据集依据被试阅读的两个阶段划分为两个部分并分开训练。这样就可以做*article*阶段和*gist*阶段的对比分析。

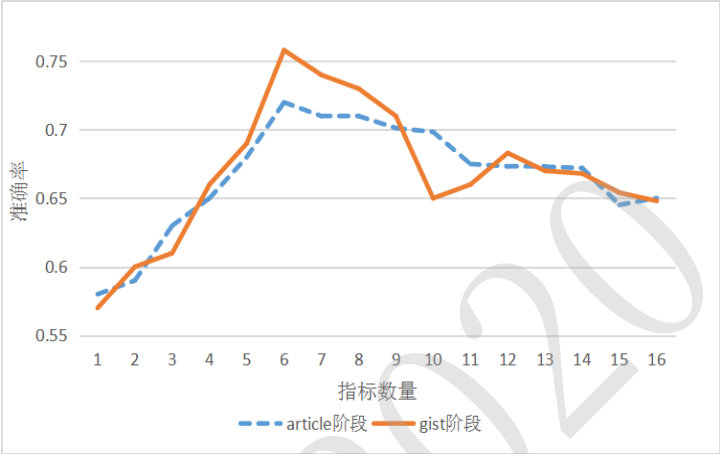


图 3: 不同阶段比较分析

图 3给出了两个不同阶段的准确率折线图。可以看到对于整体趋势而言，*gist*阶段和*article*阶段较为相似，都是在指标数为6个左右时准确率达到最佳。但是*gist*阶段的最高准确率为0.758，相应地比*article*阶段的准确率高一些。这也反映出*gist*阶段的被试数据质量要更高，即带有概括主旨任务的被试数据更具备判断关键信息的模式。

表 4给出了*article*和*gist*两个阶段的眼动特征重要性排序表，最左列为指标名称以及部分含义解释，右侧两列为两个不同阶段的指标重要性排序，数字越小表示重要性越高。表中*article*列按照重要性从高到低降序排序，*gist*列为与*article*相对应的排序。这样处理方法可以更清楚地对比两个阶段的异同。

整体而言，*article*和*gist*两个阶段眼动指标的整体排序有相似之处也有不同之处。首先，对于排名前6的最为重要的指标而言，*article*和*gist*都包含了所有注视点总注视时间、该兴趣区所有注视点个数、注视该兴趣区次数和该兴趣区阅读遍数四个指标。这四个指标反应的是被试对于一个PCS层小句的阅读频率和阅读时长。为了判断小句是否为关键句，被试对关键信息的阅读频率会更高，甚至多次多遍阅读。而对于非关键信息而言，正好相反。该结果可以从两方面来解读。一方面，总注视点个数以及次数在心理学里反映了被试对于阅读材料的认知加工负荷情况 (Henderson and Ferreira, 1990)，认知负荷较大的材料，注视点个数以及次数也越多。而含有主旨信息的内容可以被看作是句群中需要重点加工处理的对象，所以在主旨判断过程中，该四个指标的重要程度靠前是合理的。另一方面，不同阅读水平的被试阅读同一材料的注视点总个数和次数也不同，熟练阅读者的次数要明显少于不熟练阅读者的次数 (Rayner et al., 2011)，而实验二中的筛选被试环节也减少了阅读者熟练度差异所导致的误差。所以这四个指标的重要性正确地反映了主旨提取认知过程，同时被试的筛选环节也使得结果更加可靠。

其次，*article*和*gist*阶段指标有着不同的分布特性。在*article*阶段，含有总时间和次数的特征重要性靠前，第一遍阅读相关的特征其次，第二遍、第三遍阅读的相关特征重要性位于最后。而在*gist*阶段，最后一遍的阅读时间次数特征重要性靠前，含有总时间和次数的特征重要性其次，第一遍和第二遍特征重要性排在最后。这也符合人的阅读理解认知过程。在*article*阶段，被试实际上不会对关键信息重复确认，大部分句子都只会阅读一遍，如果第一遍阅读相关

指标含义	article排序	gist排序
所有注视点总注视时间	1	4
该兴趣区的所有注视点个数	2	3
注视该兴趣区次数（注视点落入后离开算注视一次）	3	6
第一遍进入兴趣区到离开的所有注视点时间之和	4	11
该兴趣区阅读遍数	5	5
第一遍阅读这个兴趣区的注视点个数	6	8
最后一次进入兴趣区的注视次数	7	1
最后一次进入兴趣区的总时间	8	2
第三遍阅读这个兴趣区的注视点个数	9	7
从第三次进入兴趣区到离开所有注视点时间之和	10	10
第二遍阅读这个兴趣区的注视点个数	11	9
从第二次进入兴趣区到离开所有注视点时间之和	12	12
第一个注视点的注视时间	13	14
第三个注视点的注视时间	14	13
第二个注视点的注视时间	15	16
最后一个注视点的注视时间	16	15

表 4: 眼动指标两阶段排序

的时间和注视点个数较多，说明被试在该小句上花费的时间较久，则可认为此句包含较重要的事件信息的可能性更大。第一遍阅读属于前期加工指标，反映了被试对于信息的预处理情况。这个结果也反映了被试在信息的预处理阶段就可以大致分辨出句群中的关键信息。在*gist*阶段，被试需要在阅读后总结主旨，此时最后一遍阅读相关文本的信息较为重要。在人的阅读行为中，涉及较长文本的语篇时，被试需要首先通读全文，然后找到包含主旨信息的小句着重阅读，提取主要信息。这些认知过程则直接反应在最后一遍阅读行为上，在关键信息句上可能会多次阅读。同时该指标也属于后期加工指标 (Clifton Jr et al., 2007)，反映了人对于信息的重加工处理，被试经过前期加工处理得到大致的主旨信息范围后，再通过重加工以确定主旨小句。所以*gist*阶段的指标重要性排序也有较好的解释。

另外，值得注意的是，回视相关的指标是篇章理解过程的重要标志，在*article*阶段，第一遍阅读重要性相比第二、三遍阅读重要性更高，在*gist*阶段，最后一遍阅读重要性最高。最后一遍实际上指代的可能是第四或者第五甚至更多遍，这反映了*gist*阶段对于兴趣区的回视次数多于*article*阶段。在心理学研究中，阅读次数的增加有利于理解准确率的增加 (Schotter et al., 2014)，所以可以认为*gist*阶段被试对于句群的理解要好于*article*阶段，这也从认知机理的角度提供了实验三中*gist*阶段的准确率要高于*article*阶段的原因。

最后，对于重要性排序较为靠后的指标，*article*和*gist*重合度较高，例如第一个，第二个，第三个和最后一个注视点的注视时间等指标都较为靠后，关于阅读兴趣区的注视点特性并不能和该句的重要性产生直接关联，在阅读过程中，上述几个指标的注视点都处于阅读的小句内部，小句内部的注视点更可能反应的是小句内部的字或词语级别细粒度的信息重要程度，例如小句内部不同字或词语对于该小句的重要性，不能够反映小句对于整个语篇的重要性。实际上，在心理学研究中，以次为单位的注视时间是衡量以字或词为兴趣区的眼动指标，而以遍为单位的注视时间则是衡量以短语或者句子为兴趣区的眼动指标 (朱滢, 2000)。对于不同细粒度而言，研究者的研究重点是不同的，像只有一个字或词的兴趣区，研究者更关注于语言特征和词汇通达，而对于包含短语或者小句的较大兴趣区而言，研究者则关注词语的逻辑分析和句法的结构判断。重要性较低的指标启发我们对于句群的主旨信息提取而言，人的认知过程是建立在PCS层小句为单位上的，小句内部的眼动数据在此细粒度上不能够发挥作用。

图 4给出了*article*和*gist*两个阶段的指标权重分布可视化情况。图中分为上下两部分，分别为16列5行的像素矩阵，5行代表着随机选择的PCS层次片段，16列代表着16个指标，矩阵中的颜色越深代表该特征越重要，即特征重要性和矩阵块颜色深浅成正比。在图中可以看到，对于*article*阶段而言，整体的颜色分布大致为前六个左右较为深，即总注视时间和次数以及第一遍相关的指标重要性较强，第二、三注视时间的相关指标颜色最浅。对于*gist*阶段而言最后一

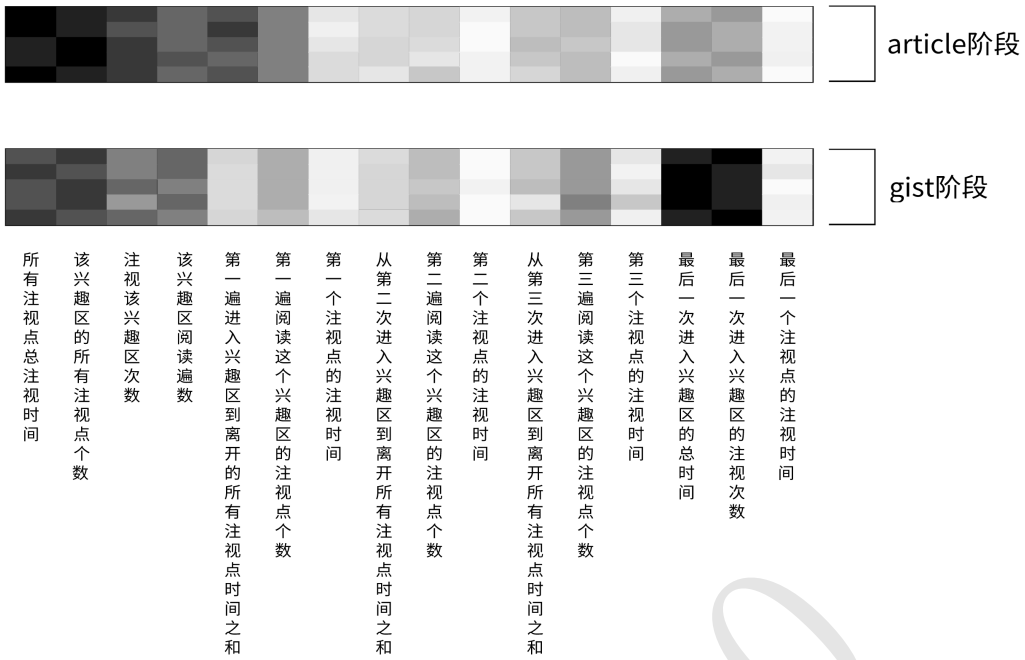


图 4: 不同阶段指标分布比较

次进入兴趣区以及总时间时间次数相关的指标颜色最深，重要性最强，同时颜色较深和颜色较浅的区域也和*article*阶段的区域有部分重合，也反映了两个阶段的相似性和相异性。该结果展示了两个阶段的指标不同显著性分布，凸显了自由阅读和主旨归纳阅读过程的眼动特征，也更加直观地感受到主旨归纳实际上更注重文本后期加工，以及两个阶段的重要指标从前期加工到后期加工的过渡。明确了主旨提取任务中的眼动模式。

5 结语

该文从眼动数据出发，结合人工主旨结构标注数据，验证了通过眼动数据建立主旨信息判断计算模型的可行性。

从线性到非线性的实验中可以看到眼动注视数据的复杂性，其更适合用非线性方法去处理和建模。在区分被试的实验中也看到了不同被试的阅读水平有一定的差异性，对于主旨信息的理解也有一定的差别，高阅读水平的被试更有可能提取出更精确的信息。在最后的自由阅读和主旨归纳两个不同的阅读状态分析中，两个阶段的共同性和差异性反应出人在提取主旨信息时候的认知过程，并且此认知过程能够在眼动数据中找到支撑。

最终经过上述过程得到了0.76左右的PCS小句分类准确率，并且指标分析也有一定的认知解释，给出了主旨提取过程的眼动模式。据此可以知道对于文本主旨概括研究而言，结合眼动数据分析是一个可行的方法。未来从词语等不同的细粒度角度或者眼跳等不同的眼动模式角度分析处理眼动数据指标也是很重要的课题，同时对眼动数据的深度挖掘，结合文本语义模型构成多模态模型也是可以研究的方向。

致谢

论文工作得到国家自然科学基金重点项目61433018资助。中科院心理所博士生李琳主持完成了相关眼动数据收集和整理，中央民族大学4名本科生协助完成了相应短讯句群的主旨结构分析标注，在此一并表示感谢。

参考文献

Hervé Abdi. 2007. Z-scores. *Encyclopedia of measurement and statistics*, 3:1055–1058.

Irene Ablinger, Walter Huber, and Ralph Radach. 2014. Eye movement analyses indicate the underlying reading strategy in the recovery of lexical readers. *Aphasiology*, 28(6):640–657.

Charles Clifton Jr, Adrian Staub, and Keith Rayner. 2007. Eye movements in reading words and sentences. In *Eye movements*, pages 341–371. Elsevier.

Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. 2002. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1-3):389–422.

John M Henderson and Fernanda Ferreira. 1990. Effects of foveal processing difficulty on the perceptual span in reading: Implications for attention and eye movement control. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 16(3):417.

Jukka Hyönä and Robert F Lorch. 2004. Effects of topic headings on text processing: Evidence from adult readers' eye fixation patterns. *Learning and instruction*, 14(2):131–152.

Jukka Hyönä, Robert F Lorch Jr, and Johanna K Kaakinen. 2002. Individual differences in reading to summarize expository text: Evidence from eye fixation patterns. *Journal of Educational Psychology*, 94(1):44.

Zhenchao Jiang, Lishuang Li, and Degen Huang. 2016. A general protein-protein interaction extraction architecture based on word representation and feature selection. *International Journal of Data Mining and Bioinformatics*, 14(3):276–291.

Johanna K Kaakinen and Jukka Hyönä. 2010. Task effects on eye movements during reading. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 36(6):1561.

Francis Quintal Lauzon. 2012. An introduction to deep learning. In *2012 11th International Conference on Information Science, Signal Processing and their Applications (ISSPA)*, pages 1438–1439. IEEE.

Joseph Lee Rodgers and W Alan Nicewander. 1988. Thirteen ways to look at the correlation coefficient. *The American Statistician*, 42(1):59–66.

Quanzhong Liu, Chihau Chen, Yang Zhang, and Zhengguo Hu. 2011. Feature selection for support vector machines with rbf kernel. *Artificial Intelligence Review*, 36(2):99–115.

Ya Lou, Yanping Liu, Johanna K Kaakinen, and Xingshan Li. 2017. Using support vector machines to identify literacy skills: Evidence from eye movements. *Behavior research methods*, 49(3):887–895.

Olvi L Mangasarian and Gang Kou. 2007. Feature selection for nonlinear kernel support vector machines. In *Seventh IEEE International Conference on Data Mining Workshops (ICDMW 2007)*, pages 231–236. IEEE.

Christoph Molnar. 2020. *Interpretable Machine Learning*. Lulu. com.

Mohamad T Musavi, Wahid Ahmed, Khue Hiang Chan, Kathleen B Faris, and Donald M Hummels. 1992. On the training of radial basis function classifiers. *Neural networks*, 5(4):595–603.

Alexander Pollatsek, Keith Rayner, and David A Balota. 1986. Inferences about eye movement control from the perceptual span in reading. *Perception & Psychophysics*, 40(2):123–130.

Keith Rayner, Jinmian Yang, Monica S Castelhano, and Simon P Liversedge. 2011. Eye movements of older and younger readers when reading disappearing text. *Psychology and aging*, 26(1):214.

Keith Rayner. 1979. Eye guidance in reading: Fixation locations within words. *Perception*, 8(1):21–30.

Erik D Reichle, Andrew E Reineberg, and Jonathan W Schooler. 2010. Eye movements during mindless reading. *Psychological science*, 21(9):1300–1310.

SR Research. 2010. Eyelink 1000 user's manual, version 1.5. 2.

Elizabeth R Schotter, Randy Tran, and Keith Rayner. 2014. Don't believe what you read (only once) comprehension is supported by regressions during reading. *Psychological science*, 25(6):1218–1226.

Johan AK Suykens and Joos Vandewalle. 1999. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300.

Jiehang Xie, Xiaoming Wang, Xinyan Wang, Guangyao Pang, and Xueyang Qin. 2019. An eye-tracking attention based model for abstractive text headline. *Cognitive Systems Research*, 58:253–264.

Yangtao Xue, Li Zhang, Bangjun Wang, Zhao Zhang, and Fanzhang Li. 2018. Nonlinear feature selection using gaussian kernel svm-rfe for fault diagnosis. *Applied Intelligence*, 48(10):3306–3331.

吴为章. 2000. 汉语句群. 商务印书馆.

周强. 2019. 微博短讯的主旨结构分析和质量检控研究. 清华大学信息技术研究院语音和语言技术中心技术报告, TH-RIIT-CSLT-TR20190809.

朱滢. 2000. 实验心理学. *M]*. 北京大学出版社, *2004* 年, 7.

李航. 2019. 统计学习方法第二版.

闫国利, 熊建萍, 臧传丽, 余莉莉, 崔磊, and 白学军. 2013. 阅读研究中的主要眼动指标评述. 心理科学进展, 21(4):589–605.

汉语竞争类多人游戏语言中疑问句的形式与功能

张文贤
北京大学对外汉语教育学院
北京市海淀区颐和园路5号, 100871
zhwenxian@pku.edu.cn

苏祺
北京大学外国语学院
北京市海淀区颐和园路5号, 100871
sukia@pku.edu.cn

摘要

本文基于自建的竞争类多人游戏对话语料库对汉语疑问句的形式与功能进行了考察。文章首先在前人研究的基础上将疑问句的类型分为五大类，然后考察不同类型的疑问句在对话中出现的位置与功能。研究显示，是非问（包括反复问）与特指问是最常见的类型，选择问使用频率最低。大部分疑问句会引起话轮转换，具有询问功能，此外，否定与指出事实也是疑问句的主要功能。特指问的否定功能与附加问指出事实的功能比较突出。

关键词： 疑问句；形式；功能；对话；语料库

The Form and Function of Interrogatives in Multi-party Chinese Competitive Game Conversation

Wenxian Zhang
School of Chinese as a Second Language
Peking University
5 Yiheyuan Road, Haidian District
Beijing, China 100871
zhwenxian@pku.edu.cn

Qi Su*
School of Foreign Languages
Peking University
5 Yiheyuan Road, Haidian District
Beijing, China 100871
sukia@pku.edu.cn

Abstract

Based on self-constructed corpus of multi-party conversations, this paper investigates the forms and functions of Chinese interrogatives. Drawing on the previous studies, it divides the interrogatives into five types: wh-question, yes/no question, tag question, alternative question and declarative question, then it examines both the positions and functions of these interrogatives in sequence. It finds that yes-no questions (including positive-negative interrogatives) and wh-questions are the most frequent while alternative questions are the least frequent; most interrogatives lead to turn-taking, and thus inquiry-prone, their major functions also are negating and fact-pointing; the negating function of wh-questions and the fact-pointing function of tag questions are quite prominent.

Keywords: Interrogatives , Forms , Functions , Conversation , Corpus

1 引言

*通讯作者

邵敬敏 (2013)注意到了疑问句形式与功能的关系，指出疑问句可分为结构类和功能类，其中功能类包括回声问、反问、设问。但这些类别实际上并不完全是功能类，比如回声问主要是针对上文的句子或者句子中的部分词语发出疑问，这仍是从形式上概括的。显然，功能类倾向于无疑，而“疑问”本身是一个模糊的概念，在“有疑而问”与“无疑而问”之间存在疑问程度不同的句子。徐杰、张林林 (1995)，徐盛桓 (1999)都强调疑问句内部存在疑问程度的差别。在各种问句中，反问句似乎没有疑问程度的问题，因为它表达否定 (吕叔湘, 1942; 张伯江, 1996; 齐沪扬、丁婵婵, 2006; 胡德明, 2010)。虽然叫反问句，实际不是问，其表达的意义是“无疑”的。但也有一些学者发现反问句既可以“无疑”，也可以“有疑”，如 (苏英霞, 2000)，(李宇凤, 2008)。关注疑问句的功能必然会涉及到其除了表达询问之外的所能够实施的行为，相关研究自然集中在反问句上。刘松江 (1993)认为使用反问句是说话人对自己感情的宣泄。郭继懋 (1997)指出反问句的语义语用条件在于间接地告诉别人他的行为不合情理。邵敬敏 (1996)、刘娅琼和陶红印 (2011)等都认为反问句有反驳等功能。此外，高华、张惟 (2009)认为“寻求核实”与“请求允可”是附加问句的两类基本话语功能。

前人虽然做出了大量的研究，但疑问程度、功能与从结构上分出的疑问句句式之间的关系错综复杂，前人对疑问句形式与功能的对应关系缺乏考察，更缺乏定量分析。在前人研究的基础上，我们提出以下问题：

1. 在实际言谈中，各种形式的疑问句使用情况如何？哪种形式最为常用？
2. 在对话中，疑问句分布在话轮首、话轮中还是话轮尾？是否一定会引起话轮转换？
3. 疑问句的功能除了询问、反问，还有哪些常用功能？
4. 汉语疑问句的功能与形式之间有没有较强的联系？

为了回答上述问题，本文自建多模态多人对话语料库，对其中的疑问句进行穷尽式的统计与分析，以得出疑问句形式、分布与功能的对应关系。本文所使用的语料来自网络电视节目《饭局的诱惑》中游戏部分的对话，该节目的嘉宾有9或者10个人，他们玩儿狼人杀游戏的整个过程被录制了下来。嘉宾在玩游戏时自发产出无准备的对话，语体的性质为竞争类多人游戏语言。与双人对话相比，这种多人对话更加复杂。这种语体属于口语，但不同于自然闲谈类口语的是，该类语体中语言的产出都与游戏的内容有关，游戏的进程需要依靠语言的推进。由于游戏规程的需要，每个嘉宾都要想方设法辨别对方的真实身份，因此疑问句出现的频率相对较高，话轮长度与转换也可能随之受到影响。为使语料的性质更加纯粹，我们只将该电视节目中的玩游戏时的对话部分逐字转写下来，不转写玩游戏之前的热身部分以及介绍游戏规则的独白部分。共转写了11期节目，约10小时，9万字。语料收集好后进行标注。语料标注由两位语言学专业的老师独立进行，标注结束后再核对，对于两位老师标注不一致的句子，当面讨论，根据上下文语境达成最终标注结果。标注的内容包括疑问句的形式、位置、功能。标注所用的具体符号随文说明。

2 疑问句的形式类型及其分布

2.1 疑问句的形式类型

吕叔湘 (1942)将疑问句分为“特指问”和“是非问”。这种以疑问域为切入点的分类对后来的研究影响较大，陆俭明 (1982)、袁毓林 (1993)、邵敬敏 (1996)与张伯江 (1997)继续发展了这一思想，特别是后三位学者对疑问句的分类层级性很强、很细致。从大类上来说，特指问与是非问的形式特征不同，首先二者疑问词不同，其次二者对疑问句的回答也不同，是非问可以用“对、不对”回答，而特指问不能。以结构特征为标准分出的特指问、是非问、选择问、反复问、附加问等虽然辨识度高，但在分类层级方面存在一些争议，比如是非问与选择问的关系就比较复杂。除了把是非问放在第一层级，还有一种观点认为，除特殊疑问句之外，其余都是选择问，而选择问有特指选择问和是非选择问之分，是非问句就属于后者 (范继淹, 1982)。谢心阳 (2018)从互动的角度分析了汉语的问答形式，认为是非疑问句包括由形态－句法－词汇手段构成的疑问句、陈述疑问句(简称陈述问)和附加疑问句三大类，该分析为我们理解问句提供了重要的参考。

我们认为，可以根据回答的情况来给疑问句分类。特指问有“谁”“什么”“为什么”“哪儿”“怎么”“怎么样”等疑问代词，回答会针对疑问点进行，应该单独一类，如例[1]。是非问与反复问可以归为同一类，是非问带疑问词“吗”，反复问从正反方面进行提问，它们都是用肯定或否定来回答，如例[2][3]。附加问虽然从形式上来说也可以用肯定或者否定来回答，但是对附加问的回答常常是对所实施行为的回答，宜把附加问单独归为一类，如例[4]。选择问要求听话人从发问者提供的选项中选择一个作答，宜单独归一类，如例[5]。通过对语料的分析，我们发现，听话人把某些不带疑问词的陈述句也作为疑问句来回答，因为这些陈述句是B-events（指的是对于下一说话人B来说是已知信息，对于上一说话人A是未知信息）[0]，陈述疑问句事实上是一种通过互动参与者之间认知不平衡性获取回应的互动行为 (谢心阳, 2016)。我们并没有把这一类归入是非问是因为这一类疑问句是从功能角度而不是从形式角度定义的，如例[6]。

[1] 马东：你肯定投我，为什么我可以再留一轮?
那威：我哪知道，因为大家不一定听我的呀。
[2] 胡可：必须得投吗?
那威：是。你可以弃权。
[3] 撒贝宁：现在有没有可能是两狼在?
侯佩岑：有！
[4] 尼格买提：你们不要让他带走我好不好?
马东：不是，我就想问一下，你为什么偏偏把我扔地上？把尼格买提还留在桌子上。
[5] 艾力：阿娇你是平民还是有身份?
阿娇：我是平民。
[6] 马东：当时你睁着眼。
大王：我当时在睁眼，因为我是女巫。

例[1]是特指问，有疑问代词“为什么”。例[2]为是非问，有疑问词“吗”。例[3] “有没有.....？”是反复问。例[2]与[3]都需要听者做出肯定或者否定回答，都归为是非问。例[4]“.....好不好？”是附加问，第二个说话人马东并没有回答“好”或者不好，没有理睬尼格买提的请求。例[5]“.....还是.....”是选择问，提供了平民和有身份这两个选项。例[6]是陈述问。马东说“当时你睁着眼”虽然没有疑问词，但睁没睁眼只有大王自己知道，她才是信息的权威知晓者，因此大王对马东的陈述做出了肯定的回答。

这样，根据形式，我们将疑问句分为特指问、是非问（包括反复问）、附加问、选择问、陈述问五大类，分别用字母Q、P、T、A、D在语料库中进行标注。在这一轮标注中，我们完全从形式出发，并没有考虑疑问句的功能。比如例[7]大王根据那威说的不会杀谁推论出会杀谁，那威对大王的推理并不赞同，“不是得罪人吗？”实为否定大王的说法，但因为有“吗”这一疑问词，因此仍标为P。再比如例[8]的第一个话轮中，尼格买提说的“你验了8号的结果是什么？”中的“什么”是真性问，而第三个话轮撒贝宁说的“真是高手，她这说的什么？”虽然也有疑问词“什么”，但并不是真正的寻求信息的疑问句，而是表达质疑，在第一轮标注中，例[8]中两个带有“什么”的句子均标为Q。

[7]那威：侯佩岑我不会杀。马东我不会杀。你我更不会杀。
大王：等于我们剩下的该死.....
那威：你非逼着我这么说，不是得罪人吗?
[8]尼格买提：你验了8号的结果是什么?
颜如晶：他是好人！我第二验的撒老师，撒老师是，是狼。.....
撒贝宁：真是高手，她这说的什么?

按照疑问句的形式统计的结果如图1所示：

[0]根据Stivers (2010)，英语中大部分陈述疑问句都包含一定程度的上升语调，他将之归在是非问句里，并认为陈述疑问句占主导地位。例如：You are married?是陈述疑问句。Are you married? 则是标记性是非问句。汉语中有些没有语调上扬或者上扬不明显的B-events陈述句实际也在表达问，听话人会回答这样的句子，我们将这类句子单独区分出来。在A、B信息交互方面，张文贤、乐耀 (2018)介绍了A、B-events理论。B-events(Known to B，but not to A)是基于B的事件信息。

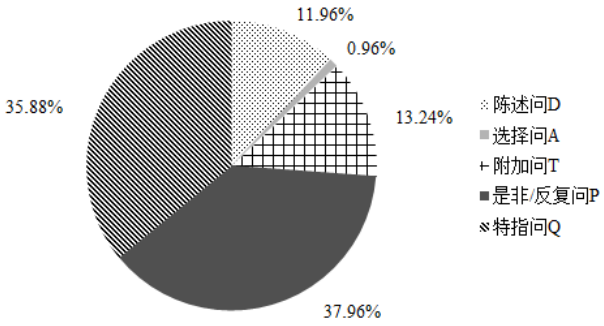


Figure 1: 疑问句的形式

从图1可以看出，是非问与反复问占的比例最大，为37.96%（238例），特指问与之比例相近，为35.88%（225例），附加问比陈述问稍高，分别为13.24%（83例）与11.96%（75例）。选择问出现的最少，只占0.96%（6例）。我们统计的结果与Stivers（2010）所调查的2-5人美式英语自然对话中疑问句的情况一致，也为是非问最高，但不同的是，我们的语料中特指问与是非句地位相当。

2.2 不同形式的疑问句在对话中的位置

根据Sacks等 (1974)，一次只有一方说话（一个人一次说的话叫话轮），在对话中发生话轮转换，话轮转换是会话组织的基础。当疑问句是在询问对方、请求对方回答或确认时，大多会发生话轮转换，如例[9]。可是疑问句不必然引起话轮转换，也可能发出疑问句后继续言谈或者疑问句出现在话轮的中间，如例[10]。

[9] 沙溢：你又要把他——一个人家说是预言家的要投出去，现在你在这里搅混水，你又要把我投出去。你到底是什么身份?

那威：我是好人，好人。好。我没有搅浑水，我是完全充满着对胡可老师的那种信任。

[10]马东：我是预言家！我查杀她（胡可），她（侯佩岑）不说话。如果那个时候她就应该知道我是狼了对吗? 如果我是狼跳的话.....

侯佩岑：没有，我有接受特训，我不能那么早跳！

例[9]沙溢先做出一番评论再提问，疑问句“你到底是什么身份？”位于话轮尾。那威接过话轮，回答了沙溢的问题。例[10]马东说的“如果那个时候她就应该知道我是狼了对吗？”处于话轮中间的位置，“.....对吗？”并不是寻求对方确认，而是引出推论的前提。

我们将语料库中所有疑问句进行位置标注，统计不同形式的疑问句在话轮首、中、尾的分布情况。S表示疑问句单独占一个话轮，B表示在话轮起始位置但不单独成为话轮，M表示出现在话轮中间位置，E表示出现在话轮结束位置。所以，如果是非问或反复问单独占一个话轮，则标记为SP，位于话轮尾为EP，位于话轮首但不单独占一个话轮标为BP，位于话轮中标为MP。其他类型疑问句的情况以此类推。

Table 1: 疑问句在话轮中的位置

疑问句的类型	单独占一个话轮（S）	位于话轮尾（E）	位于话轮首（B）	位于话轮中（M）	小计
是非问与反复问（P）	146（61.34%）	34（14.29%）	45（18.91%）	13（5.46%）	238
特指问（Q）	145（64.44%）	25（11.11%）	30（13.33%）	25（11.11%）	225
附加问（T）	47（56.63%）	4（4.82%）	13（15.66%）	19（22.89%）	83
陈述问（D）	58（77.33%）	6（8%）	10（13.33%）	1（1.33%）	75
选择问（A）	5（83.33%）	1（16.67%）	0	0	6
总计	401	70	98	58	627

表1显示，所有类型的疑问句都是单独占一个话轮的情况最多。但疑问句并不完全是用来问的，也就是说，说话人发出疑问句并不必然需要对方回答或者不等待对方回答。说话人可能发出一个疑问句之后继续言谈，或者在言谈过程中发出了一个疑问句而这个疑问句并没有使说话人交出话语权。为了更好地观察疑问句是否引起话轮转换，我们将表2中的数据进行归类，分为引起话轮转换与未引起话轮转换两种情况，单独占一个话轮的疑问句与位于话轮尾的疑问句均引起话轮转换的，归为一类，位于话轮首但不单独占一个话轮与位于话轮中的疑问句均未引起话轮转换的，归为一类。统计结果见表2。

Table 2: 疑问句在话轮中的位置

疑问句的类型	引起话轮转换	未引起话轮转换
是非问与反复问（P）	180（75.63%）	58（24.37%）
特指问（Q）	170（75.56%）	55（24.44%）
附加问（T）	51（61.45%）	32（38.55%）
陈述问（D）	64（85.33%）	11（14.67%）
选择问（A）	6（100%）	0

从表2可以看出，选择问100%会引起话轮转换，听说人听到选择问后会接过话轮，做出回答。其次是陈述问，85.33%的陈述问会被接过话轮，而是非问与特指问相当，都是约为76%，最特别的是附加问，只有61.45%引起了话轮转换。这说明大部分疑问句被识解为有疑而问，听话人认为需要回答，所以在疑问句结束后就接过话轮。但是有些疑问句特别是附加问并不是为了问而发出的，这需要我们对疑问句的功能即所实施的行为做进一步的分析。

3 疑问句的功能

要想更好地解释疑问句的功能，离不开疑问句的言谈环境。如果把疑问句放到口语对话中，从交际互动的角度去分析，就可以更好地认识问句的功能 (张文贤、乐耀, 2018)。在实际言谈交际中，从形式上看是疑问句的句子并不一定是表达“疑”或者“问”，不管是特指问、是非问，还是附加句，除了疑问功能外还具有其他功能，比如实施言语行为或者组织话题等。比如例[11]- [13]：

[11]蔡康永：我本来想休息来着，结果你把我救回来干什么？

大王：所以，我们现在可以来想一想，谁会第一天晚上就把康永哥，就把康永哥给杀了，而且很明显就是骗解药的。

[12]大王：我第一把我就说了，我是一个很好的身份，记得吗？所以我只敢肯定，他肯定是狼人。

陈怡馨：啊.....我真的不知道选谁.....但是，因为我不，就是佩岑姐姐，我是，我只是因为凭，就是她那个反应，但是，就刚刚她说的话也的确没什么问题.....

[13]张大大：我认为预言家现在没有危险，你即便是预言家，你都可以跳出来，但你好像不是预言家的感觉。我相信场上已经走了一匹狼了，所以预言家可以跳了好不好？我来保护你，只要你跳我就相信你，我带走你说的那个人，一瓶毒药，说完了。

例[11]蔡康永“.....结果你把我救回来干什么？”一句中有特殊疑问词“什么”，但却不是疑问句，而是反问句，否定大王的做法，埋怨大王不该把他救回来。从大王的回应中我们也可以看出，大王并没有把蔡康永的话作为疑问句来回答，而是用“所以”拉回之前正在讨论的话题，即谁是狼人。例[12]大王说“.....，记得吗？”有疑问词“吗”，从形式上看是一般疑问句，但大王实际上并不是问大家是否记得，而是指出事实，使之前的话语重新回到当前的言谈中，她说完“记得吗？”之后不等大家回答就继续说自己的判断。从听话人来看，也没有人回答记得或者不记得，陈怡馨接过话轮，继续分析哪个是狼人。例[13]附加疑问句“.....好不好？”用在话轮中间，宣布接下来要做的事情，“走了一匹狼之后预言家可以跳”是大家都了解的游戏规则，无需其他会话参与者回应。

正如前文所说，前人对汉语疑问句的功能缺乏系统性的研究，相关研究多集中在反问句或者某些具体格式上，并没有我们可以直接拿来运用的分类框架。在对功能进行分类时，我们既

考虑问句也考虑答句，对疑问句的功能分了三大类，它们是：

1.询问。指的是说话人有疑问，不知道某一信息或对某一信息不确定，因此询问对方，希望得到回答。在语料库中用数字“1”标注。具体包括：

a.说话人就某一信息提问，对方通常会做出回答。

[14]张大大：你想跳预言家是吗?
颜如晶：啊，是。我想跳预言家。

例[14]张大大对颜如晶是否想跳预言家有疑问，他提出问题后颜如晶给予了正面回答。

b.说话人对所说信息确定或基本确定，但仍然需要对方确认信息或回应。对方通常正面回答。

[15]金靖：对，好。我就过了，我的状态很阳光，一看就是一个明显的好人，是不是?
肖骁：哈哈哈。
张歆艺：哈哈哈。
肖骁：挺你。

例[15]金婧说自己“一看就是一个明显的好人”是在自夸，是要表明自己的好身份。在她自己心里，该论断是没有什么疑问的，但是她要征求的是大家的意见，对于大家是不是也这样认为并不确定，期待大家的回答。肖骁的回应“挺你”给出了肯定回答。

c.提议、建议对方或者在场的所有人做某事，征求意见。通常会有人回应。

[16]马东：我有一个提议，我觉得这个时候我们可以转着圈的数数，然后谁接的迟疑一点，谁有可能就正在有事干，好不好?
蔡康永：你数吧。

例[16]马东明确说“我有一个提议，.....好不好？”，这个提议是要求大家跟他一起做事情，因此要取得大家的同意，等待大家的回答，蔡康永作为法官，允许了这种行为。尽管“.....好不好？”在这里是一个行为，但因为是需要得到大家同意的提议，我们也把它归入询问。

d.怀疑对方说话或做事的真实性或者揣测对方存在不良用意。听话人可能正面回答也可能拒绝回答。

[17]肖骁：其实我现在大胆猜一下，我觉得会不会康永哥有可能是预言家。
马东：盖乐世手机请拿走我的瞳孔。
大王：可是他们有必要玩得这么深吗?
肖骁：对呀，所以就是因为我就觉得，其实他说完之后我就已经跟着他走了，然后后来康永哥出来一脚我的妈呀，我说他干什么呀这些人。
大王：一直质疑。

为了掩盖自己的身份，游戏里的人都会说谎。例[17]在大王发出“可是他们有必要玩的这么深吗？”这个疑问句之前，蔡康永、肖骁等人均根据队友的表现对谁是狼人进行了分析。大王对他们的分析不是百分百认可，肖骁回应的“对呀”表示对大王的质疑也表示赞同。例[17]的最后一个话轮“一直质疑”也再次明确了大王的怀疑态度。

e.自问自答。

[18]颜如晶：.....但是这一局你确实有一点错。错在哪呢？第一，为什么我第一局就保你，因为我第一局就验你，所以我保你，我暗暗地保你，我保一圈情况之下也保你进去。

例[18]颜如晶提出一个问题“错在哪儿呢？”，这个问题不是问大家的，而是为了引出自己的分析。自问自答询问的是自己，我们也将之归入询问类。

我们将例[14]- [18]这样的疑问句归入第一类——询问。但实际上这一类所涵盖的内容较多，疑问程度也有一定差别，这种差别源于有些疑问句的功能并不单一。因为我们是从答看问，采取下一话轮验证的方法（next turn proof），如果下一话轮回答了，则该疑问句有询问的功能。上面的分类中，a、b两类对疑问句的回应是对询问信息的回应，而c是对询问行为的回应、d是对询问立场的回应，e是无疑而问，自问自答。

2.否定、反驳。说话人并没有疑问，所要表达的意思与字面意思相反。若是肯定句，就是表达否定的意思，若是否定句，就是表达肯定的意思，并且带有强烈的感情色彩。如果句子的内容是前文已经提到的相关事实，说话人用问句是为了反驳对方的观点。这类疑问句也就是反问句，在语料库中用数字“2”标注。

[19]陈怡馨：所以大王你觉得可能是谁?
大王：你呀！
（众笑）
大王：陈怡馨！
陈怡馨：哪里？大王，我们私下那么恩爱，你怎么能怀疑我呢?
[20]王博文：我只能说，真的，你演得太好了，我真的佩服你，你真的是高玩。
伊能静：我演什么呀?

例[19]的会话序列中，第一个话轮“所以大王你觉得可能是谁？”是特指问，属于第一类（询问），而第四个话轮陈怡馨的“哪里？大王，我们私下那么恩爱，你怎么能怀疑我呢？”是反问句，属于第二类（否定、反驳），表达的意思是你不应该怀疑我。例[20]伊能静不赞同王博文对自己的评价，“我演什么呀？”意思是我没有演，反驳“你演得太好了”，并表达强烈的情感。前文中的例[11]也属于这一类。由于前人对反问句的研究成果较为丰富，对其否定、反驳功能认识也较为一致，我们对这一功能不再赘述。

3.指出事实。使用疑问句是为了把之前的情景拉回到当前对话或者表达评价，有话题组织的功能。说话人没有疑问，只是重复前文出现的事实。说话人对事实持肯定态度，不等对方回答就继续言谈。在这种用法的问句中，“是吗？”“是不是？”“对吗？”“对不对”“好不好”“记得吗？”“你知道吗？”是常见的标记。在语料库中用数字“3”标注。但我们并不能说有这些附加标记的都是用于指出事实，如上文中的例[15][16]都归为了询问，而前文中的[12][13]与下文的[21][22]则是指出事实。

[21]马东：说我的人，说我是狼的人都是狼。
范恬恬：但马老师这局也没有踩人，也没有保人，是吗？他是狼的话，他超爱踩自己狼友的，嗯。
[22]蔡康永：好，袁弘。
袁弘：然后我说是不是？我真的不是狼人，如果我是狼人的话，我不会傻到去杀2号，因为我一直在质疑2号，对不对？而且，我相信不止我一个人，在场不止我一个人质疑2号。

例[21]“马老师这局也没有踩人，也没有保人”是大家都知道的事实，“是吗？”是附加问，但不是询问，也没有疑问，只是提请大家注意这一事实。例[22]法官蔡康永已经点名请袁弘说了，袁弘接过话轮还说“然后我说是不是？”，实际上是宣告自己要开始说话了。同一话轮中另外一个问句“因为我一直在质疑2号，对不对？”也是请大家注意自己之前的言行。疑问句三个功能的具体使用情况如表3所示：

表3显示，选择问100%是用来询问的，陈述问的询问功能达到百分之八十以上，是非问与

Table 3: 疑问句的类型与功能

疑问句的类型	询问	非询问功能		小计
		否定	指出事实	
是非问与反复问（P）	178（74.79%）	37（15.55%）	23（9.7%）	238（100%）
特指问（Q）	151（67.11%）	69（30.67%）	5（2.22%）	225（100%）
附加问（T）	35（42.17%）	4（4.82%）	44（53.01%）	83（100%）
陈述问（D）	61（81.33%）	13（17.33%）	1（1.33%）	75
选择问（A）	6（100%）	0	0	6（100%）
总数	431	123	73	627

反复问、特指问的询问功能也均过半。比较特别的是附加问，其指出事实的功能高达53.01%，特指问在所有疑问句类型中用于否定的可能性最大，否定功能占到特指问的30.67%。选择问不用于否定，附加问极少用于否定。表3的统计结果说明，疑问句的形式与功能之间有一定关系。

4 结语

本文在语料库中考察了疑问句的形式和功能，从整体上观察疑问句的形式、分布与功能的对应关系。研究发现，从总体上来说，疑问句使用频率最高的为是非问（包括反复问）与特指问。大部分疑问句会引起话轮转换。选择问只具有询问功能，陈述问、是非问、特指问的主要功能是表示询问，而附加问的非询问功能多于询问功能，主要用来指出事实，特指问用于否定功能的比例远远高于其他类型。需要说明的是，本文的研究结论是基于竞争类游戏语言这种具体的语体得出来的，结论是否适用于其他语体，还有待验证。

在各种语言中，疑问句都不只是用来询问信息的，还具有多种功能。根据Enfield等 (2010)对语料库的研究，英语疑问句的功能只有35%是严肃地请求信息，其主要用法还有修正之前的话语与要求确认。Levinson (2012)也指出，疑问句在功能竞技场上是干苦力的，它可以用来介绍（如，How do you do? ）、修正（如，He said what? ）、建议（Why don”t we get a coffee? ）、要求（Would you mind taking this? ）、陈述（Well，what damn fool would trust a bank with their money? ）、训斥（如，Who do you think you are? ）。毫无疑问，汉语的疑问句功能也比较丰富。接下来，我们将对多人对话中疑问句的非疑问功能进行更深入的挖掘。

参考文献

Enfield, N., Stivers, T. and Levinson, S. C. 2010. Question–response sequences in conversation across ten languages: An introduction. *Journal of Pragmatics*, 42(10):2615–2619.

Levinson. 2010. Interrogative intimations: on a possible social economics of interrogatives. *In Jan P. De Ruiter(ed.), Questions*. Cambridge: Cambridge University Press.

Harvey Sacks, Emanuel A. Schegloff, and Gail Jefferson. 1974. A simplest systematics for the organization of turn-taking for conversation. Language, 50(4): 696–735.

Tanya Stivers. 2010. An overview of the question–response system in American English conversation. *Journal of Pragmatics*, 42: 2772—2781.

范继淹. 1982. 是非问句的句法形式. 中国语文, 6: 426–434.

高华、张惟. 2009. 汉语附加问句的互动功能研究. 语言教学与研究, 5: 45–52.

郭继懋. 1997. 反问句的语义语用特点. 中国语文, 2：111–121.

胡德明. 2010. 从反问句生成机制看反问句否定语义的来源. 语言研究, 3: 71–75.

李宇凤. 2008. 汉语语用偏向问研究. 中国社会科学研究院博士学位论文.

刘松江. 1993. 反问句的交际作用. 语言教学与研究, 3: 46–49.

刘娅琼、陶红印. 2011. 汉语谈话中否定反问句的事理立场功能及类型. 中国语文, 2: 110–120.

陆俭明. 1982. 由“非疑问形式+呢”构成的疑问句. 中国语文, 6: 435–438.

吕叔湘. 1942. 中国文法要略. 北京：商务印书馆.

齐沪扬、丁婵婵. 2006. 反诘类语气副词的否定功能分析. 汉语学习, 5：3–13.

邵敬敏. 1996. 现代汉语疑问句研究. 上海：华东师范大学出版社.

邵敬敏. 2013. 疑问句的结构类型与反问句的转化关系研究汉语学习, 2: 3–10.

苏英霞. 2000. “难道”句都是反问句吗？. 语文研究, 1: 56–60.

谢心阳. 2016. 问与答：形式和功能的不对称. 中国社会科学院博士学位论文.

谢心阳. 2018. 汉语自然口语是非疑问句和特殊疑问句的无标记回应. 世界汉语教学, 3：372–386.

徐杰、张林林. 1985. 疑问程度和疑问句式. 江西师范大学学报（哲学社会科学版）, 2: 71–79.

徐盛桓. 1999. 疑问句探询功能的迁移. 中国语文, 1: 3–11.

袁毓林. 1993. 正反问句及相关的类型学参项. 中国语文, 2: 103–111.

张伯江. 1996. 否定的强化. 汉语学习, 1：15–18.

张伯江. 1997. 问句功能琐议. 中国语文, 2：104–110.

张文贤、乐耀. 2018. 汉语反问句在会话交际中的信息调节功能分析. 语言科学, 2：147–159.

融合目标端句法的AMR-to-Text生成

朱杰，李军辉
苏州大学, 计算机科学与技术学院/ 江苏省苏州市
zhujie951121@gmail.com, lijunhui@suda.edu.cn

摘要

抽象语义表示到文本（AMR-to-Text）生成的任务是给定AMR图，生成相同语义表示的文本。可以把此任务当作一个从源端AMR图到目标端句子的机器翻译任务。目前存在的一些方法都在探索如何更好的对图结构进行建模。然而，它们都存在一个未限定的问题，因为在生成阶段许多句法的决策并不受语义图的约束，从而忽略了句子内部潜藏的句法信息。为了明确考虑这一不足，该文提出一种直接而有效的方法，显式的在AMR-to-Text生成的任务中融入句法信息，并在Transformer和目前该任务最优性能的模型上进行了实验。实验结果表明，在现存的两份标准英文数据集LDC2018E86和LDC2017T10上，都取得了显著的提升，达到了新的最高性能。

关键词： AMR-to-Text生成 ； 句法决策 ； 语义约束 ； 融入句法信息

AMR-to-Text Generation with Target Syntax

Jie Zhu, Junhui Li
School of Computer Science and Technology, Soochow University / Suzhou, Jiangsu
zhujie951121@gmail.com, lijunhui@suda.edu.cn

Abstract

The task of AMR-to-text generation is to generate text with the same semantic representation given an AMR graph. This task can be viewed as a translation task from the source AMR graph to the target sentence. Some existing methods are currently exploring how to better model the graph structure. However, they all have an unrestricted problem, because many syntactic decisions in the generation phase are not constrained by the semantic graph, thus ignoring the syntactic information hidden within the sentence. In order to clearly consider this shortcoming, this paper proposes a direct and effective method, which shows the integration of syntactic information in the task generated by AMR-to-Text, and has conducted experiments on Transformer and the current model of the optimal performance of the task. The experimental results show that on the two existing standard English data sets LDC2018E86 and LDC2017T10, both have achieved significant improvements and reached new state-of-the-art.

Keywords: AMR-to-text generation , Syntactic decision , Semantic constraints , Incorporate syntactic information

1 引言

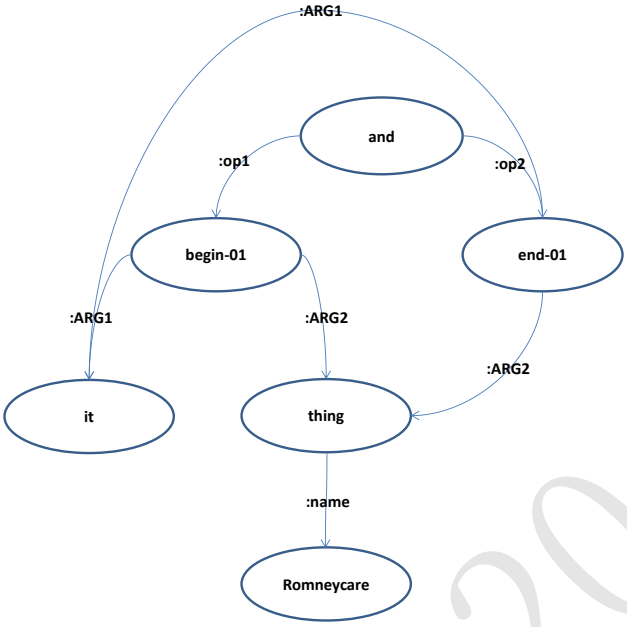


图 1. “It begins and ends with Romneycare .” 抽象成AMR图的一个例子

抽象语义表示（Abstract Meaning Representation，简称AMR）(Banarescu et al., 2013)是一种新型的语义表示方法，它是从文本中抽象出来捕捉核心的“谁对谁做了什么”的语义结构，形式上是一种单根有向无环图的结构。图 1给出了一个AMR图示例，它是由句子“It begins and ends with Romneycare .”抽象而成的。文本中的实词被抽象成AMR图中的概念节点（concept），如图中“begin-01”和“thing”等节点称作为概念。概念之间的相互关系则被抽象为边（edge），表示两个概念之间存在的语义关系，比如“:ARG0”和“:op1”等。AMR图在语义表示中已经得到了广泛的应用，并且在机器翻译(Tamchyna et al., 2015)，问答系统(Mitra and Baral, 2016)，事件抽取(Li et al., 2015)等自然语言处理相关任务也得到了实践。与此同时，AMR-to-Text生成在近年来也受到了越来越多的关注。

AMR-to-Text生成是在给定AMR图的条件下，自动生成相同语义的文本。该任务现存的一些方法(Flanigan et al., 2016; Konstas et al., 2017; Song et al., 2016; Song et al., 2018; Beck et al., 2018; Damonte and Cohen, 2019; Zhu et al., 2019)都着重在考虑如何对图关系进行建模，从而忽略了生成时存在的句法约束。

最初的工作是采用基于统计的方法(Pourdamghani et al., 2016; Song et al., 2017; Flanigan et al., 2016)，随后Konstas(2017)将该任务引入到了序列到序列（sequence-to-sequence，简称S2S）模型上，使用双向长短时记忆网络（Bi-LSTM）进行编码。但是S2S模型需要将AMR图进行序列化去适应模型的输入，这样会损失大量的图结构信息。因此，为了更好的对图关系进行建模，Beck等(2018),Song等(2018),Damonte等(2019),Guo等(2019),Zhu等(2019)提出了图到序列（graph-to-seq，简称G2S）的框架，使用图模型来对AMR图进行建模。然而，他们的工作都将句子表示为单词序列，并没有考虑到句子中潜在的句法信息。最近的一些研究也表明，即使百万级的平行语料，模型仍然无法从中捕获深层的句法信息(Li et al., 2017)。

针对上述存在的问题，本文提出一种显示的方法来融入句法信息，从而给定生成时一些句法约束,并且不需要对模型本身进行任何修改。为了更好的证明本文方法的有效性，本文选取了S2S中最优的Transformer模型和G2S中现存最优的模型(Zhu et al., 2019)进行了实验。最终，在两份标准的英文数据集LDC2015E86和LDC2017T10上都取得了显著的提升。

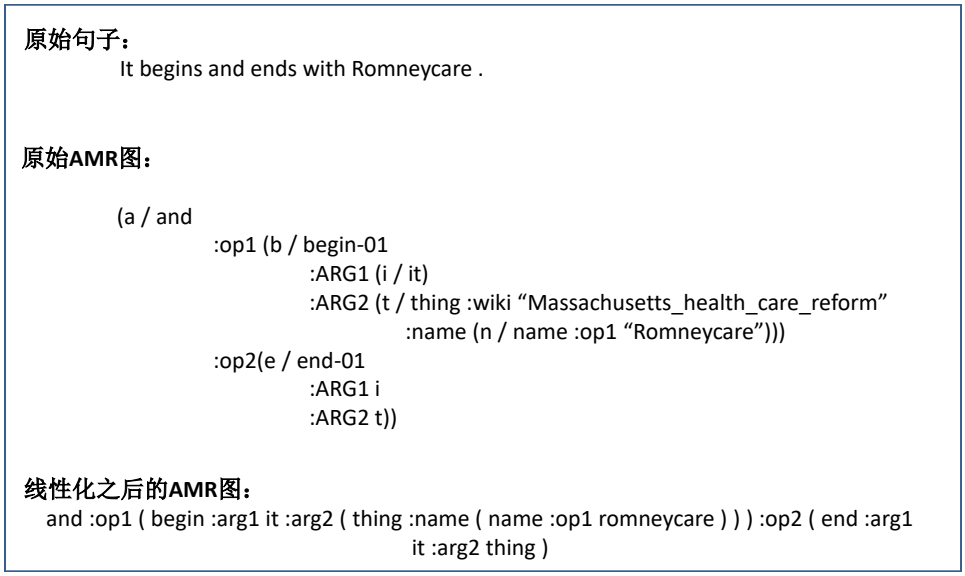


图 2. 一个AMR图线性化示例

2 相关工作

目前AMR-to-Text生成的任务大致可以分为两类：基于统计的方法和基于神经网络的方法。而基于神经网络的方法，现在又可以分为seq2seq和graph2seq两类。

2.1 基于统计的方法

早期神经网络未普及时候，在AMR-to-Text生成上的工作大都使用基于统计的方法。Flanigan等(2016)将AMR图转换为合适的生成树，并应用树-串（tree-to-string）转换器生成文本。Song等(2016)将一个AMR图拆分成了许多小的片段，并生成所有片段的翻译，最终通过采用非对称广义旅行商问题解法来对片段确定其顺序。Song等(2017)使用同步节点替换语法来对AMR图进行解析，并生成相应的句子。Pourdamghani等(2016)采用基于短语的机器翻译模型来对线性化AMR图进行建模。

2.2 基于神经网络的方法

随着神经网络的兴起，最近的研究都是使用神经网络来生成。在Sutskever等(2014)证明了深度神经网络的优越性之后，Konstas等(2017)提出使用序列到序列（S2S）模型来生成文本，利用双向LSTM来对线性化的AMR图进行编码。为了限制生成的文本具有更合理的句法，Cao等(2019)将AMR-to-Text生成的任务拆分成两个步骤，先使用句法模型去预测最优的目标端句法结构，再利用预测的句法信息去辅助生成模型更好的生成句子。但是也相应的损失了深度神经网络端到端的特性，并且和本文方法相比更加复杂，增加了网络的复杂度和参数。

随后，为了解决seq2seq模型将AMR图线性化之后信息损失的问题，大家的研究热点都着重在研究图神经网络上。图到序列（Graph-to-Sequence）模型常优于序列到序列（S2S）模型，包括图状态LSTM(2018),GGNN(2018)等。图状态LSTM通过每步的迭代交换相邻节点的信息来更新节点。同时也对每个节点增加一个向量单元保存历史信息。GGNN是一个基于门控的图神经网络，将AMR图结构完整的融入模型中，并且将边信息也转化为节点，解决了参数爆炸问题的同时，也给了解码器更丰富的信息。着重于解决AMR图中重入节点的问题，Damonte等(2019)提出了一种堆栈式的编码器，由图卷积神经网络和双向LSTM堆栈而成。Guo等(2019)提出了一种深度连接图卷积网络（GCN）更好的获取局部与非局部信息。Zhu等(2019) 在Transformer的基础上，受到(Shaw et al., 2018)对相对位置建模的启发，提出了一种Structure-Aware Self-Attention的编码方法可以对图结构中任意两两节点进行完整的建模（不论节点之间是否直接相连），在该任务上取得了最高的性能。

3 方法

本文采用了两种方法作为基准模型（Baseline）。
1. Transformer，最先进的seq2seq模型，最初使用于神经机器翻译和句法分析任务(Vaswani et al., 2017)。
2. Zhu等(2019)提出的Structure-Aware Self-Attention模型，目前在AMR-to-Text生成的任务上取得了最高的性能。

3.1 基准模型1（Baseline1）

3.1.1 Trasnformer

Baseline1是使用的Transformer模型，它采用了编码器-解码器（Encoder-Decoder）的架构，由许多编码器和解码器堆栈组成。每一个编码器都存在两个子层：自注意力机制层（self-attention）后面紧接着前馈神经网络层（position-wise feed forward）。自注意力层使用了多个注意力头（attention head），将每个注意力头的结果进行连接和转换之后，形成自注意机制层的输出。每个注意力头使用点乘注意力机制（scaled dot-product）来计算输入一个序列$x = (x_1, \cdots, x_n)$，得到一个同样长度的新的序列$z = (z_1, \cdots, z_n)$:

$$z = Attention\,(x) \tag{1}$$

其中$x_i \in R^{d_x}$，$z \in R^{n \times d_z}$。每一个输出元素z_i是输入元素的线性变换的加权和:

$$z_i = \sum_{j=1}^{n} \alpha_{ij} \left(x_j W^V\right) \tag{2}$$

其中$W^V \in R^{d_x \times d_z}$是一个可学习的参数矩阵. 公式2中的向量$\alpha_i = (\alpha_{i1}, \cdots, \alpha_{in})$是通过自注意力机制模型得到的，该机制捕获了$x_i$和其它元素之间的对应关系。具体来说，每个元素$x_j$的自注意力权重$\alpha_{ij}$是通过一个softmax函数计算得到:

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{n} exp(e_{ik})} \tag{3}$$

其中

$$e_{ij} = \frac{\left(x_i W^Q\right)\left(x_j W^K\right)^T}{\sqrt{d_z}} \tag{4}$$

是一个对齐函数，它用来度量输入元素x_i和x_j的匹配程度. $W^Q, W^K \in R^{d_x \times d_z}$是可学习的参数矩阵。

3.1.2 线性化预处理

因为Transformer是seq2seq模型，输入只支持序列化的输入，所以需要对AMR图进行线性化的预处理。本文采用Konstas等(2017)提出的深度优先遍历的线性化方法来对AMR图进行预处理，从而得到简化版的AMR图。在线性化之前，首先移除了图中的变量、wiki链接和语义标签。图2展示了一个AMR图线性化示例。

3.2 基准模型2（Baseline2）

3.2.1 Structure-Aware Self-Attention

Zhu等(2019)扩展了传统的自注意力机制框架,提出了一种新颖的结构化的注意力机制，在对齐函数中显式地对元素对(x_i, x_j)之间的关系进行编码，用公式5替换公式4。

$$e_{ij} = \frac{\left(x_i W^Q\right)\left(x_j W^K + r_{ij} W^R\right)^T}{\sqrt{d_z}} \tag{5}$$

x_i	x_j	结构标签序列
begin-01	and	:ARG1↑
begin-01	Romneycare	:ARG2↓ :name↓
begin-01	begin-01	None

表 1. 图1中一些概念对之间的结构路径示例。

其中$W^R \in R^{d_z \times d_z}$ 是一个参数矩阵. 然后, 再相应地更新公式2，将结构信息传播到子层的输出。

$$z_i = \sum_{j=1}^{n} \alpha_{ij} \left(x_j W^V + r_{ij} W^F \right) \tag{6}$$

其中，$W^F \in R^{d_z \times d_z}$ 是一个参数矩阵。$r_{ij} \in R^{d_z}$代表了元素对（x_i, x_j）之间的关系，它是通过3.2.2学习到的一个向量表示。

3.2.2 学习图概念（concept）对之间的向量表示r

上述的Structure-Aware Self-Attention机制可以用来获取到图中任意两两概念对（concept pairs）之间的图结构关系。定义使用沿着概念x_i到x_j之间边标签（edge label）组成的一条路径当作概念对之间的图结构关系[0]。同时，为了区分方向，也给每条边标签相应的增加了方向符号。Table 1展示了图 1中的几个概念对之间的结构标签序列。

现在已经给定了一个结构标签路径$s = s_1, \cdots, s_k$，然后获取到它的向量表示$l = l_1, \cdots, l_k$，最后本文使用基于卷积神经网络(Kalchbrenner et al., 2014)（CNN-based）[1] 的方法来获得公式 5和公式 6中的向量表示 r_{ij}。

CNN-based

使用CNN来卷积标签序列l获得一个向量r:

$$\begin{aligned} conv = Conv1D(&kernel\_size = (m),\\ &strides = 1,\\ &filters = d_z,\\ &input\_shape = d_z\\ &activation =' relu') \end{aligned} \tag{7}$$

$$r = conv\,(l) \tag{8}$$

实验中 m 的大小常设置为4。

3.3 数据稀疏性

在训练AMR-to-Text模型的时候，因为语料数量的限制，常常会受到数据稀疏性的影响。为了解决这个问题，前人的工作有采用匿名化的方法来删除命名实体和罕见词(Konstas et al., 2017)，或者使用复制机制(Gulcehre et al., 2016)来学习，使模型可以学会从源端输入复制未登录词到目标端。在本文中，我们提出使用字节对编码（BPE）(Sennrich et al., 2016)将未登录词拆分成更细粒度，更高频的单词。再根据该任务的特性考虑，共享了源端和目标端的词表。Zhu等(2019)也在实验中证明了该方法的有效性。

3.4 融合句法信息

前人的工作都是使用平行语料来进行训练，输入源端AMR图去生成对应的句子。他们大都是将句子视为单词序列，但是却忽略了句子本身的一些外部知识，没有考虑到句子中潜藏的句

[0]当同时存在多条路径组合时，默认选择最短的那一条。

[1](Zhu et al., 2019)使用了多种方法来学习图结构表示方法，本文选择了CNN-based这一方法作为基线模型。

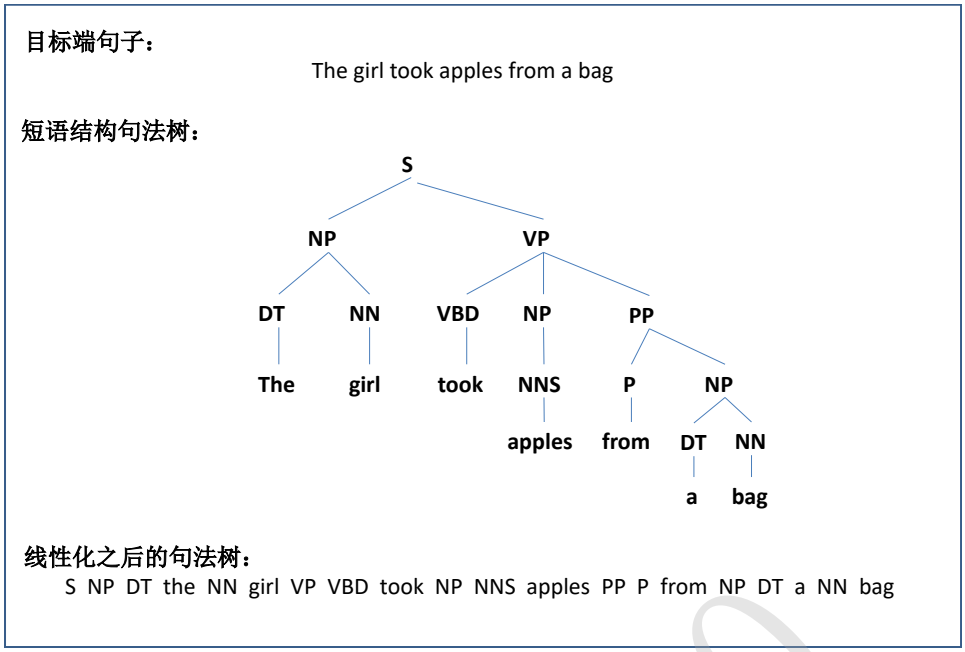


图 3. 一个目标端句子解析的短语结构句法树及其线性化示例

法信息。为了使模型能够学习到目标端句子的句法信息和内部结构，本文提出一种显式的方法来融入目标端的句法信息。

融合目标端句法信息的基本思想是将目标端句子经过解析得到句法树，之后再通过深度优先遍历得到最终的句法标签序列。句法的标注形式大致有两种，短语结构句法树和依存句法树。本文在训练时选择使用线性化的短语结构句法树(Vinyals et al., 2015)来替换目标端的句子。如图3所示，给出了一个目标端句子解析的短语结构句法树和其线性化结果。之所以选择短语结构句法树，是因为与依存树相比，它具有良好的线性化顺序的优点。此外，短语结构句法树也更容易实现，因为它们有效的对应句子中单词的顺序。在解码阶段，只需要将句法标签去除之后，就是最终预测生成的句子。

不幸的是，AMR标注数据并没有发放句法标注数据。因此，本文使用斯坦福解析器（Stanford Parser）(Manning et al., 2014)解析训练集和验证集语料，从而获得对应结构语法树的银语料（Silver-Standard）。

4 实验

4.1 数据集

为了评估方法的有效性，本文使用LDC发行的现存的两份标准英文语料集进行实验，分别是 LDC2015E86 和 LDC2017T10。两份语料集分别包含了 16,833 和 36,521 条训练数据，并且共享了 1,368 条验证集和 1,371 条测试集。训练集和验证集使用斯坦福解析器[2]获取到目标端句子所对应的 Penn treebank-style 风格的结构句法树。

4.2 实验设置

本文分别通过使用 10K 和 20K 的操作数来对 LDC2015E86 和 LDC2017T10 两份语料进行BPE操作。从BPE处理之后的训练集中根据词频建立词汇表，参考Ge等(2019)的工作，共享了源端和目标端的词汇表。为了公平的对比，模型中的词向量使用随机初始化的方式。

本文使用*OpenNMT*(Klein et al., 2017)的框架作为Transformer的基准模型[3]。在超参数的设置上，模型的编码器和解码器为 6 层。在优化器方面，本文使用$beta1$=0.1(Kingma and Ba, 2015)的Adam优化算法。自注意头的数量设置为 8。此外，模型中向量和隐藏状态的维度位置为 512，批处理大小（batch size）设置为 4096。为了模型计算速度考虑，限定路径标签的最大

[2]https://nlp.stanford.edu/software/lex-parser.html
[3]https://github.com/OpenNMT/OpenNMT-py

System	LDC2015E86			LDC2017T10		
	BLEU	Meteor	chrF++	BLEU	Meteor	chrF++
(Konstas et al., 2017)*	22.00	-	-	-	-	-
(Cao and Clark, 2019)*	23.5	-	-	26.8	-	-
(Song et al., 2018)†	23.30	-	-	-	-	-
(Beck et al., 2018)†	-	-	-	23.3	-	50.4
(Damonte and Cohen, 2019)†	24.40	23.60	-	24.54	24.07	-
(Guo et al., 2019)†	25.7	-	-	27.6	-	57.3
(Zhu et al., 2019)(CNN-based)†	29.10	35.00	62.10	31.82	36.38	64.05
(Song et al., 2016)‡	22.44	-	-	-	-	-
Baseline1	25.13	33.08	59.36	26.98	34.36	61.05
+ Syntax	26.39	33.63	59.84	28.13	34.82	61.73
Baseline2	28.64	34.83	61.89	31.10	36.07	63.87
+ Syntax	**29.71**	**35.49**	**62.52**	**32.28**	**36.81**	**64.61**

表 2. 本文方法在LDC2015E86和LDC2017T10测试集上的实验结果及与其它模型的比较。* 代表seq2seq模型，† 代表graph2seq模型，‡ 代表其它模型。

长度为 4。解码时，默认的额外长度从50增加至150，该值表示模型解码时允许生成句子长度是源端最大长度加150。在所有实验中，以学习率 0.5 在 Tesla P40 GPU上训练 300K 步停止。本文实验代码已开源公布至https://github.com/Amazing-J/structural-transformer。

为了更好的体现本文方法的有效性，采用了BLEU(Papineni et al., 2002)，Meteor(Banerjee and Lavie, 2005; Denkowski and Lavie, 2014)，chrF++(Popović, 2017)三种评测指标。BLEU是基于语料级的评估，后两者是基于句子级的评估。相对来说，后两者的分数更接近于人工评测。

4.3 实验结果

表2给出了本文两个基准模型在融合了目标端句法信息前后AMR-to-Text生成的性能对比。从表2可以看出，融合目标端句法信息之后，AMR-to-Text生成的性能有着显著的提升。在两个基准模型上，分别提高了1.26和1.07（LDC2015E86），1.15和1.18（LDC2017T10）BLEU。这也有力的证明，在目标端融入句法信息，可以帮助模型学习到句子中潜藏的一些知识，从而在生成时考虑到句法信息的约束。该方法与融合源端句法和语义角色信息的机器翻译方法类似(Li et al., 2013)，本文进一步证明了在生成任务中目标端融入句法信息同样可以有着显著的提升。

表 2也给出了与其它现存模型在该任务上的性能比较。值得注意的是，LDC2015E86和LDC2017T10的验证集和测试集是相同的，区别只是训练集的数量相差了一倍左右。从表2可以看到，与seq2seq模型相比，本文的baseline1就已经显著的超越了它们，并且在融入句法信息（+Syntax）之后，性能依然有着明显的提升。目前最高的性能是Zhu等(2019)提出的Structure-Aware Self-Attention模型，本文在它们的基础之上也同样有着有效的提高，创造了新的最高的性能（SOTA）。可以证明本文的方法无论用在seq2seq模型或者graph2seq模型上都有效。

4.4 参数数量和训练时间

本文融合目标端句法信息的方法是将目标端句子替换为线性化结构句法树，不会对模型进行任何修改，也就意味着并不会给模型增加参数，这也是本文方法的一大优点。但是，目标端句子替换成线性化结构句法树之后，它的序列长度会相应的变长，这就会导致训练的时间略微增加。据统计，本文baseline1基准模型在LDC2015E86上进行训练，完场一轮训练的时间大概需要288秒（约4.80分钟），而融合目标端句法信息之后，大概花费345秒（约5.75分钟）。

4.5 融合不同形式句法信息的影响

从实验结果可以得到融合目标端句子的句法信息可以显著的提升AMR-to-Text生成的性能，但是为了探究哪种形式的句法信息对生成性能最为有效，本文做了进一步的实验分析。

一	(S (NP (DT the)DT (NN girl)NN (VP (VBD took)VBD (NP (NNS apples)NNS (PP (P from)P (NP (DT a)DT (NN bag)NN)NP)PP)NP)VP)NP)S
二	S NP the girl VP took NP apples PP from NP a bag
三	S NP DT the NN girl VP VBD took NP NNS apples PP P from NP DT a NN bag

表 3. 三种线性化结构句法树的示例

	一	二	三
BLEU	24.84	25.89	26.39

表 4. 三种线性化结构句法树在LDC2015E86测试集的性能对比

如表3所示，本文探索了三种线性化短语结构句法树对生成性能的影响。第一种形式包括了一个完整的句法树，不仅保留了句法树的所有节点，还给相应节点增加了结束标签，例如)NN,)NP,)S等。第二种形式则没有增加节点的结束标签，并且为了缩减句子的长度，把句法树中的词性标签删除，仅保留句法树的主干成份，剔除如DT，NN，VBD等词性标签。第三种形式则是在第二种形式的基础上保留了单词的词性标签。

本文在LDC2015E86的测试集上，对上述三种句法树的形式做了实验。从表4可以看出，当使用第一种形式时候，性能最差。因为它会将目标端的句子长度几倍的增长，极大地增加了模型的学习难度。从第二种和第三种形式得到的性能可以看出，保留短语结构句法树中的词性标签信息是对生成有着明显的贡献，有着0.5BLEU值的提升。本文则是使用的第三种短语结构句法树的形式。

5 结论

本文提出了一种直接而有效的方法融合目标端句子的句法信息，并且在最优的seq2seq模型Transformer上以及AMR-to-Text生成任务中最优的模型上都进行了实验。实验结果表明，使用该方法可以有效的对目标端句子的句法信息进行学习，从而提高AMR-to-Text生成的性能。在该任务最优模型的基础上同样也有着 1.07 和 1.18 BLEU值的提升，创立了新的最高性能。未来的工作中，由于句法分析和生成任务有着比较高的关联性，所以将会探索句法分析与AMR-to-Text生成任务之间的联合学习方向。

参考文献

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *Proceedings of 7th Linguistic Annotation Workshop & Interoperability with Discourse*, pages 178–186.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of ACL*, pages 65–72.

Daniel Beck, Gholamreza Haffari, and Trevor Cohn. 2018. Graph-to-sequence learning using gated graph neural networks. In *Proceedings of ACL*, pages 273–283.

Kris Cao and Stephen Clark. 2019. Factorising amr generation through syntax. In *Proceedings of NAACL*, pages 2157–2163.

Marco Damonte and Shay B. Cohen. 2019. Structural neural encoders for AMR-to-text generation. In *Proceedings of NAACL*, pages 3649–3658.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of WMT*, pages 376–380.

Jeffrey Flanigan, Chris Dyer, Noah A. Smith, and Jaime Carbonell. 2016. Generation from abstract meaning representation using tree transducers. In *Proceedings of NAACL*, pages 731–739.

DongLai Ge, Junhui Li, Muhua Zhu, and Shoushan Li. 2019. Modeling source syntax and semantics for neural amr parsing. In *Proceedings of IJCAI*, pages 4975–4981.

Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *Proceedings of ACL*, pages 140–149.

Zhijiang Guo, Yan Zhang, Zhiyang Teng, and Wei Lu. 2019. Densely connected graph convolutional networks for graph-to-sequence learning. *Transactions of the Association of Computational Linguistics*, 7:297–312.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *Proceedings of ACL*, pages 655–665.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of ACL, System Demonstrations*, pages 67–72.

Ioannis Konstas, Srinivasan Iyer, Mark Yatskar, Yejin Choi, and Luke Zettlemoyer. 2017. Neural AMR: Sequence-to-sequence models for parsing and generation. In *Proceedings of ACL*, pages 146–157.

Junhui Li, Resnik Philip, and Daumé III Hal. 2013. Modeling syntactic and semantic structures in hierarchical phrase-based translation. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 540–549.

Xiang Li, Thien Huu Nguyen, Kai Cao, and Ralph Grishman. 2015. Improving event detection with abstract meaning representation. In *Proceedings of the first workshop on computing news storylines*, pages 11–15.

Junhui Li, Deyi Xiong, and Zhaopeng Tu. 2017. Modeling source syntax for neural machine translation. In *Proceedings of ACL-2017*, pages 688–697.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The stanford corenlp natural language processing tooklkit. In *Proceedings of ACL-2014*, pages 55–60.

Arindam Mitra and Chitta Baral. 2016. Addressing a question answering challenge by combining statistical methods with inductive rule learning and reasoning. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Kishore Papineni, Salim Roukos, Ward Todd, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318.

Maja Popović. 2017. chrf++: words helping character n-grams. In *Proceedings of WMT*, pages 612–618.

Nima Pourdamghani, Kevin Knight, and Ulf Hermjakob. 2016. Generating english from abstract meaning representations. In *Proceedings of the 9th International Natural Language Generation conference*, pages 21–25.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of ACL*, pages 1715–1725.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *Proceedings of NAACL*, pages 464—-468.

Linfeng Song, Yue Zhang, Xiaochang Peng, Zhiguo Wang, and Daniel Gildea. 2016. Amr-to-text generation as a traveling salesman problem. In *Proceedings of EMNLP*, pages 2084–2089.

Linfeng Song, Xiaochang Peng, Yue Zhang, Zhiguo Wang, and Daniel Gildea. 2017. Amr-to-text generation with synchronous node replacement grammar. In *Proceedings of ACL*, pages 7–13.

Linfeng Song, Yue Zhang, Zhiguo Wang, and Daniel Gildea. 2018. A graph-to-sequence model for AMR-to-text generation. In *Proceedings of ACL*, pages 1616–1626.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Aleš Tamchyna, Chris Quirk, and Michel Galley. 2015. A discriminative model for semantics-to-string translation. In *Proceedings of the 1st Workshop on Semantics-Driven Statistical Machine Translation (S2MT 2015)*, pages 30–36, Beijing, China, July. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N.Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NIPS*, pages 5998–6008.

Oriol Vinyals, Lukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. 2015. Grammar as a foreign language. In *Advances in Neural Information Processing Systems 28*, pages 2773–2781.

Jie Zhu, Junhui Li, Muhua Zhu, Longhua Qian, Min Zhang, and Guodong Zhou. 2019. Modeling graph structure in transformer for better amr-to-text generation. In *Proceedings of EMNLP-2019*, pages 5458–5467.

基于神经网络的连动句识别

孙超 [1] ，曲维光 [1,2] ，魏庭新 [2,3] ，顾彦慧 [1] ，李斌 [2] ，周俊生 [1]
(1. 南京师范大学 计算机科学与技术学院，江苏省 南京市 210023
2. 南京师范大学 文学院，江苏省 南京市 210097
3. 南京师范大学 国际文化教育学院，江苏省 南京市 210097)

摘要

连动句是具有连动结构的句子，是汉语中的特殊句法结构，在现代汉语中十分常见且使用频繁。连动句语法结构和语义关系都很复杂，在识别中存在许多问题，对此本文针对连动句的识别问题进行了研究，提出了一种基于神经网络的连动句识别方法。本方法分两步：第一步，运用简单的规则对语料进行预处理；第二步，用文本分类的思想，使用 BERT 编码，利用多层 CNN 与 BiLSTM 模型联合提取特征进行分类，进而完成连动句识别任务。在人工标注的语料上进行实验，实验结果达到 92.71% 的准确率，F1 值为 87.41%。

关键词： 连动句 ； 文本分类 ； 神经网络

Recognition of serial-verb sentences based on Neural Network

SUN Chao[1], QU Weiguang[1,2], WEI Tingxin[2,3], GU Yanhui[1], LI Bin[2], ZHOU Junsheng[1]
(1.School of Computer Science and Technology,Nanjing Normal University,Nanjing, Jiangsu 210023,China;2.School of Chinese Language and Literature,Nanjing Normal University,Nanjing,Jiangsu 210097,China;3.International College for Chinese Studies, Nanjing Normal University，Nanjing,Jiangsu 210097,China)

Abstract

Serial-verb sentence is a sentence with several coordinated verbs in it. As a special syntactic structure it is very common and frequently used in modern Chinese. The grammatical structure and semantic relationship of serial-verb sentences are very complicated, which brings obstacles in its automatic recognition. This paper focuses on the recognition of Serial-verb sentence and proposes a recognition model based on neural networks. This method is implemented in two steps: the first step is to use rules to preprocess the corpus; the second step is to use BERT, the multi-layer CNN and the BiLSTM model to jointly extract features for classification, and then complete the sentence recognition task. Experimental results show that our model performs good in serial-verb sentence recognition,and the accuracy reaches 92.71% accuracy, while the F1 value reaches 87.41%.

Keywords: Serial-verb sentence , text classification , neural network

收稿日期： ；**定稿日期：**
基金项目：国家自然科学基金“汉语抽象意义表示关键技术研究”(61772278)；江苏省高校哲学社会科学基金“面向机器学习的汉语复句语料建设研究”(2019JSA0220); 国家社会科学基金“中文抽象语义库的构建及自动分析研究”（18BYY127）。

1 引言

通常情况下，动词是句子理解的关键，对句子的理解一般从动词入手。现代汉语中动词连用的现象大量存在，但相似形式所代表的语法结构和语义结构却不一定相同，有时甚至千差万别。在句子级别的语义研究方面，谓词所处的事件框架中包含的各种语义关系就构成了句子的语义结构。连动句包含多个谓词，蕴含了十分丰富的知识。连动句表示的是多个事件, 这些事件相互依赖并呈现出语义上的方式、顺承、目的、因果等关系，相互依赖影响并产生相互关联的事件 (刘雯旻, 2017)。因此获取连动句的方法将在自然语言理解领域中发挥重要的作用，有效的连动句识别可以在大规模语料中获取其中的连动句，从而便于对连用的动词或动词性短语之间的语义关系进行研究，有助于自然语言处理中句子级别的语义分析任务的研究和句法解析任务的研究，获取连动句的方法将在常识获取、智能网页等人工智能应用领域中发挥重要的作用。同时也为其他特殊句式的获取和处理提供了思路，从而帮助人们更加深入地理解自然语言。

连动句是汉语中一种较特殊的句式结构，在汉语中非常普遍。对连动句的研究可以从马建忠的《马氏文通》(成书于 1898 年) 中找到最早的踪迹。随后，又有很多语言学家都对连动句作了深入研究，其中，赵元任（《北京口语语法》）、张志公（《汉语语法常识》）、丁声书等（《现代汉语语法讲话》）、吕叔湘（《现代汉语八百词》）等都曾对连动结构做过分析和界定 (洪淼, 2004)。一般认为，连动句是指句中谓语为连动谓语的句子，即这个句子的谓语是两个或两个以上的动词连用，这些动词之间没有联合、动宾、偏正或主谓等关系，也没有明显的语音停顿，不用关联词语，而且这些动词都由同一个主语发出 (韩志玲, 倪蓉, 2012)。

综合前人的研究 (许有胜, 2007; 彭国珍 et al., 2013; 洪淼, 2013; 陈波 et al., 2013)，本文将所研究的连动句定义如下：在一个单句中，含有两个或两个以上的动词（或动词结构）且动词的施事为同一对象。其中第一个动词（简称 V1）的主语位置出现的名词短语 NP1 位置固定，而第二个动词结构（简称 V2）的名词短语 NP2，与 V1 的 NP1 同形，且必须隐含。其句法格式为：NP1+V1+(NP2[NP1])+V2，且 V1 和 V2 之间在语义上具有时序、目的、方式和原因等关系。本文所研究的连动句是形如“我去图书馆看书”，此句中的“去”和“看书”两个动词的施事都是“我”且只出现第一个动词的主语位置，“去”和“看”具有时序关系，且二者皆为动作行为动词，而像“地面人员看到丁毅找准跑道”一句中也含有两个动词“看到”和“找准”，但两者的施事分别是“地面人员”和“丁毅”，所以此句属于本文定义的非连动句。

抽象语义表示（abstract meaning representation, AMR）是近年来新兴的一种句子级的语义表示方法，突破了传统的句法树结构的限制，将一个句子语义抽象为一个单根有向无环图，很好地解决了论元共享的问题 (曲维光 et al., 2017)。而在连动句中存在着内部概念节点论元共享的现象，即在单句中多个谓词共享同一论元角色，这种 V1、V2 间的施事主语共享现象正是连动句区别于其他特殊句法结构的最主要特征，AMR 会将缺省的论元进行补全，得到完整语义表示。基于此特征可以从 AMR 图中获取可能的连动句，再经过人工校对得到连动句集合和非连动句集合，故本文选取了李斌等 (2017) 设计建立的中文 AMR 语料作为部分实验数据集。

本文的数据集主要有两部分组成，除使用抽象语义表示（AMR）体系标注的人教版小学 1-6 年级语文教材外；另一部分是清华树库的语料，经过人工标注赋予每个句子是连动句或非连动句的标签。

本文提出的连动句的识别方法分为两步：第一步：首先利用简单的规则剔除掉语料中一部分非连动句；第二步：基于神经网络做文本分类，将连动句与非连动句看作两个类别的文本进行分类，使用 BERT 编码，利用多层 CNN 和 BiLSTM 模型联合提取特征，进行句子分类，标签为“连动句”的文本，即为模型识别出的连动句。在此方法中，不需要手工筛选复杂特征，同时降低了对 NLP 领域的前置知识的需求。文本分类的实验效果达到 92.71% 的准确率，连动句识别的 F1 值为 87.41%。同时本工作也可以帮助 AMR 标注工作，定位连动句的位置，在后续工作中完成连动句中连动词和主语的识别以及连动词的语义关系识别，即可实现连动句的 AMR 自动标注。

2 相关工作

近年来针对连动句的研究主要集中在连动句对外教学的研究以及从汉语言文学角度研究分析连动句的句法和语义问题，针对连动句识别的研究工作较少。

许有胜 (2013) 提出了连动结构的自动识别和分析的方法。他主要从形式特征和语义角色两个方面编制出一些规则，尝试对连动结构进行自动识别和分析。但是由于连动结构的复杂性，

所设置的规则并不能涵盖所有情况，使得他提出的自动识别方法在很多环节的处理都存在问题，但他提供了一种基于“规则识别”的思路。

刘雯旻, 张晓如 (2017) 提出了一种基于规则和统计的连动句识别方法，他们构建了基于连动句形式特征和语义角色的基础规则库和被动名词库，利用互信息计算谓语动词与主语候选项的搭配强调，从而达到连动句识别的目的，实验结果达到 79.42% 的准确率,F1 值为 70.83%。

随着深度学习的发展，神经网络在解决文本信息处理相关任务中取得了较大的进展。本文将连动句的识别问题视为文本分类问题，提出一种基于神经网络的识别方法。深度学习的文本分类方法需要将文本输入到一个深度网络中，得到文本的表示形式，然后将文本表示形式输入到 softmax 函数中，得到每个类别的概率。目前，利用神经网络进行文本分类已经取得很多进展。Kim 最早提出将 CNN 应用于文本分类任务 (Kim Y, 2014)，Lai 等提出了 RCNN 模型，更好地利用了上下文信息 (Lai S et al., 2015)，Conneau 等在此基础上提出 VDCNN 模型，采用了深度卷积网络方法 (Conneau A et al., 2017)。Liu 等针对文本多分类任务提出基于 RNN 的不同共享机制模型 (Liu P et al., 2016)，Wang 等提出了 DRNN 模型，通过固定信息流动的步长提高文本情感分析的准确率 (Wang B, 2017)。

虽然许多神经网络的模型在文本分类任务中都取得了不错的表现，但连动句与非连动句的分类又与一般的文本分类任务不同，许多形式上十分相似的句子很可能不属于同一类别，更需要关注句中的动词间的语义关系和它的施事。基于此，本文利用 BERT 得到文本的表示形式，在训练过程中可以根据上下文动态的调整词向量，将其与多层 CNN 和 BiLSTM 进行组合作为连动句识别的模型，在人工构建的语料库上取得不错的效果。

3 模型设计

本模型采用 BERT 的输出结果作为字表示，将 BiLSTM 与两层 CNN 获取的局部特征相结合，用 Concatenate 连接，再经过一个全连接层，最后通过 softmax 层输出最终的判断结果，整体模型图如图 1所示。

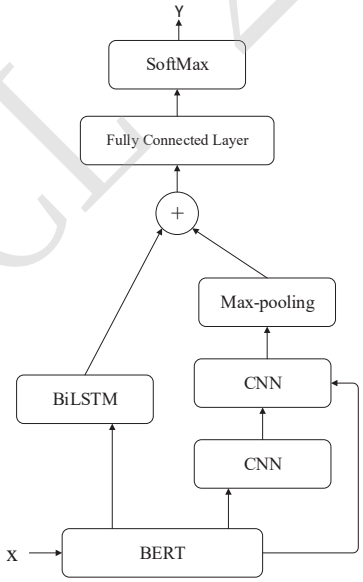


Figure 1: 模型设计

3.1 文本表示

BERT 是基于 Transformer 的双向编码器表示（Bidirectional Encoder Representation from Transformers） (Devlin J et al., 2018), 旨在通过联合调节所有层中的上下文来预先训练深度双向表示。使用双向的 Transformer 进行编码，使得在处理每个词的表示时都要考虑上下文信息，具体模型图如图 2所示。同时 BERT 预训练过程中使用了 Masked LM 和 Next Sentence Prediction 两种方法，迫使模型更多地依赖于上下文信息去预测词汇和句子，并且赋予了模型一定的纠错能力，分别捕捉词语和句子级别的 representation。

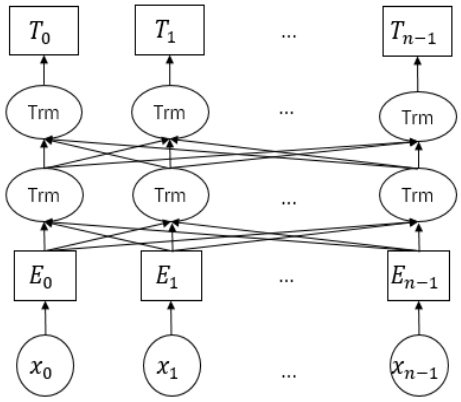


Figure 2: BERT 模型图

BERT 模型的输入不仅仅是字本身，它由三个 embeddings 向量相加而成，包含更多信息，输出则是已经融入全句语义的各个字的向量表示。BERT 输入向量表示示意图如图 3所示，Token Embedding 层将各个字转换成固定维度的向量，在 BERT 中每个字都会被转换成 768 维向量表示。BERT 能够处理输入句子对，segment embedding 层的作用是区分两个句子，前一个向量将 0 赋给第一个句子中的各个 token，后一个向量是把 1 赋给第二个句子中的各个 token。在本文中，输入的是一个句子，所以 segment embedding 全为 0。Position Embedding 实现编码序列的顺序性，当一个句子同一个字出现多次时，position embedding 提供了不同的向量。最终对“我去图书馆看书”一句将得到 3 个维度为（1,9,768）的向量，3 个向量按位相加最终得到大小为（1,9,768）的合成表示，富含更加丰富的语义信息。

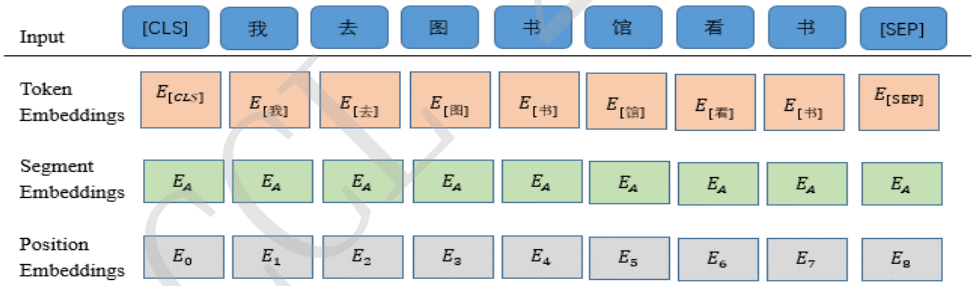


Figure 3: BERT 输入向量表示

3.2 特征提取

3.2.1 BiLSTM 层

将待判断的单句进行字级别的编码后将结果送入 BiLSTM 层，BiLSTM 将利用字在句子中的前后顺序，同时还可以捕获双向的较长距离的语义依赖关系，从而更好地判断当动词间相隔较远情况下是否可以构成连动关系。

BiLSTM 可以学习输入词的前后信息，从而有助于分类。给定由 n 个字组成的句子 X，它表示为一组向量 $(x_0, x_1, ..., x_{n-1})$，通过公式组（1）-（5）计算每个时间 t 的 LSTM 单元 (Chung J et al., 2018)。

$$i_t = \sigma(W_{x_i} \cdot x_t + W_{h_i} \cdot h_{t-1} + W_{c_i} \cdot c_{t-1} + b_i) \tag{1}$$

$$f_t = \sigma(W_{x_f} \cdot x_t + W_{h_f} \cdot h_{t-1} + W_{c_f} \cdot c_{t-1} + b_f) \tag{2}$$

$$c_t = (1 - i_t) \bigodot c_{t-1} + i_t \bigodot tanh(W_{x_c} \cdot x_t + W_{h_c} \cdot h_{t-1} + b_c) \tag{3}$$

$$o_t = \sigma(W_{x_o} \cdot x_t + W_{h_o} \cdot h_{t-1} + W_{c_o} \cdot c_t + b_o) \tag{4}$$

$$h_t = o_t \bigodot tanh(c_t) \tag{5}$$

其中 x_t，h_{t-1}，c_{t-1} 表示输入，h_t 和 c_t 表示输出。i_t、o_t、f_t 分别表示输入门、输出门和遗忘门。c_t 表示记忆单元向量。W_i、W_o、W_c 分别表示输入词向量 x_t，隐藏层状态 h_t 和记忆单元 c_t 的权重矩阵，b_i、b_f、b_c 和 b_o 分别表示偏差向量。$\bigodot$ 表示按位乘操作，σ 表示 sigmoid 激活函数。

通过 LSTM 可以得到与句子长度相同的隐层状态序列 $\{h_0, h_1, ..., h_{n-1}\}$，将前向 LSTM 与后向的 LSTM 结合成为 BiLSTM。通过对正向的时间序列和反向的时间序列进行训练，使输出的数据包含上下文的信息。解决了 LSTM 网络缺乏对上下文的联系，从而使模型获取更多的上下文信息 (Mike Schuster and Kuldip K Paliwal, 1997)。本文中使用的是两个 BiLSTM 堆叠形成的模型，中间使用一个全连接层进行降维，前一层 BiLSTM 的输出作为下一层 BiLSTM 的输入。

3.2.2 CNN 层

卷积神经网络是神经网络中提取局部特征的一种网络 (Wang J et al., 2017)，具有强大的特征学习和表示能力，基本结构由四层构成，分别是输入层、卷积层、池化层、全连接层。本模型中使用了两层 CNN 网络进行串联，第一层 CNN 的输入使用 BERT 的输出，将第一层 CNN 的输出和 BERT 的输出进行拼接作为第二层的输入。卷积层本质上是特征提取器，输入经过过滤器进行卷积操作后得到新的特征。设滤波器 $W \in \mathbb{R}^{m \times n}$，卷积得到：

$$y_{i,j} = \sum_{u=1}^{m} \sum_{v=1}^{n} w_{uv} x_{i-u+1, j-v+1} \tag{6}$$

其中，$m \ll M$，$n \ll N$。另外在卷积的标准定义基础上，根据不同任务的需求，可调整滤波器的滑动步长 (stride)、引入零填充 (zero padding) 来增加卷积的多样性，更灵活地进行特征抽取。两层滤波器采用不同的滑动步长，获取不同尺度的局部特征信息，有利于捕获句中不同距离动词间的信息。卷积层 (Wang S et al., 2018) 通过局部连接大大减少了网络参数的数量，通过权重共享使特征提取与数据位置无关，但其输出的神经元个数并没有显著地减少，容易造成过拟合，所以在卷积层之后再加上一个池化层，使用 Max-pooling 进行降维，同时增加平移不变性，模型更关注是否存在某些特征而非其位置，使得网络对一些细小的局部形态改变保持不变性，在减少数据量的同时保留有用的信息，最终得到固定长度的输出。

3.3 分类预测

将经过 CNN 层获得的特征与经过 BiLSTM 层获得的特征进行拼接，通过全连接层将特征整合到一起，同时对网络进行 Dropout 处理，以防止过拟合，随后送入 Softmax 进行预测。由于在数据集中连动句与非连动句分布不均，连动句数量较少，在计算损失函数时使用加权损失函数，使模型更多的关注样本数较少的类，更有利于模型识别连动句。

4 实验设置

4.1 连动句识别流程

本文对连动句识别流程如图 4所示，第一步，首先需要对语料进行切分操作，本文对连动句的识别是以单句为单位，以标点符号“。”、“，”、“：”、“；”为切分依据；之后再对语料进行词性标注工作。本文中使用的词性标注器是使用清华树库语料作为原始语料、用 BiGRU-CRF 模型自行训练所得，该模型可以实现对动词的细分类。本文采用范晓的《汉语动词概论》的分类系统 (范晓, 1980)，动词可根据它的表义功能分为动作行为动词、心理动词、使令动词，存现动词、判断动词、能愿动词、趋向动作动词、先导动词。动作行为动词和心理动词可充当连动

句中的 V。像“科长的口袋一下子鼓了起来”一句中，“起来”被标注为趋向动作动词，所以本文认为该句中只含有一个动词“鼓”。更加精准的词性标注，有利于语料在进行规则筛选时预先识别出更多的非连动句。由于在真实语料中，连动句与非连动句所占比例相差很大，且大部

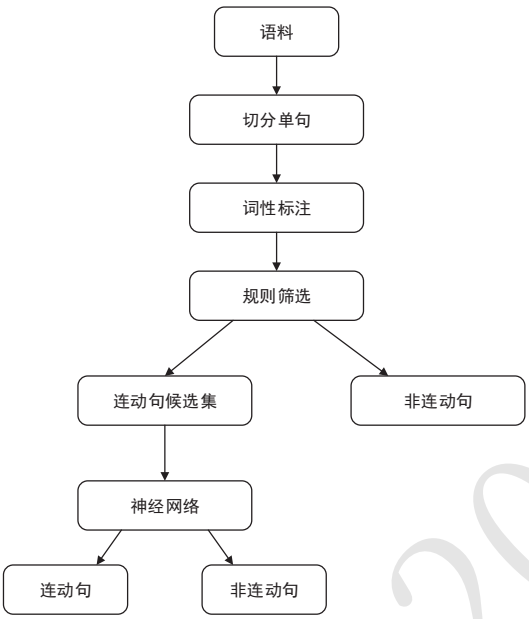


Figure 4: 连动句识别流程

分的简单非连动句（句中不含动词或只含有一个动词或动词结构）比较容易分辨，为了使神经网络模型将注意力放在学习与连动句在形式上比较相似的非连动句上，更好地学习两者的特征，本文首先制定了相应规则对语料进行预处理，预先识别出部分非连动句，其余句子可能为连动句，作为连动句的候选集送入神经网络进行分类。

本文设计规则将预先筛选的以下三种条件的句子排除在候选集之外：

条件一：不含动词或只含有一个动词（或动词结构）的句子。

条件二：含有关联词的紧缩句或复句。例如“常海一进病房就大咧咧地坐到程信的病床上”，句中含有“一…就…”形似连动，但该句属于紧缩句。

条件三：只含有多个动词和虚词的句子。当一个长句经过标点切分为独立小句或有些小句充当标题时会只有动词和虚词构成，例如“竞争与冲突”、“搜索前进”缺乏上下文信息，无法判断动词的主语，进而无法判断其是否为连动句。

通过“第一步”简单规则的辅助，可识别出大量非连动句，使连动候选集中连动句与非连动句的占比差距大大缩小，从相差 14 倍多缩小到不到 3 倍，表 1展示了筛选前后连动句与非连动句的数据的变化。

	连动句	非连动句	连：非连
处理前	7200	103188	1:14.3
处理后	7200	17852	1:2.5

Table 1: 规则筛选处理对比表

经过规则筛选后得到的连动句候选集中除连动句外，还含有大量的非连动句。这些非连动句中有一部分是和连动句相差明显的句子，例如“我在天色微明时走到了杨柳镇”根据语义可以得到“微明”和“走到”的施事明显是不一致；但还有部分与连动句相似度很高的其他特殊句式，例如，兼语句“他帮助妇女摆脱贫困”，虽然此处“帮助”和“摆脱”的施事并不一致，但此类句子在句式上与连动句相似，有一定区分难度。又如含有被动语义的句子，在判断动词的施事时也可能遇到困难，例如“他被通知住院”，“通知”和“住院”的施事并不相同，但在“他被诊断宣判生命只剩 2 个月”中“诊断”和“宣判”的施事又是同一个，该句为连动句，这些相似度极大的句子给本文的工作带来困难，也是神经网络模型需要重点学习的内容。

4.2 评价标准

本实验采用精确率（Precision）、召回率（Recall）、F1 值（F1-measure）、准确率（Accuarcy）作为评价标准。其中，TP：正确分类中连动句个数；FP：错误分类中连动句个数；TN：正确分类中非连动句个数；FN：错误分类中非连动句个数。

$$P = \frac{TP}{TP + FP} \tag{7}$$

$$R = \frac{TP}{TP + FN} \tag{8}$$

$$P = \frac{2 \cdot P \cdot R}{P + R} \tag{9}$$

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

4.3 实验数据

连动句中 V1 和 V2 有共同的主语，在传统的语义表示方法上，由于树结构的限制，一般只标注主语和核心谓词的关系，而连动结构中其他谓词与主语的关系则被隐含。但这种隐含的语义关系正是连动结构区别于其他特殊句法结构最重要的特点。AMR 将补充出句中省略或隐含的成分，以还原出较为完整的句子语义，弥补传统句法表示的严重缺陷。

例如：“小白兔连忙挎起篮子往家跑。”一句的 AMR 图示注如图 5，“挎”和“跑”的主语都是“白兔”，原句中第二个动词“跑”的主语被省略了，在 AMR 图中会将此缺省补全，同时它也表示出了两个动词间的语义关系，此例中两个动词间的语义关系为“temporal（时序）”。

```
x1_小 x2_白兔 x3_连忙 x4_挎 x5_起 x6_篮子
x7_往 x8_家 x9_跑
(x11 / temporal
    :arg1()  (x4_x5 / 挎-01
          :arg0()  (x2 / 白兔
                :arg0-of()  (x1 / 小-01))
           :arg1()  (x6 / 篮子)
          :manner()  (x3 / 连忙))
     :arg2()  (x9 / 跑-01
          :arg0()  (x2 / 白兔)
         :arg3(x7/往)  (x8 / 家))
)
```

Figure 5: AMR 标注示例

本文根据 AMR 图抽取出小学语文 1-6 年级课本中的连动句，剩余句子为非连动句，同时又对清华树库的语料进行人工标注，共计 40667 个完整的句子。将句子切分为独立小句后，得到 11 万句分句，其中 7200 个独立小句为连动句。经过“第一步”处理后，共计 25052 个独立小句进行“第二步”神经网络模型的实验，按照 6:2:2 的比例划分训练集、开发集和测试集。

4.4 实验参数设置

使用 BERT 的基础版本，网络层数设置为 12，隐藏层数设置为 768，Self-Attention Head 设置为 12。在 BERT 中要预先设置 max_seq_length 参数，未达到此长度的句子要做 padding 处理，而超过此长度的数据将会被截断，造成信息丢失。同时若此参数设置过大会占用大量内存空间。本实验主要参数设置如表 2所示。

参数名	参数值
句子最大长度	50
神经元丢弃率	0.5
学习率	0.0001
提前终止	3

Table 2: 模型参数设置

4.5 实验结果及分析

为验证本文提出的方法的有效性，本实验主要与以下几种目前流行的文本分类模型进行对比，实验结果如表 3所示。

(1) 基于规则和统计：刘雯旻, 张晓如在 2017 年提出，他们构建了基于连动句形式特征和语义角色的基础规则库和被动名词库，利用互信息计算谓语动词与主语候选项的搭配强度，在他们人工标注的数据集进行实验。

(2)FastText：利用简单的三层模型（输入层、单层隐藏层、输出层），根据上下文预测文本的类别 (Joulin A et al., 2016)。

(3)TextCNN：利用 CNN 来提取句子中的关键信息，先将文本分词做 embedding 得到词向量, 再将词向量经过一层卷积, 一层 max-pooling, 最后将输出外接 softmax 实现文本类别的预测。

(4)TextRNN：RNN 模型由于具有短期记忆功能, 它通过前后时刻的输出链接保证了“记忆”的留存，引入门控机制解决长期依赖问题，捕获输入样本之间的长距离联系。

(5)BERT：用 Transformers 作为特征抽取器的深度双向预训练语言模型，在许多自然语言处理任务有很好的表现。

由表 3不同文本分类模型进行连动句识别的结果可知，本文提出的模型在连动句与非连动句分类的任务上具有很好的效果。除基于规则和统计的方法使用作者标注的语料外，其余神经网络的模型均使用本文中介绍的利用简单规则筛选后的语料。对比结果发现，FastText 模型基本没能学习到连动句的特征，在本任务上的效果较差；TextCNN 和 TextRNN 的效果相差不大，但都表现的还不够理想；而 BERT 模型 F1 较之前的模型有较大的进步，通过进一步分析 BERT 模型识别错误的句子发现，BERT 模型对长句的识别效果比较差。“连动句”这种语言现象可以出现在任何领域，它关注的是动词与动词的发出者之间的关系，而不是整个句子的语义关系，而且与词序有关。本文提出的模型使用 BERT 编码使模型获得了更多的语义信息，BiLSTM 层可以提取上下文不同距离的语义化信息, 同时 CNN 可以获取局部的特征，将多种特征进行组合，从而完成对连动句与非连动句的区分。

模型	连动句			ACC
	P	R	F1	
基于规则和统计	75.48%	66.72%	70.83%	79.42%
FastText	40.30%	98.24%	57.15%	41.41%
TextCNN	66.70%	68.68%	67.68%	74.09%
TextRNN	81.39%	57.58%	67.45%	77.89%
BERT	79.94%	79.18%	79.56%	89.03%
本文	86.78%	88.04%	87.41%	92.71%

Table 3: 不同文本分类模型结果对比表

在时间消耗方面，本文所提出的模型的收敛速度很快，图 6和图 7展示了模型在开发集上的迭代次数与 loss 和 acc 的变化曲线, 由图像可知模型在迭代几轮后便可得到在开发集上效果最好的模型参数，之后 loss 值会发生小范围的波动，为防止模型训练造成过拟合的问题，所以在实验中设置了提前终止参数，并使用 dropout 使模型得到更好的泛化效果。

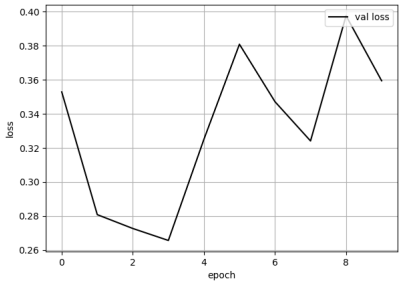


Figure 6: loss 曲线图

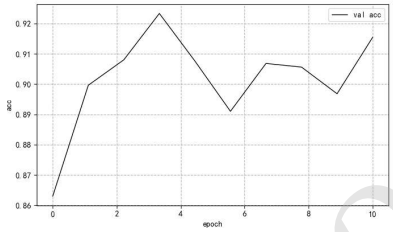


Figure 7: acc 曲线图

同时为了验证模型各层结构在实验中所起到的效果，特设置消融实验，结果如表 4 所示。由实验结果可知，使用随机初始化的 char 级别的词向量代替 BERT，实验的 F1 降低了约 20%，可见语义信息的获取和使用在连动句识别中起到了至关重要的作用，同时在训练过程中，本文提出的模型也会对词向量进行微调，以达到更好的表现。与此同时本文在获取特征时也采取了局部特征和全局特征组合的方式，更有助于连动句识别。通过实验发现，在使用 BERT+CNN 模型时，实验的 R 值较高，这说明模型可以尽量多将连动句挑选出来，将连动句误分为非连动句的情况较少，但同时它也将部分非连动句错误地识别为连动句，这是因为 CNN 侧重于提取句子局部信息，当句子局部出现两个动词或动词短语时，这一特征就会被 CNN 捕捉到，然而并非所有两个动词短语连用都是连动结构，可能是紧缩复句、兼语句、动词短语做宾语句等其他语言现象，因此造成这一模型 R 值较高而 P 值较低。而使用 BERT+BiLSTM 的模型则恰好不同，它的 P 值较高，BILSTM 侧重于捕捉句子全局信息，从整句角度去考察句子特征，因此很容易将复句、兼语句、动词宾语句等在句子层面排除在外，然而连动结构除了在句法层出现外，还可以以短语形式出现在各种句式结构的多个句法位置，BILSTM 模型对这类连动结构的识别能力较差，导致 R 值较低。本文使用的模型将两者的优势集中起来，提高了模型的 F1 值。

模型	连动句			ACC
	P	R	F1	
CNN+BiLSTM	60.80%	76.10%	67.60%	72.58%
BERT+CNN	75.71%	**92.90%**	83.43%	89.66%
BERT+BiLSTM	**88.33 %**	76.36%	81.91%	90.30%
BERT+BiLSTM+CNN	86.78%	88.04%	**87.41%**	**92.71%**

Table 4: 消融实验结果

根据实验结果我们发现连动句的识别错误分为两种，其一是非连动句错误识别为连动句，主要分为以下几种情况：

（1）汉语中一些动词的主语并非动词的施事者，导致模型判断出错，例如“出租车招手即停”一句中，句子的主语是“出租车”，但“招手”的施事是“人”而“停”的施事是“出租车”，二者并不相同。

（2）部分动词或动词短语做状语的状中结构和动词或动词短语做宾语的动宾结构识别易出错，它们在形式上与连动句相似，例如“宋东山心平气和地向小伙子笑笑”，此句中“心平气和”和“笑笑”的施事皆为“宋东山”，但该句为状中结构而非连动句；又如“川川总爱刨根问底”，“刨根问底”为动词，充当“爱”的宾语，且二者的施事都为“川川”，此句为动宾结构。

其二是某些连动句无法识别出来，主要分为以下两种情况：

（1）对多义词的识别效果不好，汉语中很多词语存在一词多义现象，但有些词语模型无法识别出它的动词义项，导致模型判断出错。例如“小刚看见这句话火了”, 火有多个义项，可以是名词、动词、形容词，但此处 embedding 未能准确表达出它为动词的语义。

（2）对长句的识别效果不好。例如“海淀区红山口甲 3 号国防大学医院疑难病研究中心的法集河使用近百味中药炮制膏药”模型识别为非连动句，但对“法集河使用近百味中药炮制膏药”可正确识别其为连动句。当句子某些修饰成分过长时，会影响模型的识别效果。

5 总结展望

本文根据连动句定义标注了连动句数据集，介绍了一种基于神经网络的连动句识别方法，先对语料进行切分和词性标注工作，再通过简单的规则进行第一轮非连动句的判断，之后使用 BERT 编码，将 BiLSTM 和 CNN 模型获取的特征进行组合，进行第二轮连动句与非连动句的判断，进而完成连动句的识别任务。实验表明，该模型取得了不错的识别效果。

本文的下一步工作是进一步提高连动句识别的准确率，同时对语料中识别出的连动句进一步找出其中的连动词，并识别它们之间的语义关系，从而帮助处理 CAMR 中连动句式的标注与解析工作。

参考文献

陈波, 姬东鸿, 吕晨. 2013. 基于特征结构的汉语连动句语义标注研究 [J]. 中文信息学报, 27(05):60-66+74.

范晓. 1980. 汉语的句子类型 [M]. 太原：书海出版社.

韩志玲, 倪蓉. 2012. 原型理论启发下的现代汉语连动句类型研究 [J]. 上海理工大学学报 (社会科学版), 34(01):41-45.

洪淼. 2004. 现代汉语连动结构研究 [D]. 南京师范大学.

洪淼. 2004. 现代汉语连动句式的语义结构研究 [J]. 西南民族大学学报 (人文社科版), 25(007):423-426.

李斌, 闻媛, 宋丽, 卜丽君, 曲维光, 薛念文. 2017. 融合概念对齐信息的中文 AMR 语料库的构建 [J]. 中文信息学报, 31(06):93-102.

刘雯旻. 2017. 基于汉语连动句的常识获取方法研究 [D]. 江苏科技大学,

刘雯旻, 张晓如. 2017. 一种基于规则和统计的连动句识别方法 [J]. 电子设计工程, 25(22):18-22.

彭国珍, 杨晓东, 赵逸亚. 2013. 国内外连动结构研究综述 [J]. 当代语言学, 15(03):324-335+378.

曲维光, 周俊生, 吴晓东, 戴茹冰, 顾敏, 顾彦慧. 2017. 自然语言句子抽象语义表示 AMR 研究综述 [J]. 数据采集与处理, 32(01):26-36.

许有胜. 2007. 连动结构研究综述 [J]. 兰州学刊, （09）：137-142.

许有胜. 2013. 连动结构的自动识别和分析 [J]. 巢湖学院学报, 15(04):108-115.

Chung J , Gulcehre C , Cho K H ,Bengio Y. 2018. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling[J]. In *arXiv preprint arXiv,1412.3555.*

Conneau A , Schwenk H , Barrault L , Yann Lecun. 2017. Very Deep Convolutional Networks for Text Classification[C]// In *Proceedings of the 15th Conference of the European Chapter of the Association for Compu-tational Linguistics EACL*,pages 1107-1116.

Devlin J, Chang M-W, Lee K, et al. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[C]// In *The 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*,pages 4171-4186.

Joulin A , Grave E , Bojanowski P , et al. 2016. Bag of Tricks for Efficient Text Classification[C]// In *Proceedings of the fifty-fourth Annual Meeting of the Association for Computational Linguistics*,pages 427-431.

Kim Y. 2014. Convolutional Neural Networks for Sentence Classification[C]// In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*,pages 1746-1751.

Lai S, Xu L, Liu K,et al. 2015. Recurrent convolutional neural networks for text classification[C]// In *Proceedings of the twenty-ninth AAAI Conference on Artificial Intelligence*,pages 2267-2273.

Liu P, Qiu X, Huang X. 2016. Recurrent Neural Network for Text Classification with Multi-Task Learning[C]// In *Proceedings of the Twenty-fifth International Joint Conference on Artificial Intelligence*,pages 2873–2879.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural network [J]. In *IEEE Transactions on Signal Processing*,45(11):2673-2681.

Wang B. 2018. Disconnected Recurrent Neural Networks for Text Categorization[C]// In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*,pages 2311-2320.

Wang J, Wang Z, Zhang D, Yan J. 2017. Combining knowledge with deep convolutional neural networks for short text classification[C]// In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*,pages 2915-2921.

Wang S, Huang M, Deng Z. 2018. Densely Connected CNN with Multi-scale Feature Attention for Text Classification[C]// In *Proceedings of the Twenty-Seventh In-ternational Joint Conference on Artificial Intelligence*,pages 4468-4474.

融合全局和局部信息的汉语宏观篇章结构识别

范亚鑫[1]，蒋峰[1]，褚晓敏[1]，李培峰[1 2]，朱巧明[1 2]
[1]苏州大学计算机科学与技术学院，苏州，中国
[2]苏州大学人工智能研究院，苏州，中国
{20194227042, 20194027003}@stu.suda.edu.cn, {xmchu, pfli, qmzhu}@suda.edu.cn

摘要

作为宏观篇章分析中的基础任务，篇章结构识别任务的目的是识别相邻篇章单元之间的结构，并层次化构建篇章结构树。已有的工作只考虑局部的结构和语义信息或只考虑全局信息。因此，本文提出了一种融合全局和局部信息的指针网络模型，该模型在考虑全局的语义信息同时，又考虑局部段落间的语义关系密切程度，从而有效地提高宏观篇章结构识别的能力。在汉语宏观篇章树库（MCDTB）的实验结果表明，本文所提出的模型性能优于目前性能最好的模型。

关键词： 宏观篇章分析；结构识别；自顶向下；指针网络

Combining Global and Local Information to Recognize Chinese Macro Discourse Structure

Yaxin Fan[1], Feng Jiang[1], Xiaomin Chu[1], Peifeng Li[1 2], Qiaoming Zhu[1 2]
[1]School of Computer Science and Technology, Soochow University, Suzhou, China
[2]AI Research Institute, Soochow University, Suzhou, China
{20194227042, 20194027003}@qq.com, {xmchu, pfli, qmzhu }@suda.edu.cn

Abstract

As the fundamental task in macro discourse analysis, the discourse structure recognition task aims to identify the structure between adjacent discourse units and build a discourse structure tree hierarchically. Existing work only considers local structural and semantic information or only global information. Therefore, this paper proposes a pointer network model that integrates global and local information. It can effectively improve the ability of macro text structure recognition by considering the global semantic information and the closeness of the semantic relationship between paragraphs. The experimental results in the Chinese macro discourse treebank show that the proposed model outperforms the state-of-the-art model.

Keywords: Macro Discourse Analysis , Structure Recognition , Top-Down , Pointer Network

1 引言

基金项目：国家自然科学基金(61772354,61836007,61773276);江苏高校优势学科建设工程资助项目

当前，自然语言处理的研究内容已经从词汇理解、句法分析等浅层语义分析领域延伸到深层语义理解的篇章分析领域。篇章分析是自然语言处理领域的重点和难点，其主要任务是从整体上分析一篇文章的逻辑结构和篇章单元之间的语义关系，进而从更深的层次挖掘自然语言文本的语义和结构信息。篇章分析有助于理解篇章的中心思想和主要内容，可以提升自然语言处理相关应用的性能，例如问答系统(Liakata et al., 2013)和自动文摘(Cohan and Goharian, 2015)等。

篇章分析的研究分析可分为微观和宏观两个层面。微观层面研究的是句子和句子、句群和句群之间的结构和关系；宏观层面研究的是段落和段落、章节和章节之间的结构和关系。当前篇章分析主要集中在微观层面，而宏观层面的研究较少。Chu et al. (2020)提出了一个宏观篇章结构表示体系，其中，以段落为基本篇章单元（Elementary Discourse Units,EDUs），相邻两个段落以篇章关系连接在一起，并构成更大的篇章单元（Discourse Units,DUs）,这些篇章单元层层向上，最终将一篇文章构成一棵完整的篇章结构树。

宏观汉语篇章树库（Macro Chinese Discourse Treebank,MCDTB）(Jiang et al., 2018b)对宏观篇章结构进行了标注。本文以MCDTB中的一篇文章（chtb_0282）来说明宏观篇章结构，如例1所示。其中，p_1介绍了推行公务员制度交流会的情况，p_2补充了会议时间以及参会人员；p_3讲述了李鹏总理肯定了推行公务员制度的成效，p_4讲述了李鹏总理提出推行公务员制度要依法办事；p_5补充其他参会人员。p_2补充了p_1描述的交流会的相关信息，因此p_1与p_2构成补充关系，p_2和p_4分别阐述了交流会的内容，因此构成了并列关系，其形成的篇章单元对上文（p_1和p_2构成的篇章单元）进行解说，形成解说关系，p_5是对全文的补充，即对p_1到p_4的信息进行补充。

p_1:国务院总理李鹏今天在中南海紫光阁会见中国推行公务员制度经验交流会全体代表时指出，推行公务员制度是中国政治体制改革的一项重要内容，是干部人事制度的重大改革，是建立社会主义市场经济体制的客观需要，要有领导、有步骤地加快推行步伐。
p2:这次推行公务员制度经验交流会是昨天开始召开的，各省、自治区、直辖市人事厅局长、国务院各部委、直属机构人事部门的负责人共一百二十多人出席了会议。
p3:李鹏肯定了一年来国家公务员制度推行工作取得的成效。他说，我们要认真总结和推广这些好的经验，建立起激励竞争和勤政廉政机制，建立一支以为人民服务为宗旨、密切联系群众、精干高效、廉洁奉公、忠于职守的国家公务员队伍，增强政府机关的生机和活力。
p4:李鹏提出，推行公务员制度，要按照《国家公务员暂行条例》依法办事，不能有随意性。要把这项工作作为政治体制改革的一件大事来抓，结合改革、精简机构来推行公务员制度；要形成公务员的新陈代谢机制，使青年人才不断地进入到公务员队伍当中。
p5:国务委员李贵鲜、罗干参加了会见。（完）

例1.李鹏强调要加快推行公务员制度

p_1-p_5构成的篇章结构树如图1所示。图中，叶子节点（p_1-p_5）为段落，即宏观篇章结构中的基本篇章单元（EDUs）；相邻叶子节点通过篇章关系联系起来，通过连接后构成的节点是篇章单元（DUs），表示两个基本篇章单元之间的关系；箭头指向的是核心，即重要的篇章单元。具体而言，篇章单元之间通过篇章关系相连接，最终形成一棵完整的篇章结构树。本文研究的主要内容就是识别相邻篇章单元之间的结构，并层次化构建篇章结构树。

在MCDTB语料库上，已有的篇章结构识别的研究(Jiang et al., 2018a; Zhou et al., 2019)都只考虑相邻两个篇章单元的语义关系，如果相邻两个篇章单元语义关系很接近，那么这两个篇章单元就会大概率以某种关系连接起来，形成一个更大的篇章单元，进而层次化的构建篇章结构树。但是这些研究都只考虑局部的上下文信息，而没有将整个文章的语义信息（全局信息）有效的运用到篇章结构识别任务中。

在RST-DT(Carlson et al., 2007)的篇章结构识别任务中,Lin et al. (2019)提到每次考虑相邻两个篇章单元容易受到局部信息的影响，而错误的相邻篇章结构判断会将错误的信息传播到上层，从而影响上层结构的识别。而Van (1980)的宏观篇章结构理论也指出宏观结构是更高层次的结构，表现为篇章整体的语义连贯，每一层的宏观结构都是由下层结构支撑起来的。篇章的

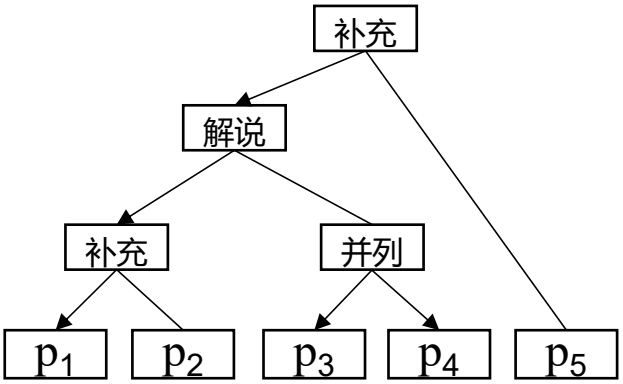


图 1. 宏观篇章结构树（chtb_0282）

宏观语义信息（即全局信息）往往能体现篇章的展开结构，可用于检验一个篇章是否连贯。因此本文认为在考虑局部信息的同时，全局信息也应该被考虑用来辅助篇章结构的识别。

基于以往的研究都只考虑局部的上下文信息，且受到宏观篇章结构理论的启发，本文提出一种融合全局和局部信息的指针网络模型，用于自顶向下的识别篇章结构，并构建篇章结构树。在该模型中，本文采用交互注意力机制捕获相邻两个段落之间的语义联系，即局部信息；指针网络的编码层用来捕获整个篇章的语义，即全局信息；而指针网络的解码层用来融合全局和局部信息，为两个段落之间的语义分配一个概率，概率越大，表明这两个段落之间的语义联系越弱，则需要进行篇章单元的切分。对切分形成的两个篇章单元，根据深度优先原则，递归地进行切分，从而自顶向下的构建完整的篇章结构树。在MCDTB上的实验结果表明，本文的模型优于目前性能最好的模型。

2 相关工作

在已有的研究工作中，无论中文还是英文都更注重微观篇章结构的分析，而对于宏观篇章结构的分析还处于起步阶段。涉及到宏观篇章结构的语料库主要有英文修辞结构篇章树库（RST Discourse Treebank,RST-DT）(Carlson et al., 2007)和中文的宏观汉语篇章树库（MCDTB）(Jiang et al., 2018b)。现将两个语料和相关模型介绍如下：

修辞结构篇章树库（RST-DT）以修辞结构理论（RST）为理论依据，标注了385篇《华尔街日报》文章。在该语料库的研究中，Hernault et al. (2010)提出了基于SVM的篇章分析器HILDA,该模型以贪婪的方式自底向上构建篇章结构树；Joty et al. (2013)等利用动态CRF模型分别构建了句子级别和篇章级别的分析器；Ji and Eisenstein (2014)参考深度学习的做法，采用线性变换将表面特征转换成隐空间通过移进规约进行篇章解析；Lin et al. (2019)采用指针网络，构建了一个句子级的篇章解析器，但上述研究都是在微观层面。在宏观层面，Sporleder and Lascarides (2004)对RST-DT修正和裁剪后采用最大熵模型进行了宏观篇章结构识别。

宏观汉语篇章树库（MCDTB）遵循RST修辞结构理论，对720篇文章进行了宏观篇章信息的标注，包括篇章结构、主次和语义关系等。在MCDTB上进行篇章结构识别，构建完整篇章结构树的研究不多。Jiang et al. (2018a)采用序列标注的思想，提出一个基于条件随机场的模型（LD-CM）。该模型对结构和主次进行联合学习，从而自底向上的构建篇章结构树；Zhou et al. (2019)提出了一个基于神经网络的模型（MVM）。该模型从多个角度匹配两个篇章单元之间的语义，从而识别篇章结构，并采用移进规约的方法构建篇章结构树。然而LD-CM是基于传统机器学习的方法，用到了较多的手工特征，考虑相邻两个篇章单元的语义联系；同样MVM也只考虑相邻两个篇章单元的语义联系。这两种方法都只考虑了局部的上下文信息，没有有效运用全局信息辅助篇章结构的识别。

3 PNGL模型

本文提出了一种融合全局和局部信息的指针网络模型（Pointer Network on Global and Local information）的模型自顶向下的识别汉语宏观篇章结构，其架构如图2所示。该架构包括三个部分:1)段落编码层（Paragraph Encoder Layer,PEL）,用来捕获段落的语义表示；2)段

落交互层（Paragraph Interactive Layer,PIL）,用来捕获相邻两个段落的语义联系，即局部信息；3)指针网络（Pointer Network），指针网络的编码层用来捕获整个篇章的语义表示，即全局信息，解码层融合局部和全局信息,用来识别篇章结构并自顶向下的构建篇章结构树。

对于一篇文章表示为$P = \{p_1, p_2, \cdots, p_m\}$,其中$p_i$是段落词语序列，m是文章的段落数。将$p_i$通过段落编码层（PEL），得到段落编码为$R = \{r_1, r_2, \cdots, r_m\}$。将相邻两个段落的编码通过段落交互层（PIL），得到表示相邻两个段落语义联系的表示$H = \{h_1, h_2, \cdots, h_{m-1}\}$,$h_i$表示段落$p_i$和$p_{i+1}$之间的语义联系的紧密程度，即得到了局部信息。同时将r_i平均池化之后通过指针网络编码层，编码层是双向GRU，最后一个时间步输出作为整个篇章的语义表示，即全局信息（例如e_5表示整个篇章的语义）。

指针网络解码层是单向GRU，本文根据深度优先的原则，使用栈来生成篇章结构树。在第t步，栈顶的篇章单元$DU_{(l,r)}$出栈。解码层的输入为篇章单元$DU_{(l,r)}$的语义表示e_r，即编码层第r步的输出；解码层的输出为d_t,d_t和局部信息H进行交互，通过计算注意力来融合全局信息和局部信息，从而为每一个h_i分配一个概率，其中$l \leq i \leq r-1$。概率越大，则表示段落p_i和p_{i+1}之间的语义联系越松散，则应该在p_i和p_{i+1}之间进行切分，形成新的篇章单元$DU_{(l,i)}$和$DU_{(i+1,r)}$。切分后段落数大于2的篇章单元入栈，递归地对栈顶篇章单元进行切分，直至栈空。根据切分得到的所有篇章单元构建篇章结构树。

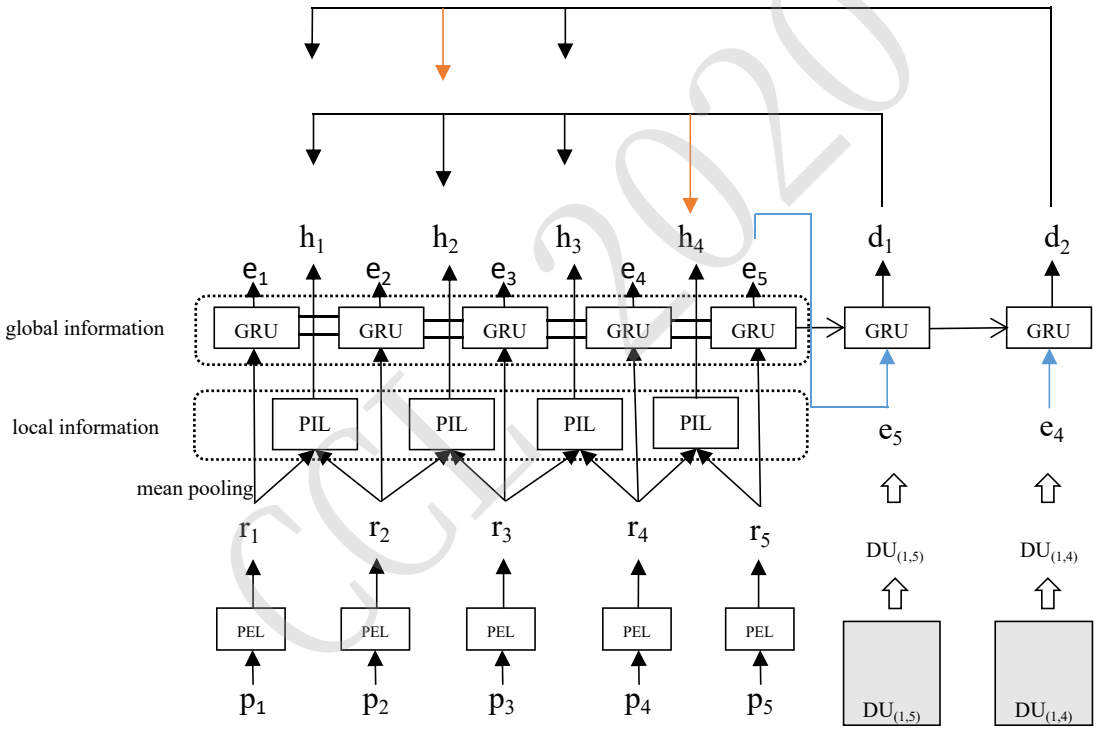


图 2. PNGL模型框架图

3.1 段落编码层

段落编码层（PEL）用来对段落进行编码，获得段落的语义信息。目前大多数的工作大多采用LSTM(Hochreiter and Schmidhuber, 1997)对输入序列进行编码。LSTM虽然具备一定长序列建模能力，但是在处理宏观篇章单元的时候，仍稍显不足。因为宏观篇章单元的最小颗粒度是段落，包含更多的词语，随着词数的增加使得篇章单元内出现更复杂的词间依赖，而LSTM按照时序来处理文本，当相距很远的词语存在依赖关系时，LSTM很难捕获到这种关系。最近，通过注意力机制直接对输入序列进行编码(Vaswani et al., 2017; Xu et al., 2019)可取得不错的效果，其计算公式如式（1）所示。

$$Attention\,(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

在其编码的过程中，序列中的每一个词语都与序列中的其他词语进行匹配计算，因而更容易捕获长距离词语之间的依赖关系，本质上注意力机制是对输入序列进行加权求和，因而比LSTM保留了更多的原始输入的信息。而多头注意力机制允许模型可以在不同的表示子空间中学习到相关的信息，可以使得模型更好的捕获长远距离依赖关系。因此在PNGL模型中，本文采用多头注意力机制进行段落层编码。如式（2）所示。

$$\begin{aligned} MultiHAtt\left(Q,K,V\right) &= Concat\left(head_1,\cdots,head_h\right)W^O \\ head_i &= Attention\left(QW_i^Q, KW_i^K, VW_i^V\right) \end{aligned} \tag{2}$$

参数矩阵$W_i^Q \in \mathrm{R}^{d_m\times d_k}$,$W_i^K \in \mathrm{R}^{d_m\times d_k}$,$W_i^V \in \mathrm{R}^{d_m\times d_v}$,$W^O \in \mathrm{R}^{hd_v\times d_m}$。段落编码层输入词语序列$p=\{x_1,x_2,\cdots,x_l\}$，l是段落中词语的个数，每一个词语$x_i \in R^e$使用其对应的词向量表示。得到段落编码结果$r_i \in R^{m\times d_m}$，如式（3）所示，其中$W_S^Q, W_S^K, W_S^V \in R^{d_m}$是共享的转换矩阵，从而在编码时将段落映射到相同的特征空间。

$$r_i = MultiHAttt(pW_S^Q, pW_S^K, pW_S^V) \tag{3}$$

3.2 段落交互层

段落交互层（PIL）用来捕获相邻两个段落之间的语义联系（局部信息）。一些研究人员通过注意力机制直接对序列之间的交互建模，并且提出了一些交互注意力机制。例如，Guo et al. (2018)提出一种模拟双向阅读的交互注意力机制，他从人类阅读的角度出发，发现人类在判断两个序列之间的关系时往往需要来回阅读这两个序列，尤其是考虑两个序列中联系比较紧密的词之间的语义联系。受交互注意力机制工作的影响，Xu et al. (2019)采用式（1）对序列之间的交互进行建模，并在篇章关系识别任务中取得了不错的效果，因此本文利用多头交互注意力机制获得段落之间交互的语义联系。

对于两个段落$p_1=\{x_1,x_2,\cdots,x_m\}$和$p_2=\{x_1,x_2,\cdots,x_n\}$，使用式（3）得到段落编码$r_1$和$r_2$,然后使用式（4）对段落之间的交互进行建模。

$$\begin{aligned} I_1 &= MultiHAtt\left(r_2W_{i1}^Q, r_1W_{i1}^K, r_1W_{i1}^V\right) \\ I_2 &= MultiHAtt\left(r_1W_{i2}^Q, r_2W_{i2}^K, r_2W_{i2}^V\right) \end{aligned} \tag{4}$$

式（4）首先通过转换矩阵$W_{i1}^Q, W_{i1}^K, W_{i1}^V \in R^{d_m\times d_i}$和$W_{i2}^Q, W_{i2}^K, W_{i2}^V \in R^{d_m\times d_i}$对输入序列做了映射。在多头注意力交互层，通过交换两个序列的query值，每个序列的词语都根据与另一个序列中所有词语的联系进行了重新编码，从而得到段落p_1和p_2彼此相关的向量表示$I_1 \in R^{m\times d_i}$和$I_2 \in R^{n\times d_i}$。最后通过平均池化操作获得包含彼此信息的段落表示$C_1, C_2 \in R^{d_i}$。在包含彼此信息的段落表示C_1，C_2上，通过非线性变换进一步捕获段落之间的交互信息，将变换得到的向量h_1表示段落p_1和p_2之间语义联系的紧密程度,如式（5）所示,其中,$W_h \in R^{d_m\times 3d_i}$是参数矩阵。

$$h_1 = tanh(W_h[C_1, C_2, C_1 - C_2]) \tag{5}$$

3.3 指针网络

序列到序列的模型(Sutskever et al., 2014)提供了输入序列和输出序列长度可以不同的灵活性,但是由于该模型仍然需要固定输出词汇表的大小，而输出词表的大小取决于输入序列的长度，从而限制了需要指向输入序列某个位置的问题的适用性。而指针网络(Vinyals et al., 2015)通过使用注意力作为一个指向机制解决了这个问题。具体说来，对于输入序列$X=\{x_1,x_2,\cdots,x_n\}$，首先经过编码层得到输出$Y=\{y_1,y_2,\cdots,y_n\}$。在解码层的每一个时间步t，输出的状态d_t会和序列Y进行交互，计算注意力，然后通过softmax层获得关于输入序列的概率分布。因此，在PNGL模型中，本文运用指针网络获得关于文章相邻两个段落之间的语义联系（H）的概率分布，进而确定文章的切分位置。

3.3.1 编码层

Chung et al. (2014)的研究表明，GRU(Cho et al., 2014)和LSTM在很多任务上的性能不分伯仲，但是GRU拥有更少的参数，容易收敛，因此在编码层本文使用两层的双向GRU进行编码。以chtb_0282为例，本文将文章$P = \{p_1, p_2, p_3, p_4, p_5\}$通过段落编码层，得到段落编码$R = \{r_1, r_2, r_3, r_4, r_5\}$,然后采用平均池化操作输入到双向GRU中。双向GRU的输出为$E = \{e_1, e_2, e_3, e_4, e_5\}$,其中$e_i = [e_i^f; e_i^b]$。$e_i^f$和$e_i^b$分别是正向和反向的输出。此时$e_i$综合了前面i-1个段落的语义信息，即获得全局信息。而该全局信息隐含了篇章单元之间的结构信息和语义联系，对于最终篇章结构树的构建起着不可忽视的作用。

3.3.2 解码层

在解码层采用的也是一个两层的GRU。以chtb_0282为例，本文将编码层的输出$E = \{e_1, e_2, e_3, e_4, e_5\}$作为Decoder层的输入。在第t步解码时，篇章单元$DU_{(l,r)}$出栈，解码层会综合当前篇章的全局信息e_r和t步之前生成的结构语义信息生成当前状态d_t。d_t和段落交互层的输出$H = \{h_l, h_{l+1}, \cdots, h_{r-1}\}$进行交互，融合全局和局部信息，通过一个softmax层得到关于H的概率分布。如式（6）所，其中$\sigma(.,.)$是融合全局和局部信息的函数，具体为点积运算；α_t为关于H的概率分布。

$$
\begin{aligned}
s_{t,i} &= \sigma(d_t, h_i), i = l \cdots r-1 \\
\alpha_t &= softmax(s_t) = \frac{exp(s_{t,i})}{\sum_{i=l}^{r-1} exp(s_{t,i})}
\end{aligned}
\tag{6}
$$

如果通过softmax层后h_i被分配的概率值越大，表明段落p_i和p_{i+1}之间的语义联系越松散，因此更应该切分开，从而将整个篇章分为两个篇章单元$DU_{(l,i)}$和$DU_{(i+1,r)}$。根据深度优先的原则，每一步解码，段落数量大于2的篇章单元将继续入栈，递归地对篇章单元进行切分，直至栈空，过程如图3所示。

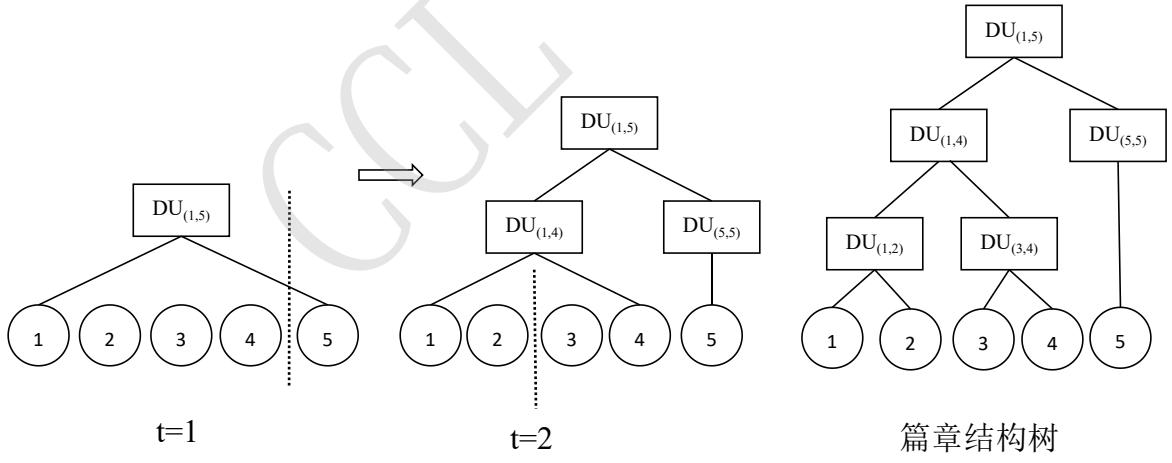


图 3. 解码过程

3.4 损失函数

在PNGL模型中，损失函数本文采用负对数似然函数进行计算，如式（7）所示。$y_{<t}$是在解码层第t步之前已经产生的篇章单元，T是入栈的篇章单元数。为了防止过拟合，本文在指针网络的编码和解码层进行了dropout操作。

$$
L(\theta_s) = -\sum_{i=1}^{batch} \sum_{t=1}^{T} logP_{\theta_s}(y_t | y_{<t}, X) \tag{7}
$$

4 实验

4.1 实验设置

本文在宏观汉语篇章树库（MCDTB）上对模型结构识别的性能进行了评估。MCDTB定义了三大类十五小类篇章关系，并标注了摘要，段落中心句、篇章结构等宏观篇章信息。MCDTB总计有720篇新闻报道的文章，每篇文章的段落数从2到22不等，其段落分布如表1所示。

段落	2	3	4	5	6	7	8	9	10	11	12	> 12
数量	29	122	159	144	91	58	37	33	15	13	14	15

表1.段落分布

本文使用Jiang et al. (2018a)遵循段落分布划分好的数据集进行试验，其中训练集576篇，测试集144篇。为了与Zhou et al. (2019)的实验设置一致，本文将所有的非二叉树都转换为右二叉树。另外，本文遵循Morey et al. (2017)对RST-DT上篇章结构分析模型的评价标准，同样采用内部节点正确率（等价于micro-F1）来衡量模型性能。本文将词向量维度设置为300，采用Word2Vec(Mikolov et al., 2013)进行预训练。在段落编码层和段落交互层转换矩阵映射的维度d_m和d_i都被设置为512;段落编码层多头注意力机制中头数h设置为8，其中$d_k = d_v = d_m/h = 64$;训练过程中batch大小设置为32，dropout率设置为0.5。

4.2 实验结果

本文将文中提出的模型PNGL和基准系统进行了对比，基准系统分为两种：1）只考虑局部信息2）只考虑全局信息，基准系统介绍如下：

LD-CM：性能最好的传统模型(Jiang et al., 2018a)，只考虑局部信息。该模型采用条件随机场，运用较多的手工特征，考虑相邻两个篇章单元能够合并，贪婪的自底向上识别篇章结构，从而构建篇章结构树。

MVM:性能最好的神经网络模型(Zhou et al., 2019)，只考虑局部信息。该模型从词、局部上下文以及话题这三个角度出发，提出了词对相似度机制来衡量相邻两个篇章单元的语义。并采用移进规约的方法每次考虑相邻两个篇章单元能否合并，从左到右识别篇章结构，从而构建篇章结构树。

PN:本文复现了在RST-DT上表现优异的结构识别模型PN(Lin et al., 2019)，只考虑全局信息。该模型是一个指针网络，在编码层使用双向GRU对整个文章进行编码，解码层使用单向GRU进行解码，自顶向下的识别篇章结构，构建篇章结构树。

模型	内部节点正确率（%）
LD-CM	54.71
MVM	56.11
PN	56.25
PNGL	**58.42**

表2.模型在MCDTB上的性能比较

实验结果如表2所示。PNGL模型比仅考虑局部信息的LD-CM模型性能提升了3.71，比仅考虑局部信息的MVM模型（目前在MCDTB上最好的结构识别的模型）性能提升了2.31，比仅考虑全局信息的PN模型性能提升了2.17。宏观篇章结构理论(Van, 1980)指出，文章会有一个总摄全篇的主题，并层层分解，由下层命题展开。这说明段落或篇章单元之间的关系并非很松散，都是在对主题进行分层面的展开叙述。

而LD-CM和MVM都是考虑相邻两个篇章单元联系的紧密程度，但是这两个篇章单元是围绕共同的主题展开的，如果仅仅考虑两个篇章单元之间的联系，模型往往会偏向于将这两个篇章单元合并成更大的篇章单元。而PN模型通过考虑整个篇章单元的语义信息，将篇章单元切分成两个较小的篇章单元。PN模型会对所有可能形成的两个较小篇章单元语义联系的紧密程度进行排序，取语义联系最松散的两个较小篇章单元作为切分结果。但是每个篇章单元往往包含较复杂的段落语义信息，仅仅考虑全局信息，模型很难对两个较小篇章单元之间的语义联系的紧密程度进行正确的排序。

本文的模型PNGL通过改进段落的语义编码，在指针网络编码层学习到更好的全局信息的同时，又考虑相邻两个段落之间语义联系的紧密程度，从而在性能上有所提升，这说明综合考虑全局和局部信息对于识别篇章结构并构建篇章结构树非常有效。

5 实验分析

5.1 全局和局部信息的影响

以往的研究表明(Lin et al., 2019)，采用基于转移的方法进行结构识别，往往对于底层的识别能力比较好，而上层的识别能力比较差。主要原因是每一步的识别都只考虑局部信息，这会将错误传播到后续步骤，导致上层的结构的识别能力较差。

为了研究局部信息和全局信息分别对底层和顶层结构识别的影响，本文在PNGL模型的基础之上去掉段落交互层，即只考虑全局信息，得到模型PNGL(-local)。本文对只考虑局部信息最好的模型MVM以及只考虑全局信息最好的模型PNGL(-local)在最底下两层内部节点正确率和最顶上三层内部节点正确率[0]进行了统计分析，如表3所示。

模型	最底下两层节部结点正确率%	最顶上三层内部节点正确率%
MVM	**46.95**	60.28
PNGL(-local)	42.68	**65.35**

表3.局部和全局信息分别对底层和顶层结构识别的影响

由表3的实验结果可知，相比于只考虑全局信息的模型，MVM在最底下两层节点正确率更高，这说明考虑局部信息的对于底层结构识别有帮助。PNGL（-local）在最上三层的节点正确率要高于MVM，说明相比于考虑局部信息的模型，只考虑全局信息对上层结构识别有帮助。因此本文认为在全局信息的基础上加入局部信息可以增强模型对于底层节点的识别能力。

为了研究在全局信息的基础之上融合局部信息对于结构识别的影响，本文在模型PN的基础之上，加入段落交互层，综合考虑全局和局部信息，得到模型PN(+local)；而PN和PNGL（-local）都是只考虑全局信息的指针网络模型，它们的区别在于PN采用双向GRU对段落进行编码，而PNGL（-local）采用多头注意力机制对段落进行编码。本文统计了内部结点正确率以及最底下两层内部结点正确率，如表4所示。

模型	内部节点正确率（%）	最底下两层内部节点正确率（%）
PN	56.25	46.65
PN(+local)	56.87	47.26
PNGL(-local)	56.57	42.68
PNGL	58.42	48.48

表4.加入局部信息后模型识别性能比较

表4实验结果表明，在加入局部信息之后PNGL和PN（+local）的最底下两层内部节点正确率分别提高了1.01和5.8。PNGL相较于PN（+local），性能有更多的提升，其原因在于PN（+local）是直接使用双向GRU对段落进行编码，而PNGL是使用多头注意力机制对段落进行编码，由于多头注意力机制相较于双向GRU更容易捕获长距离单词之间的依赖关系，能保留更多原始的信息，对段落的编码更有效。那么相邻两个段落的编码输入到段落交互层进行交互，段落交互层就能更好的捕获段落之间的语义联系。通过捕获到更好的局部信息，模型PNGL增强了对底层结构的识别能力，从而从整体上提高了模型的性能。

5.2 模型对长短文识别性能比较

为了比较模型对于长文和短文的识别能力，本文分别统计了长文和短文内部节点正确率，如表5所示。从表中数据可知，模型对短文结构的识别性能较好，而长文结构的识别性能较差。主要原因在于无论采用什么方法构建篇章结构树，都会产生级联错误，而对长文来说，则更加明显。但和只考虑局部信息的模型以及只考虑全局信息的模型相比，本文的模型PNGL综合考虑全局和局部信息，在短文和长文的结构识别的性能都有提升。

[0]由于最顶层的根节点所表示的结构总是固定，因此本文考虑最顶上三层和最底下两层内部结点正确率来表示模型对于顶层和底层结构识别的性能好坏。

模型	内部节点正确率%	
	≤ 6	> 6
LD-CM	65.24	42.23
MVM	65.81	44.59
PN	66.38	44.26
PNGL	**68.95**	**46.62**

表5.模型对长短文结构识别性能比较

5.3 不同模型结构识别的比较

图4从左到右展示了只考虑局部信息、只考虑全局信息以及考虑全局和局部信息的模型对chtb_0756（文章内容及标准结构树见附录A）的预测结果。MVM应用栈和队列，采用移进规约的方法，考虑栈顶的篇章单元和队首的段落能否合并成一个更大的篇章单元，如果可以合并则采取规约操作，否则采取移进操作。由于MVM只考虑局部信息，在从左到右进行结构识别的时候，未能识别出来相邻两个段落之间是否要合并成一个大的篇章单元，因此采用了一系列的移进操作，当队列中为空之后，又采取一系列的规约操作，最终形成如图所示的结构树。

PNGL（-local）采用栈数据结构，通过自顶向下的方法递归确定文章的切分位置，从而形成结构树。PNGL(-local)首先会对$DU_{(1,1)}$和$DU_{(2,5)}$、$DU_{(1,2)}$和$DU_{(3,5)}$、$DU_{(1,3)}$和$DU_{(4,5)}$、$DU_{(1,4)}$和$DU_{(5,5)}$这四个语义联系的紧密程度进行排序，确定$DU_{(1,4)}$和$DU_{(5,5)}$之间的语义联系最松散，然后递归地对$DU_{(1,4)}$进行以上过程，确定$DU_{(1,2)}$和$DU_{(3,4)}$之间的语义联系最松散，最终形成如图所示的结构树。但由于篇章单元中往往有多个段落，包含的语义信息比较复杂，如果只考虑全局信息，会使得模型很难对相邻篇章单元之间的紧密程度进行正确排序。而本文的模型PNGL通过加入相邻两个段落之间的语义联系（局部信息），考虑到了篇章单元边界的信息，从而提升了模型结构识别的能力。

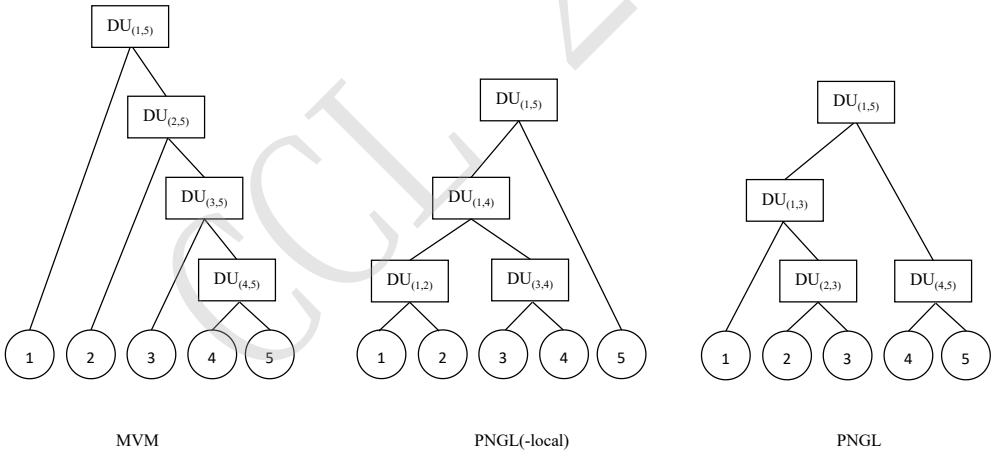


图 4. 不同模型构建的文章chtb_0756的篇章结构树

6 总结

本文针对宏观汉语篇章结构识别任务，提出了一种融合全局和局部信息的指针网络模型PNGL用于自顶向下的识别篇章结构，构建篇章结构树。其中，段落编码层采用多头注意力机制，可以有效地捕获词语之间的长距离依赖；段落交互层通过多头注意力交互机制捕获段落和段落之间的语义联系，即局部信息；指针网络的编码层用来捕获全局信息，解码层会融合全局和局部信息进行解码，自顶向下的识别篇章结构，构建篇章结构树。在MCDTB实验结果表明，本文的模型PNGL比传统机器学习的方法LD-CM性能提高了3.71%，比目前最好的模型MVM性能提高了2.31%，证明了融合全局和局部信息在篇章结构识别任务中的有效性。由于模型识别短文的性能比较好，因此在下一步工作中将融入话题分割的思想，尝试将长文划分成短文本，从而提高长文的结构识别的性能。

参考文献

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2007. RST Discourse Treebank.Current and new directions in discourse and dialogue. pages 85–112.

Kyunghyun Cho, Bart Van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1724–1734.

Xiaomin Chu, Xuefeng Xi, Feng Jiang, Sheng Xu, Qiaoming Zhu, and Guodong Zhou. 2020. Macro discourse structure representation schema and corpus construction. *Journal of Software*, 31(2):321–343.

Junyoung Chung, Caglar Gulcehre, Kyung Hyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *Computer Research Repository*.

Arman Cohan and Nazli Goharian. 2015. Scientific article summarization using citation-context and article's discourse structure. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 390–400.

Fengyu Guo, Ruifang He, Di Jin, Jianwu Dang, Longbiao Wang, and Xiangang Li. 2018. Implicit discourse relation recognition using neural tensor network with interactive attention and sparse learning. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 547–558.

Hugo Hernault, Helmut Prendinger, David A. Duverle, and Mitsuru Ishizuka. 2010. HILDA: A discourse parser using support vector machine classification. *Dialogue and Discourse*, 1(3):1–33.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, pages 1735–1780.

Yangfeng Ji and Jacob Eisenstein. 2014. Representation learning for text-level discourse parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13–24.

Feng Jiang, Peifeng Li, Xiaomin Chu, Qiaoming Zhu, and Guodong Zhou. 2018a. Recognizing macro Chinese discourse structure on label degeneracy combination model. In *Proceedings of the 7th CCF International Conference on Natural Language Processing and Chinese Computing*, pages 92–104.

Feng Jiang, Sheng Xu, Xiaomin Chu, Peifeng Li, Qiaoming Zhu, and Guodong Zhou. 2018b. MCDTB: A macro-level Chinese discourse treebank. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3493–3504.

Shafiq Joty, Giuseppe Carenini, Raymond Ng, and Yashar Mehdad. 2013. Combining intra- and multi-sentential rhetorical parsing for document-level discourse analysis. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 486–496.

Maria Liakata, Simon Dobnik, Shyamasree Saha, Colin Batchelor, and Dietrich Rebholz-Schuhmann. 2013. A discourse-driven content model for summarising scientific articles evaluated in a complex question answering task. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 747–757.

Xiang Lin, Shafiq Joty, Prathyusha Jwalapuram, and M Saiful Bari. 2019. A unified linear-time framework for sentence-level discourse parsing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4190–4200.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 27th Annual Conference on Neural Information Processing Systems*, pages 3111–3119.

Mathieu Morey, Philippe Muller, and Nicholas Asher. 2017. How much progress have we made on RST discourse parsing? A replication study of recent results on the RST-DT. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1319–1324.

Caroline Sporleder and Alex Lascarides. 2004. Combining hierarchical clustering and machine learning to predict high-level discourse structure. In *Proceedings of the 20th International Conference on Computational Linguistics*, pages 43–49.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 28th Annual Conference on Neural Information Processing Systems*, pages 3104–3112.

Dijk T A Van. 1980. Macrostructures : An interdisciplinary study of global structures in discourse, interaction, and cognition. *hillsdale, n.j.*

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31th Annual Conference on Neural Information Processing Systems*, pages 5998–6008.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *Proceedings of the 29th Annual Conference on Neural Information Processing Systems*, pages 2692–2700.

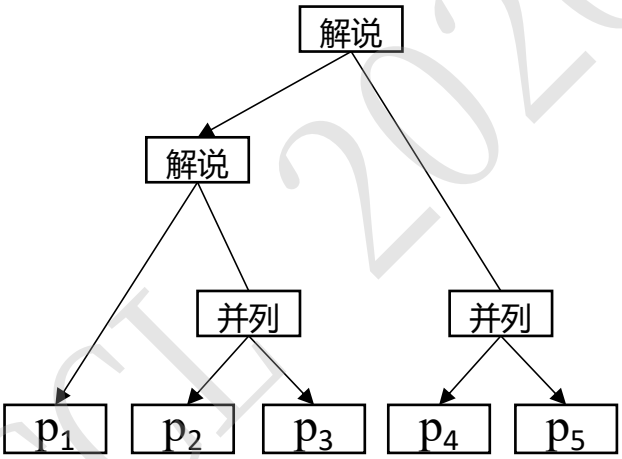
Sheng Xu, Tishuang Wang, Peifeng Li, and Qiaoming Zhu. 2019. Multi-layer attention network based Chinese implicit discourse relation recognition. *Journal of Chinese Information Processing*, 33(8):12–19.

Yi Zhou, Xiaomin Chu, Peifeng Li, and Qiaoming Zhu. 2019. Constructing Chinese macro discourse tree via multiple views and word pair similarity. In *Proceedings of the 8th CCF International Conference on Natural Language Processing and Chinese Computing*, pages 773–786.

A chtb_0756文章内容及标准结构树

p_1:由于当天公布的一份报告表明美国消费者对经济前景具有信心，纽约股市２９日全面走高。道－琼斯３０种工业股票平均价格指数上升９４．２３点，收于９３２０．９８点，增幅达百分之一。
p_2:道－琼斯指数在过去８个交易日里连续上升。到目前为止，该指数已比今年初上涨了百分之十七点九，比１１月２３日创造的最高记录９３７４．２７点也只有５３点之遥。
p_3与此同时，标准普尔５００种股票指数和以技术股为主的纳斯达克指数２９日均创下了最高纪录。标准普尔指数上升了６．３２点，收于１２４１．８１点。纳斯达克指数则上升了１．４７点，收于２１８１．７７点。此外，纽约证券交易所和美国证券交易所指数以及以小公司为主的罗斯２０００股票指数都告上升。
p_4:在当日的交易中，上涨股票以零售业为主。而前几个交易日紧俏的因特网股则因获利回吐而下跌。
p_5:当天，纽约证交所的上升股与下跌股之比为７比５，成交额从前一交易日的５．２６亿股微升到５．８２亿股。（完）

纽约股市全面上涨（chtb_0756）



宏观篇章结构树（chtb_0756）

基于图神经网络的汉语依存分析和语义组合计算联合模型

汪凯，刘明童，陈圆梦，张玉洁†，徐金安，陈钰枫
北京交通大学 计算机与信息技术学院，北京 100044
† 通讯作者，E-mail:yjzhang@bjtu.edu.cn

摘要

组合原则表明句子的语义由其构成成分的语义按照一定规则组合而成，由此基于句法结构的语义组合计算一直是一个重要的探索方向，其中采用树结构的组合计算方法最具有代表性(Tai et al., 2015)。但是该方法难以应用于大规模数据处理，主要问题是其语义组合的顺序依赖于具体树的结构，无法实现并行处理。本文提出一种基于图的依存句法分析和语义组合计算的联合框架，并借助复述识别任务训练语义组合模型和句法分析模型。一方面图模型可以在训练和预测阶段采用并行处理，极大缩短计算时间；另一方面联合句法分析的语义组合框架不必依赖外部句法分析器，同时两个任务的联合学习可使语义表示同时学习句法结构和语义的上下文信息。我们在公开汉语复述识别数据集LCQMC(Liu et al., 2018) 上进行评测，实验结果显示准确率接近树结构组合方法，达到79.54%，而预测速度提升高达30倍。

关键词： 句法分析；语义组合；图神经网络；复述识别

Joint Learning Chinese Dependency Parsing and Semantic Composition based on Graph Neural Network

Kai Wang, Mingtong Liu, Yuanmeng Chen, Yujie Zhang†
Jinan Xu, Yufeng Chen
School of Computer and Information Technology, Beijing Jiaotong University
Beijing 10004
†Corresponding Author, E-mail:yjzhang@bjtu.edu.cn

Abstract

The semantics of a sentence is composed of the meaning of its constituent components and the combination method. Therefore, syntax-based semantic composition has always been an important research direction in NLP. The semantic composition method using tree structure has became the most representative method(Tai et al., 2015). However, such methods are difficult to be applied to large-scale data. The main problem is that the order of its semantic composition depends on the structure of the specific tree, and parallel computation cannot be supported. In this paper, we present a joint framework for graph-based dependency parsing and semantic composition. The model does not need to rely on an external syntax parser for providing structural information, and the semantic composition method based on graph neural network can support parallel computation, which greatly reduces the computation time. Moreover, the joint learning of two tasks enables the model to learn the syntactic structure and semantic contextual information. Experimental results on LCQMC(Liu et al., 2018) dataset show that the

国家自然科学基金（61876198,61976015,61976016）资助

accuracy is close to the tree-based semantics composition method, reaching 79.54%, and the prediction speed is increased by up to 30 times.

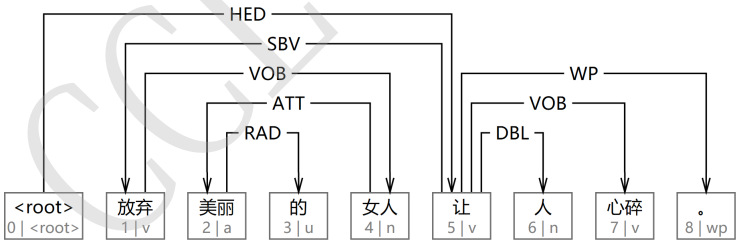
Keywords: Dependency Parsing , Semantic Composition , Graph Neural Network , Paraphrase Identification

1 引言

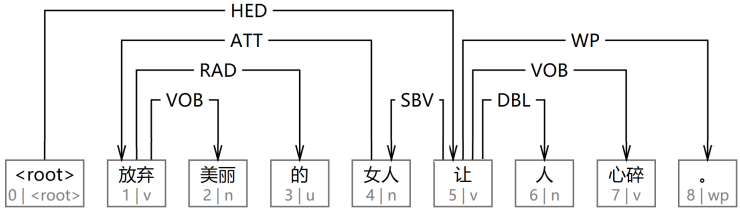
深度神经网络技术为自然语言处理发展带来崭新建模方式和性能上的巨大提升，成为主流的研究方法，其中语义表示是研究热点之一。已有研究表明有效的语义组合计算模型，如LSTM，CNN，Tree-LSTM等神经网络模型，可以提升自然语言处理应用的性能，如：机器翻译(Callison-Burch et al., 2006)、情感分析(Tai et al., 2015)、复述识别(Fan et al., 2018)、自然语言推理(Mou et al., 2016)等。

基于序列化结构的语义组合计算方法简单有效，被广泛采用(Mueller and Thyagarajan, 2016)，但是，这种方法没有考虑句法结构信息，难以捕获词序完全相同句法结构不同的句子之间的差异。比如句子“放弃美丽的女人让人心碎。”，可以有两种句法结构，如图1所示。在图1(a)的句法结构中，“美丽”作为形容词修饰“女人”，“放弃”的对象是“美丽的女人”；在图1(b)的句法结构中，“美丽”作为名词，“放弃”的对象是“美丽”。由此可见，句法结构决定句子语义，句子的语义表示应该考虑其句法结构。

随后，研究人员开始关注基于句法结构的语义组合计算方法(Tai et al., 2015; Chen et al., 2017; Mou et al., 2016)。组合原则表明句子的语义由其构成成分的语义按照一定规则组合而成，由此根据句法结构进行语义组合计算一直是一个重要的探索方向。在基于句法结构的语义组合计算方法中，采用树结构的组合计算方法最具有代表性，其中以Tai et al. (2015)提出的Tree-LSTM最具有代表性。这些方法在给定的一棵句法树上，从叶子节点开始，语义信息自底向上传递，最终在树的根节点获得句子的语义表示。树结构的语义组合方法虽然建模了单词在句法结构上的语义修饰关系，但和序列化方法相比，模型受句法树规定的语义组合顺序的限制，无法并行计算以支持批处理，另外由于需要额外的句法分析器，模型处理繁琐且计算效率低，难以大规模应用到自然语言处理各项任务中。



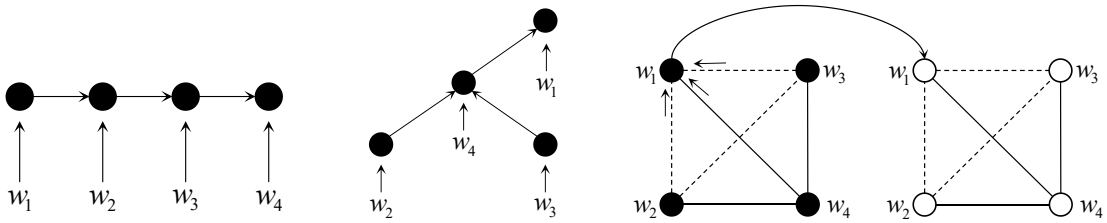
(a)



(b)

图 1: 相同句子的不同依存句法结构

为了解决上述问题，本文主要针对基于句法结构的语义组合计算方法展开研究，提出一种基于图的依存句法分析和利用图神经网络语义组合计算的联合模型。考虑依存结构描述了反映



(a) 序列化的语义组合　(b) 树结构的语义组合　(c) 图结构的语义组合

图 2: 不同的语义组合计算方法

单词间语义修饰关系的依存关系，本文采用依存结构信息指导语义组合计算。我们设计基于图的依存句法分析方法，用于生成带有概率的依存关系结构图，包含所有单词对间的有向依存弧的概率。然后，我们使用依存关系概率作为图神经网络中信息传递的权值，一方面使得语义组合计算可以按照单词间修饰关系概率结合所有单词的语义；另一方面，基于图神经网络的语义组合计算可以实现并行处理，支持训练和预测阶段的批处理，提升计算效率。本文模型与已有方法对比示意图如图2所示。其中，图2(a)是基于序列化结构的组合方法，图2(b)是基于树结构的语义组合方法，图2(c)是本文提出的基于图结构的语义组合方法。我们以复述识别作为语义组合计算的目标任务，在公开汉语复述识别数据集LCQMC(Liu et al., 2018) 上的实验结果表明，本文提出的模型优于已有序列结构的语义组合计算方法，可以有效改进复述识别性能。同时，本文提出的语义组合方式支持批处理操作，在预测阶段速度是Tree-LSTM的30倍，同时能保持较高的精度。

本文的主要贡献如下：(1)提出一种基于图的依存句法分析和语义组合计算的联合框架；(2)提出一种基于图神经网络的语义组合方法，可以实现并行处理，支持训练和预测阶段的批处理，提升计算效率。

2 相关工作

语义组合计算方法主要分为基于序列化的语义组合计算方法和基于句法结构的语义组合计算方法。在序列化的语义组合计算中，如图2(a)所示，模型从左到右依次读入单词，如LSTM,RNN等(Tang et al., 2016; Mueller and Thyagarajan, 2016)。这些方法的优点是可以表示任意长度句子的上下文信息，其语义信息从左向右积累，最终将最后时刻的隐状态向量视为整个句子的语义表示。Kim (2014)利用卷积神经网络获得句子的语义表示，具体做法是使用卷积核在输入句上从左到右滑动，每次滑动捕捉句子局部区域的特征，使得CNN更能捕获n-gram特征，最后通过最大池化获得句子的语义表示。基于序列化方法为了追求运算效率，直接对句子的文本序列进行语义组合计算，未对结构信息加以利用，难以对结构不同带来的语义差异加以区分。

近年来有许多工作(Socher et al., 2012; Li et al., 2015; Tai et al., 2015)试图引入句法结构进行语义组合计算，并在情感分类(Tai et al., 2015)，自然语言推理(Bowman et al., 2016; Mou et al., 2016)等任务中验证了比序列化模型更好的性能。Tai et al. (2015)使用了树结构进行语义组合计算，从树的叶子节点开始将语义信息从底向上传递，最后在树的根节点获得句子表示。Chen et al. (2017) 设计了增强树结构表示，利用短语结构树进行语义组合计算。但是这些方法受自底向上的组合顺序的限制无法实现并行计算，难以支持训练和预测阶段的批处理，导致计算时间过长难以满足实际需求。Mou et al. (2016)提出了树结构的卷积操作，通过对每个节点的孩子节点进行卷积操作获得该节点的语义表示，最后对所有节点使用最大池化获得句子语义表示，并在自然语言推理任务上验证其有效性。该方法虽然可以实现并行计算，但只计算了直接孩子的语义信息，没有考虑子孙节点的语义。

本文提出的基于图的依存分析模型和图神经网络语义组合计算联合框架，使用带有概率的依存关系结构图进行语义组合计算，一方面可以实现并行处理，另一方面可以考虑所有节点的语义信息。

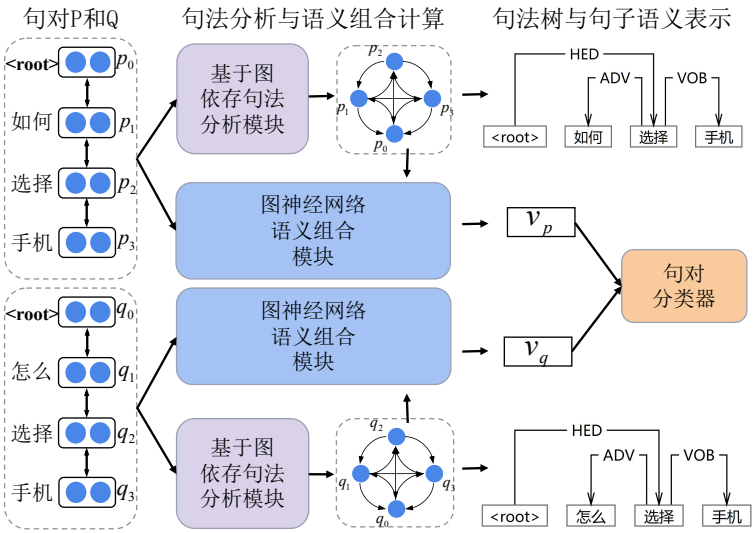


图 3: 本文提出的联合模型整体框架

3 基于图神经网络的依存分析和语义组合计算联合模型

我们采用依存句法树作为句法结构指导语义组合计算，提出了基于图的依存句法分析和语义组合计算的联合框架，模型架构如图3所示。模型接收句对$P = \{p_1, \ldots, p_N\}$和$Q = \{q_1, \ldots, q_M\}$。首先经依存句法分析分别得到带有概率的依存关系结构图，并从中得到依存树，然后经语义组合计算利用该图得到句子的语义表示，并送入句对分类器进行判断。在模型训练阶段，我们联合依存分析和复述识别任务两个目标共同学习模型参数。

3.1 依存句法分析

本文采用基于图的依存句法分析方法(Dozat and Manning, 2017)，该方法可以考虑全局信息进行依存分析决策，最近研究显示该方法在性能上超过了基于转移的依存分析方法(Ji et al., 2019)。下面，我们以句子$P = \{p_0, p_1, \ldots, p_N\}$为例，详细介绍依存句法分析模块。按照通常做法，我们在每个句子的开头加入根节点的标识“<root>”作为p_0。

首先将输入的单词序列转化为数值向量表示，我们采用预训练词向量、随机初始化词向量和词性标签向量三部分构成输入词向量。我们用$e(p_i) \in \mathbb{R}^d$表示预训练词向量，$e'(p_i) \in \mathbb{R}^d$表示随机初始化词向量，$e(pos_i) \in \mathbb{R}^{d_{pos}}$表示词性标签向量，$d_{pos}$为词性的嵌入维度，三部分的表示在训练中被更新。最终，每个单词的表示由公式1计算得出，其中$\oplus$为拼接操作。

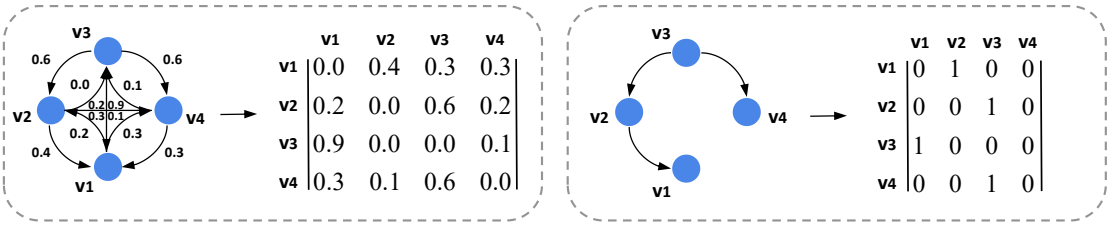
$$x_i = (e(p_i) + e'(p_i)) \oplus e(pos_i) \tag{1}$$

为了捕捉句子长距离的上下文信息，我们采用深层双向LSTM(BiLSTM)学习句子中的词表示。其中，第i时刻(对应第i个单词)的隐藏状态表示如公式2所示。

$$h_i = BiLSTM(x_i, \overleftarrow{h}_{i+1}, \overrightarrow{h}_{i-1}, \theta) \tag{2}$$

其中，$\overleftarrow{h}_i$和$\overrightarrow{h}_i$是在时刻i前向和逆向LSTM的隐藏表示；θ为BiLSTM中的参数。

本文使用图$G = (V, E)$表示句子P的依存关系图，其中$V = \{p_0, p_1, \ldots, p_N\}$是句子中单词节点集合，$E$是依存关系边集合。序列$P$中每个词与图上的节点对应，使用$p_j \rightarrow p_i$表示核心词(head)$p_j$与依存词(dep)$p_i$之间存在依存关系。由于句子中任意两个单词之间存在两种依存关系$p_j \rightarrow p_i$和$p_i \rightarrow p_j$，需要为每个单词计算其作为核心词或依存词的向量表示。为此，我们为每个单词设置两个向量表示，一个是单词作为依存词的表示，另一个是单词作为核心词的表示。对于这两种表示的计算，我们分别采用多层感知器对BiLSTM的输出h_i进行计算，如公式3和4所示(Dozat and Manning, 2017)。在此基础上，可以为所有单词对中的两种依存关系计算得分，具体的我们采用双仿射注意力机制进行计算，计算过程如公式5所示。其中，s_{ij}表



(a) 软结构　　(b) 硬结构

图 4: 基于图神经网络使用结构信息的两种方式

示$p_j \rightarrow p_i$的得分，得分越大表示构成$p_j \rightarrow p_i$的可能性越大。

$$r_i^{dep} = MLP^{(dep)}(h_i) \tag{3}$$

$$r_j^{head} = MLP^{(head)}(h_j) \tag{4}$$

$$s_{ij} = r_i^{dep} U r_j^{head} + r_j^{head} u \tag{5}$$

其中，U表示权重矩阵，u表示偏置项。

$s_i = [s_{i0}, \ldots, s_{ij}, \ldots, s_{iN}](j \in \{0, 1, \ldots, N\})$，$s_{ij}$是$p_j \rightarrow p_i$依存关系的得分，其中$s_{i0}$用于衡量第$i$个单词成为根$ROOT$的可能性。随后采用公式6进行归一化操作得到概率分布α_i，由$\alpha_i (i \in \{0, 1, \ldots, N\})$构成依存关系概率矩阵$\alpha$。最后采用最大生成树算法解码获得句子的依存结构。在训练阶段，我们使用交叉熵作为损失函数，如公式7所示。

$$\alpha_i = softmax(s_i) \tag{6}$$

$$\mathcal{L}_0 = -\sum_{k=1}^{N_s} \sum_{i=1}^{N_k} \beta_i^k log(\alpha_i^k) \tag{7}$$

目标函数$\mathcal{L}_0$表示交叉熵损失，N_s表示一个批次中句对个数，N_k表示第k个P句的单词个数，β_i^k是第k个P句中第i个单词真实核心词的独热码表示。参照(Dozat and Manning, 2017)，我们设计预测依存关系类型的预测模型，此部分构成的损失函数为$\mathcal{L}_1$。我们将预测结构的损失$\mathcal{L}_0$与预测依存关系类型的损失$\mathcal{L}_1$相加构成$\mathcal{L}_p$。同理，对于句子Q我们可以得到相应的目标函数$\mathcal{L}_q$。最后，我们将$\mathcal{L}_p$与$\mathcal{L}_q$相加作为依存分析模型的损失函数$\mathcal{L}_{dep}$。

3.2 语义组合计算

我们提出一种基于图神经网络的语义组合计算方法，通过利用上一节的依存分析提供的依存关系的概率矩阵α进行语义组合计算，以支持批处理大幅提升计算速度。根据句法分析，α_{ij}表示单词p_j是p_i核心词的概率，我们将依存句法分析学习到的权重α_{ij}视为依存关系$p_j \rightarrow p_i$的语义相关性权重，同时将h_i视为图上节点p_i的语义表示，然后在此图基础上进行语义组合计算。

本文采用图信息传递机制(Veličković et al., 2018; Huang et al., 2019) 建模图中每个节点的语义信息，首先节点p_i从邻节点收集语义信息，我们设计了两种收集的方式。第一种收集方式利用依存关系概率矩阵α直接作为权重结合邻节点的语义表示，计算公式如8所示，我们称这种方式为软结构信息，示意图如图4(a)所示。第二种收集方式依据依存结构结合具有依存关系节点的语义信息(Huang et al., 2019; Yao et al., 2018)，首先修改依存关系概率矩阵α，对于单词p_i设置概率最大的核心词的概率为1，其他单词的概率设置为0，具体修改方式如公式9所示，然后再按公式8进行语义信息的收集，我们称这种方式为硬结构信息，示意图如图4(b)所示。得到邻节点语义信息M_i后，根据公式10更新当前节点的语义表示。

$$M_i = \sum_{j=1}^{N} \alpha_{ij} h_j \tag{8}$$

$$\alpha_{ik} = \begin{cases} 1 & ,k = argmax\alpha_i \\ 0 & ,\text{else} \end{cases} \tag{9}$$

$$h_i^{'} = (1 - \eta_{p_i}) LeakyReLU(M_i) + \eta_{p_i} h_i \tag{10}$$

其中，$M_i \in \mathbb{R}^d$是节点p_i从邻节点获得的语义信息，$h_i \in \mathbb{R}^d$表示节点p_i原始的语义表示，$\eta_{p_i} \in \mathbb{R}$是节点$p_i$的语义更新权重，控制应保留$p_i$多少原来的语义信息，$1 - \eta_{p_i}$用于控制节点$p_i$接收到多少邻节点的语义信息。最后，使用平均池化获得句子的语义表示。句子语义表示定义为：

$$v_p = \frac{1}{|N_p|} \sum_{i \in N_p} h_i^{'} \tag{11}$$

其中，N_p是句子P中单词节点下标的集合，$|N_p|$是句子P中单词的个数。v_p即为句子P的语义表示，同理，对于句子Q，我们可以获得其语义表示v_q。

为了检验本文基于图神经网络的语义组合计算方法能更好的学习句子的语义表示，我们联合了复述识别任务。给定句对P和Q，预测两个句子是否具有相同的语义。首先基于语义组合计算模块，为句对中的每个句子生成语义表示v_p和v_q。然后，使用这两个句子的语义表示(v_p和v_q)构造特征向量d (Mou et al., 2016)，如公式12所示。然后将此特征向量d送入句对分类器。

$$d = v_p \oplus v_q \oplus (v_p - v_q) \oplus (v_p \odot v_q) \tag{12}$$

其中，$\odot$表示按元素乘积操作，$\oplus$表示向量拼接操作，$d \in \mathbb{R}^{4d}$是构造的特征向量，句对分类器我们采用多层感知机的方式，如公式13所示。

$$\hat{y} = softmax(MLP^{(clf)}(d)) \tag{13}$$

在训练阶段我们使用交叉熵作为损失函数，定义为：

$$\mathcal{L}_{pair} = -\sum_{i=1}^{N_s} g_i log(\hat{y_i}) \tag{14}$$

其中，N_s为一个批次中句对的个数，g_i表示第i个句对是否为复述，如果为复述关系，$g_i = [1, 0]$，如果为非复述关系，则$g_i = [0, 1]$，$\hat{y_i}$是第i个句对各类别的估计概率，如公式13所示。

3.3 联合学习

本文提出的联合模型涉及到两个任务，依存句法分析和语义组合计算，我们采用复述识别验证语义组合计算。由此，模型需要同时学习和优化多个学习目标。在传统的联合学习中，通常对各个任务的损失进行线性加权求和，如公式15，该方法权重较难设定。为了解决多目标联合学习问题，我们采用Kendall et al. (2018) 设计的自学习多目标权重方法。该方法根据噪声方差作为模型收敛程度的评估，进行比重调整。其目标函数设计如公式16。

$$\mathcal{L} = (1 - w)\mathcal{L}_{pair} + w\mathcal{L}_{dep} \tag{15}$$

$$\mathcal{L} = \frac{1}{2\sigma_1^2}\mathcal{L}_{pair} + log\sigma_1^2 + \frac{1}{2\sigma_2^2} w\mathcal{L}_{dep} + log\sigma_2^2 \tag{16}$$

其中$\sigma_1, \sigma_2 \in \mathbb{R}$为学习的参数，跟随训练过程被更新，$\mathcal{L}_{dep}$为依存分析的损失函数，$\mathcal{L}_{pair}$为复述识别的损失函数。

4 实验

4.1 数据集介绍和超参数设置

本文使用公开汉语复述识别数据集LCQMC (Liu et al., 2018)作为实验数据。我们采用高精度的哈工大语言技术平台ltp3.4.0[-1](Che et al., 2010)获取分词、词性和依存句法标注，我们将依存句法标注视为ground truth。表1给出了LCQMC数据集的统计信息。

数据集	划分	句对数	正例数	负例数	词数
LCQMC	训练集	238,766	138,574	100,192	3279k
	开发集	8,802	4,402	4,400	138k
	测试集	12,500	6,250	6,250	152k

表 1: 实验数据集

实验中采用预训练的Word2Vec 词向量(Mikolov et al., 2013)，预训练词向量为200维。词性标签向量设置100维，设置所有LSTM结构的隐藏层为400维，层数为3。对与$MLP^{(dep)}$和$MLP^{(head)}$ 设置层数都为1层隐藏层维度分别为100和500，采用leakyrelu激活函数，α设置为0.1。对于$MLP^{(clf)}$设置层数为2，隐藏层维度为800和400，采用相同的激活函数。我们采用Adam (Kingma and Ba,)优化算法，设置初始学习率大小为2e-3，为β_1为0.9，β_2为0.9。在每一轮迭代中，学习率以0.95的频率衰减。训练batch的大小为128。为了防止过拟合，我们使用了dropout。设置词向量输入层的drop率为0.33，leakyrelu层输出层的drop率为0.33。与已有工作一致，我们采用无标记依存正确率UAS和带标记依存正确率LAS作为依存分析评价指标，采用Accuracy和融合Precision和Recall的综合指标F1 值作为复述识别的评价指标。

4.2 基于自学习多目标权重的实验结果

如果按照公式15计算损失函数，为了找到合理的w需要多次实验，实验结果如表2所示。表2显示了不同权重w对依存分析和复述识别任务性能的影响结果。当w较小时，复述识别性能较好，但是依存分析精度较低；当w较大时，依存分析精度较好但是复述识别性能较低。当w设置为0.9时，依存分析的结果达到最好，带标记正确率达到94.37%，但复述识别的Accuracy只有73.37%。当w设置为0.5时，能共同得到较好的性能，复述识别Accuracy为76.31%，依存分析LAS为93.99%。

如果按照公式16，采用Srivastava et al. (2014)设计的多目标损失函数，复述识别Accuracy达到76.77%，依存分析LAS 92.70 %，与公式15中$w = 0.5$时的最好结果相比，其复述识别的准确率提高0.46个点，显示该方法优于线性加权的损失函数。随后实验中我们采用Kendall et al. (2018) 设计的多目标函数方法。

4.3 语义组合次数实验结果

3.2节介绍了每个节点结合邻近节点语义信息更新自身语义表示的组合计算方法，使得每个节点包含了直接核心词的语义信息。如果在此基础上再进行一次语义组合计算，将使每个节点获得间接核心词的语义信息。为了分析语义组合次数的影响，我们分别进行了基于0次、1次、2次和3次语义组合计算的评测，实验结果如表3所示。n=0表示没有利用结构信息，n=1,2,3表示以不同语义组合计算次数利用结构信息。与n=0相比，n=1的模型在测试集上，复述识别在F_1和Accuracy分别提高了1.96和1.97个点，说明句法结构指导语义组合计算上的有效性。

与n=2,3相比，n=1的模型在复述识别任务上均优于n=2,3的模型。实验结果表明继续增加组合次数并没有提升效果，同时，随着组合次数的增加，模型的复杂度也会增加，随后实验中我们选择一次语义组合计算。另外我们注意到联合模型并未给依存分析带来性能上的提升，一方面由于本文的重点放在语义组合计算上，还没有找到同时提升依存分析精度的有效联合方法；另一方面本文使用的依存结构标注并非人工标注，我们分析存在一定错误难以给出依存分析模型的正确评测结果。

[-1]http://ltp.ai/download.html

目标函数权重w		复述识别	依存句法
复述识别	依存句法	Acc[%]	LAS[%]
0.975	0.025	74.97	86.37
0.95	0.05	75.68	89.09
0.9	0.1	75.61	90.75
0.85	0.15	76.68	91.61
0.8	0.2	76.47	92.33
0.7	0.3	76.75	93.31
0.5	0.5	76.31	93.99
0.1	0.9	73.37	94.37
Kendall et al. (2018)		**76.77**	**92.70**

表 2: 不同w下，联合模型在开发集中两个任务上的性能

组合次数	开发集				测试集			
	依存分析		复述识别		依存分析		复述识别	
	UAS	LAS	F_1	Acc	UAS	LAS	F_1	Acc
n=0	93.77	92.67	73.93	74.07	**95.36**	**94.25**	79.88	77.57
n=1	**93.92**	**92.70**	**77.23**	**76.77**	95.32	94.19	**81.84**	**79.54**
n=2	93.82	92.60	76.94	76.02	95.27	94.16	80.76	78.24
n=3	93.77	92.57	76.74	75.47	95.23	94.08	79.92	77.00

表 3: 语义组合次数在不同任务上的性能，n=0表示没有利用句法结构信息

4.4 模型对比实验

我们与基于序列化和树结构的5种语义组合计算方法进行比较，对比模型分为以下几类：

Baseline: 上一节中n=0的模型，即包含序列信息无结构信息。

MeanVector: 将词表示的平均池化作为句子的语义表示(Blacoe and Lapata, 2012)，其中词表示的计算方法如公式1，该方式无序列信息也无句法结构信息。

CNN: 基于卷积神经网络的语义组合计算方法Kim (2014)和Liu et al. (2018)，该方式包含序列信息无结构信息。

BiLSTM: 使用前向LSTM和后向LSTM最后时刻的隐状态向量拼接作为句子表示(Mueller and Thyagarajan, 2016; Tomar et al., 2017; Liu et al., 2018)，该方式包含序列信息无结构信息。

TreeLSTM: 使用Tai et al. (2015)提出的Child-Sum Tree-LSTM，利用依存结构树进行语义组合计算，将根节点获得的隐状态向量视为句子的表示，该方式包含结构信息。

在复述识别任务上，我们的模型与5种模型在测试集上的评测结果如表4所示。

从表4的结果可以看出，在无结构信息的4种方法中，我们设计的Baseline取得了最好的结

是否利用结构信息	方法	F_1	*Acc*
否	Baseline	79.88	77.57
否	Mean vectors	78.68	75.08
否	CNN	75.70	72.80
否	BiLSTM	78.92	76.10
是	Tree-LSTM	82.02	80.22
是	Our	**81.84**	**79.54**

表 4: 在复述识别上和已有序列化和树结构语义组合方式的比较结果

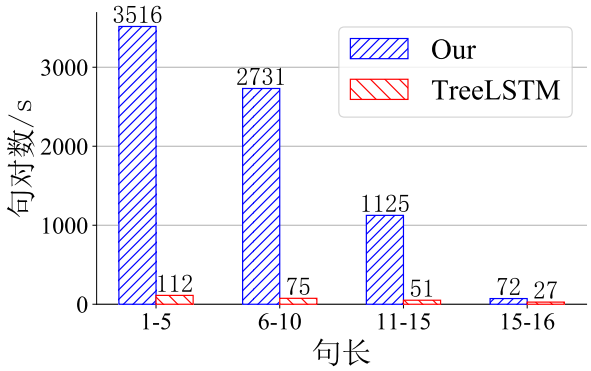


图 5: 句长对模型预测速度的影响

系统	Dev		Test	
	F_1	Acc	F_1	Acc
(1)无结构信息	73.93	74.07	79.88	77.57
(2)硬结构信息	77.01	76.51	81.52	79.15
(3)软结构信息	77.23	76.78	81.84	79.54

表 5: 模型融入依存结构信息有效性分析结果

果，在F_1和Accuracy达到79.88%和77.57%。我们分析原因是在我们的模型中使用了图神经网络，当处理较长句子时，每个节点能从较远节点收集语义信息更新自身表示，能捕捉较长句子的上下文信息。与未利用结构信息的模型中最好精度的Baseline相比，我们的模型在Baseline的基础上加入依存结构优化目标，在F_1和Accuracy上分别提高了1.96和1.97个点。实验结果表明利用句法结构信息进行语义组合计算的有效性。与利用结构信息的Tree-LSTM相比，我们模型在F_1和Accuracy略低于Tree-LSTM 0.18和0.68个点。我们分析原因是Tree-LSTM直接使用了我们视为ground truth的依存标签，而我们的模型使用的是依存标签训练之后依存句法分析模块产生的依存结构，其依存分析精度没有ground truth高。

已有的基于结构的Tree-LSTM每次只能处理一个句对，本文采用基于图的依存分析和图网络语义组合方法，可以实现对多个句对的批处理，从而解决已有模型预测速度慢的问题。我们在不同长度的句子上对模型的预测速度进行评测，并与Tree-LSTM进行对比，评测结果如图5所示。句长在1-5和6-10中我们模型预测速度是Tree-LSTM 的30倍；在句长为11-15中，速度是Tree-LSTM的20倍。这些结果显示本文提出的模型在预测速度上较Tree-LSTM有显著优势。

以上分析结果显示，本文提出的基于依存句法分析和复述识别的联合模型，采用基于图神经网络的语义组合方法，可以有效利用句法结构信息改进语义组合计算，提高复述识别系统的精度和计算速度。

4.5 结构信息有效性分析

我们分析了模型中结构信息对最终复述识别精度的影响，实验结果展示在表5中。从表5的实验结果来看，基于图神经网络引入依存结构信息，有效改进了复述识别的性能。模型（1）没有使用结构信息，仅使用了复述识别的目标函数进行优化，未考虑句子的句法结构，复述识别的Accuracy达到77.57%。模型（2）引入了句法目标训练模型参数，采用了本文提出的硬结构信息，复述识别Accuracy达到79.15%，对比没有结构信息提高了1.58个百分点，这表明引入句法结构对语义组合的有效性。模型（3）采用了软结构信息，Accuracy达到79.54%，进一步改进了复述识别的性能，同时，实验表明本文提出的软结构依存信息在性能上优于硬结构的方法。最终，实验结果表明，本文提出的基于句法结构进行语义组合计算，可以有效学习句子的语义表示，提高了复述识别系统的精度。

ID	句子1(P)	句子2(Q)	Bleu	真实标签	Our	Baseline
A	网站排名推广，主要有哪些推广方式，效果好点的.	目前网上有哪些推广方式	0.19	T	T	F
B	我想知道女生各种发型的名字，加上配图	女生各种发型名称图片	0.24	T	T	F
C	亲爱的韩语怎么说	亲爱的韩语怎么写?	0.71	F	F	T
D	小薏米和大薏米有什么区别	薏仁粉和薏米粉有什么区别	0.61	F	F	T
E	如何编织小狗狗的衣服要方法及图解	如何给小狗做衣服图片	0.24	T	F	F
F	小学二年级语文	小学二年级语文题	0.80	F	T	T

表 6: 一些复杂的例子在本文模型和Baseline上的表现，T表示是复述关系，F表示非复述关系。

4.6 实例分析

我们在LCQMC的测试集中挑选了一些句对进行进一步分析。使用1-gram计算句子P与句子Q的Bleu值，对于复述识别来说，Bleu很高的非复述句对和Bleu很低的复述句对，都是很难的任务，基于浅层信息的方法很难正确识别，需要深层语义理解才可解决。我们特地选择这样的句对评测我们模型的效果，分析结果如表6所示。

示例A-B为Bleu较低的复述句对，因此，容易识别为非复述关系。但是本文模型能够正确识别为复述关系，而Baseline错误的识别为非复述关系。这一对比结果表明本文利用句法结构进行语义组合计算的方法可以捕捉句对之间深层的语义相关性，实现正确判断。

示例C-D为Bleu较高的非复述句对，因此，容易识别为复述关系。但是本文模型能够正确识别为非复述关系，而Baseline错误的识别为复述关系。这一对比结果表明句法结构更易于解决涉及结构复杂表达的语义理解。

示例E-F是Baseline和本文模型都产生错误的情况。E为Bleu较低的复述句对。我们分析预测错误的原因是句子的表达口语化，句法分析很难进行正确分析。F为Bleu较高的非复述句对，其中含有相似的词语“语文”和“语文题”，我们分析预测错误的原因是词的语义表示不能有效的区分二者，这使模型错误的认为它们是复述的关系。对于更复杂的情况，句子的语义表示任然面临很多的问题，例如歧义性以及口语表达。模型可能需要更多的推理信息来区分这些关系并做出正确的决定，例如结合外部知识用于帮助模型更好地理解词汇和短语语义。

5 总结与展望

本文提出一种依存句法分析和语义组合计算的联合框架，设计了基于图的依存句法分析模型和基于图神经网络语义组合计算模型，利用依存分析给出的带有概率的依存关系结构图，实现软结构的语义组合计算方法。一方面图模型中的并行计算能够支持训练和预测阶段的批处理，极大提高计算速度；另一方面两个任务的联合学习可使语义表示同时学习句法结构和语义的上下文信息，投高复述识别精度。

今后，我们考虑结合预训练模型，如ELMO，BERT，以改进模型性能。同时，探索联合模型中提升依存分析精度的方法，从而进一步提升语义组合计算的精度。

参考文献

William Blacoe and Mirella Lapata. 2012. A comparison of vector-based representations for semantic composition. In Jun'ichi Tsujii, James Henderson, and Marius Pasca, editors, *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*, pages 546–556. ACL.

Samuel R. Bowman, Jon Gauthier, Abhinav Rastogi, Raghav Gupta, Christopher D. Manning, and Christopher Potts. 2016. A fast unified model for parsing and sentence understanding. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1466–1477, Berlin, Germany, August. Association for Computational Linguistics.

Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In Robert C. Moore, Jeff A. Bilmes, Jennifer Chu-Carroll, and Mark Sanderson, editors, *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 4-9, 2006, New York, New York, USA*. The Association for Computational Linguistics.

Wanxiang Che, Zhenghua Li, and Ting Liu. 2010. LTP: A Chinese language technology platform. In *Coling 2010: Demonstrations*, pages 13–16, Beijing, China, August. Coling 2010 Organizing Committee.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668, Vancouver, Canada, July. Association for Computational Linguistics.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Miao Fan, Wutao Lin, Yue Feng, Mingming Sun, and Ping Li. 2018. A globalization-semantic matching neural network for paraphrase identification. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, page 2067–2075, New York, NY, USA. Association for Computing Machinery.

Lianzhe Huang, Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2019. Text level graph neural network for text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3444–3450, Hong Kong, China, November. Association for Computational Linguistics.

Tao Ji, Yuanbin Wu, and Man Lan. 2019. Graph-based dependency parsing with graph neural networks. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Papers*, pages 2475–2485. Association for Computational Linguistics.

Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7482–7491. IEEE Computer Society.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October. Association for Computational Linguistics.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization.

Jiwei Li, Thang Luong, Dan Jurafsky, and Eduard Hovy. 2015. When are tree structures necessary for deep learning of representations? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2304–2314, Lisbon, Portugal, September. Association for Computational Linguistics.

Xin Liu, Qingcai Chen, Chong Deng, Huajun Zeng, Jing Chen, Dongfang Li, and Buzhou Tang. 2018. LCQMC:a large-scale Chinese question matching corpus. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1952–1962, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Tomas Mikolov, Greg Corrado, Chen Kai, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of the International Conference on Learning Representations (ICLR 2013)*.

Lili Mou, Rui Men, Ge Li, Yan Xu, Lu Zhang, Rui Yan, and Zhi Jin. 2016. Natural language inference by tree-based convolution and heuristic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 130–136, Berlin, Germany, August. Association for Computational Linguistics.

Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. In Dale Schuurmans and Michael P. Wellman, editors, *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 2786–2792. AAAI Press.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566, Beijing, China, July. Association for Computational Linguistics.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective LSTMs for target-dependent sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3298–3307, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Gaurav Singh Tomar, Thyago Duque, Oscar Täckström, Jakob Uszkoreit, and Dipanjan Das. 2017. Neural paraphrase identification of questions with noisy pretraining. In Manaal Faruqui, Hinrich Schütze, Isabel Trancoso, and Yadollah Yaghoobzadeh, editors, *Proceedings of the First Workshop on Subword and Character Level Models in NLP, Copenhagen, Denmark, September 7, 2017*, pages 142–147. Association for Computational Linguistics.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. *International Conference on Learning Representations*. accepted as poster.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2018. Graph convolutional networks for text classification. *CoRR*, abs/1809.05679.

基于强负采样的词嵌入优化算法

王雨晨
MobTech / 上海
wangych@yoozoo.com

林淼哲
MobTech / 上海
linmzh@yoozoo.com

詹杰凡
MobTech / 上海
zhanjf@yoozoo.com

摘要

word2vec是自然语言处理领域重要的词嵌入算法之一，为了解决随机负采样作为优化目标可能出现的样本贡献消失问题，提出了可以应用在CBOW和Skip-gram框架上的以余弦距离为度量的强负采样方法：HNS-CBOW和HNS-SG。将原随机负采样过程拆解为两个步骤，首先，计算随机负样本与目标词的余弦距离，然后，再使用距离较近的强负样本更新参数。以英文维基百科数据作为实验语料，在公开的语义-语法数据集上对优化算法的效果进行了定量分析，实验表明，优化后的词嵌入质量显著优于原方法。同时，与GloVe等公开发布的预训练词向量相比，可以在更小的语料库上获得更高的准确性。

关键词： 自然语言处理；词嵌入；强负采样

Word Embedding Optimization Based on Hard Negative Sampling

Wang Yuchen
MobTech / Shanghai
wangych@yoozoo.com

Lin Miaozhe
MobTech / Shanghai
linmzh@yoozoo.com

Zhan Jiefan
MobTech / Shanghai
zhanjf@yoozoo.com

Abstract

Word2vec is the important algorithms of word embedding in natural language processing. Traditional training method regards random negative sampling as the objective of optimization, which may cause some samples to lose their contribution in the process of training. To solve this problem, this paper proposes a hard negative sampling method based on cosine distance. This sampling method can be applied to CBOW and Skip-gram, which we call HNS-CBOW and HNS-SG. In this paper, the original random negative sampling process is divided into two steps. First, we calculate the cosine distances between the negatives and the target, and then update parameters by using the hard negatives which close to the target. We use a Wikipedia dump as corpora and conduct experiments on the Semantic-Syntactic Word Relationship test set. After analysis and experimental verification, the quality of word embedding based on hard negative sampling is significantly better than the original word2vec. We also calculate the accuracy of some pre-trained word vectors that have been published, such as GloVe, on the Semantic-Syntactic Word Relationship test set. The HNS models outperform all other baselines, often with smaller corpora.

Keywords: natural language processing , word embedding , hard negative sampling

1 引言

词嵌入技术是自然语言处理（Natural Language Processing，NLP）领域的一项基础工作，它将一个词表达成了语义空间中的实值向量，解决了传统词袋模型的高维、稀疏等问题。把嵌入后的向量当做特征可以应用于一系列NLP下游问题，比如，信息检索(Ye et al., 2016)、文本分类(Miyato et al., 2017; Lilleberg et al., 2015)、句法分析(Bansal M et al., 2014)、命名实体识别(Habibi M et al., 2017; Katharina, 2015)等。

目前广为流行的词嵌入技术是Mikolov(2013; 2013)提出的word2vec算法。该算法以神经网络语言模型为基础，主要针对负采样（Negative Sampling，NS）进行优化，其思想是对于每一个目标词，都会按照词频概率随机地抽取一部分负样本用于参数更新。但随机负采样的一个缺陷是，有的样本在学习多次后会出现贡献消失的情况，学习这类样本不仅会浪费计算资源，也不利于生成高质量的词嵌入。

因此，本文提出了一种基于强负采样（Hard Negative Sampling，HNS）的优化算法。本文的主要贡献有：1)将强采样的思想引入到词嵌入的训练过程中，并详细解释了具体方法；2)对强负采样的有效性及其参数的影响进行了验证。

本文的结构如下：第一部分介绍了词嵌入和强采样的相关工作；第二部分详细解释了通过强负采样改进词嵌入训练的具体方法；第三部分通过实验证明了强负采样的有效性并对其参数进行了分析；最后一部分总结了本研究的改进方向。

2 相关工作

Mikolov(2013; 2013)提出的word2vec框架，将词的独热编码映射成连续的向量表示，在向量空间中，许多语言规律具有了线性平移的性质，比如，“King” - “Man” + “Woman” = “Queen”。不同于传统的神经网络语言模型，word2vec框架剔除了非线性隐藏层，只保留含有输入层、投影层和输出层的3层结构，由于不涉及密集矩阵乘法，该框架实现了更低的计算成本。论文中也指出，word2vec框架相较于传统的神经网络语言模型可以获得更高质量的词嵌入。为了进一步加快训练速度，word2vec在训练时使用了负采样和高频词亚采样的策略。两种方法的采样概率都是词频的函数，词频越高，被采样的概率越大。

word2vec的方法被提出后，许多研究在其基础上进行了拓展和衍生。斯坦福大学的Pennington(2014)提出了GloVe方法，结合全局矩阵分解和局部上下文窗口方法的优点，最终得到一个全局对数双线性模型，该模型生成的词向量质量优于传统的word2vec方法。GloVe也是目前流行的词嵌入算法之一，算法生成预训练词向量已经发布在斯坦福网站上。Wang(2015)修改了word2vec模型使其更关注上下文的相对位置，改进后的模型在词性标注和语法分析任务上的表现均有提升。Ji(2016)通过使用小批量和负样本共享，提高了word2vec算法中各种数据结构的重用性，但该方法主要关注的是训练效率而非词向量质量。

Schroff(2015)提出的FaceNet框架，提供了人脸图像到欧式空间的映射，在嵌入空间中，L2距离的平方直接对应人脸的相似度。论文提出了一个新的三元损失函数，该损失函数同时考虑了在嵌入空间中某点到相同以及不同个体的距离，以期实现最小化相同个体之间距离的同时最大化不同个体之间的距离。论文认为，随机选择正负样本会导致收敛速度变慢，因此文中提出了强正样本和强负样本的概念，用以避免容易满足约束条件的样本过多地进入到训练过程中。论文同时指出，直接优化与任务相关的损失可以提升模型性能。

本文将结合强采样的思想针对词嵌入的负采样过程进行优化。

3 方法

3.1 word2vec的上下文表示方法

word2vec包含两种不同的上下文表示方法：CBOW和Skip-gram，这两种方法都是为了获得目标词和上下文的关系。不同的是，CBOW是给定上下文$w_{t-2}, w_{t-1}, w_{t+1}, w_{t+2}$来预测目标词$w_t$出现的概率，而Skip-gram则是给定目标词$w_t$预测上下文$w_{t-2}, w_{t-1}, w_{t+1}, w_{t+2}$。

在CBOW方法下，对于目标词w和w对应的上下文$Context(w)$，构造条件概率函数$p(w \mid Context(w))$，优化目标是获得参数θ使得公式(1)的概率最大化。

$$\arg\max_{\theta} \prod_{(w,Context(w))\in D} p(w \mid Context(w);\theta) \tag{1}$$

其中，D是语料库中所有目标词w与上下文$Context(w)$组合的集合。
与之相对，Skip-gram的优化目标是公式(2)的概率最大化。

$$\arg\max_{\theta} \prod_{(w,Context(w))\in D} p(Context(w) \mid w;\theta) \tag{2}$$

3.2 负采样的优化目标

负采样是一种能够提高词嵌入训练速度并且有效改善词嵌入质量的方法(Mikolov et al., 2013)。该方法定义$p(D=1 \mid w, Context(w);\theta)$是目标词$w$与上下文$Context(w)$组合出现在语料库中的概率，相对的，$p(D=0 \mid w, Context(w);\theta)$是语料库不包含目标词$w$与上下文$Context(w)$组合的概率(Goldberg and Levy, 2014)。根据定义可知，$p(D=1 \mid w, Context(w);\theta) + p(D=0 \mid w, Context(w);\theta) = 1$。与3.1节相同，此处假设两组概率都是参数$\theta$的函数。

此时，词嵌入的优化目标可以表示为寻找参数θ，使得语料库中存在的目标词w与上下文$Context(w)$组合出现的概率最大化，对其取对数后得到公式(3)。

$$\begin{aligned}
&\arg\max_{\theta} \prod_{(w,Context(w))\in D} p(D=1 \mid w, Context(w);\theta) \\
=&\arg\max_{\theta} \log \prod_{(w,Context(w))\in D} p(D=1 \mid w, Context(w);\theta) \\
=&\arg\max_{\theta} \sum_{(w,Context(w))\in D} \log p(D=1 \mid w, Context(w);\theta)
\end{aligned} \tag{3}$$

同时，若将语料库中不存在的目标词w与上下文$Context(w)$组合定义为负样本集合D'，我们希望最小化这种组合出现的概率，从形式上相当于最大化$p(D=0 \mid w, Context(w);\theta)$，同样取对数后得到公式(4)。

$$\begin{aligned}
&\arg\min_{\theta} \prod_{(w,Context(w))\in D'} p(D=1 \mid w, Context(w);\theta) \\
=&\arg\max_{\theta} \prod_{(w,Context(w))\in D'} p(D=0 \mid w, Context(w);\theta) \\
=&\arg\max_{\theta} \prod_{(w,Context(w))\in D'} (1 - p(D=1 \mid w, Context(w);\theta)) \\
=&\arg\max_{\theta} \sum_{(w,Context(w))\in D'} \log(1 - p(D=1 \mid w, Context(w);\theta))
\end{aligned} \tag{4}$$

在公式(3)和公式(4)中，参数θ相当于目标词w与上下文$Context(w)$的词向量v_w和v_c，$(v_w, v_c) \in \mathbb{R}^d$，$d$是向量长度。使用softmax函数，$p(D=1\Theta w, Context(w);\theta)$可以转化为公式(5)。

$$p(D=1 \mid w, Context(w);\theta) = \frac{1}{1+e^{-v_c \cdot v_w}} \tag{5}$$

因此，负采样的最终优化目标变为公式(6)。

$$\begin{aligned}&\arg\max_{\theta}\sum_{(w,Context(w))\in D}\log\frac{1}{1+e^{-v_c\cdot v_w}}+\sum_{(w,Context(w))\in D'}\log(1-\frac{1}{1+e^{-v_c\cdot v_w}})\\&=\arg\max_{\theta}\sum_{(w,Context(w))\in D}\log\frac{1}{1+e^{-v_c\cdot v_w}}+\sum_{(w,Context(w))\in D'}\log\frac{1}{1+e^{v_c\cdot v_w}}\end{aligned}\tag{6}$$

3.3 强负采样方法

在基于负采样的词嵌入方法中，如何选择负样本是非常重要的环节。不同负样本对于参数更新的贡献是有差别的，在多轮迭代后，某些负样本对于参数更新的贡献可能变得很小，学习这类负样本会减缓收敛的速度，同时也影响了最终的词嵌入质量。

为了保证在快速收敛的同时获得更高质量的词嵌入，在每次负采样时，需要选择能够为参数更新提供更多贡献的词作为负样本。我们提出了一个假设，负样本的贡献大小与其在当前的向量空间中和目标词的距离是相关的，距离目标词越近，负样本能够提供的贡献就越大。

因此，按照与目标词的距离远近进行采样的方法我们称之为强负采样。本文选择余弦距离来衡量向量空间中词与词之间的距离，强负样本的定义见公式(7)。

$$\arg\min_{x_i^n}(1-\frac{f(x_i^t)\cdot f(x_i^n)}{\parallel f(x_i^t)\parallel_2\parallel f(x_i^n)\parallel_2})\tag{7}$$

其中，词嵌入表示为$f(x)$，意思是将词x映射到向量空间中，强负采样时，我们希望选择的负样本x_i^n距离目标词x_i^t足够近。

在进行强负采样时，如果对于每一个目标词，都在整个词典上搜索$\arg\min$，这样做的计算成本是无法负担的，同时，全局搜索强负样本也可能导致模型过早地陷入局部最优。为此，一个显而易见的解决方案是，每次从词典中选择一个小型的批数据（mini-batch）作为候选负样本集合，目标词的强负样本只从这个集合中产生。词被挑选为候选负样本的概率通过公式(8)计算得到。

$$P_n(w)=\frac{U(w)^{\alpha}}{\sum_{u\in D}U(u)^{\alpha}}\tag{8}$$

其中，$P_n(w)$是词典D中第n个词被选为候选负样本的概率，$U(w)$是词w在语料库中出现的次数，α是用于平滑的固定值参数，一般取值是0.75，平滑参数的作用是提高词频较少的词的权重。显然，词频越高，词w越有可能被选为候选负样本。

在生成候选负样本集合时，集合的容量也会影响到训练速度和准确率。容量越小，寻找强负样本所需要的计算成本也越低；而容量越大，则越有可能找到符合全局最优的强负样本，使其对参数更新的贡献更多。在实际计算时，需要考虑平衡以上两点。

3.4 强负采样的复杂度

与随机负采样相比，强负采样算法产生的额外开销包括计算余弦距离以及根据距离对负样本排序。在复杂度方面，假设Q是训练每个词的复杂度，N是目标词上下文的数量，C是上下文距离目标词的最大距离，D是词向量长度，k是随机负采样方法中的负样本数量，$neg1$是候选负样本数量，$neg2$是强负样本数量。CBOW在随机负采样下的时间复杂度是公式(9)，强负采样下的时间复杂度是公式(10)。

$$Q=N\times D+D\times k\tag{9}$$

$$Q=N\times D+D\times neg2+D\times neg1+neg1\times\log(neg1)\tag{10}$$

由于在本文算法中，$k=neg2$，因此，在CBOW方法中，强负采样比随机负采样多出的时间复杂度为$D\times neg1+neg1\times log(neg1)$，两项分别表示计算距离的复杂度和排序取前$neg2$个强负样本的复杂度。

而Skip-gram在随机负采样和强负采样下的时间复杂度分别是公式(11)和公式(12)。同理，Skip-gram的强负采样方法距离计算的复杂度是$C \times D \times neg1$，排序的复杂度是$C \times neg1 \times log(neg1)$。

$$Q = C \times (D + D \times k) \tag{11}$$

$$Q = C \times (D + D \times neg2 + D \times neg1 + neg1 \times \log(neg1)) \tag{12}$$

4 实验与结果分析

4.1 语料库

本文使用的语料库来自维基百科2019的转储数据，全部文件共包含26亿个词例。采用gensim库提供的语料库处理工具，可以对维基百科原始文件中的HTML元数据、超链接等冗余标签做预处理，同时，对语料库分词和小写化。因为gensim一次只允许一条记录驻留在内存中，所以理论上gensim可以处理任意大的语料库。特别注意，在处理原始语料时，不需要做词形还原。

4.2 评价方法

在评价词嵌入质量方面，本文采用Mikolov(2013)在论文中整理的语义-语法词相关测试集进行实验，该测试集也是业内通用的词嵌入质量评价数据集。测试集共包含19544组相关词对，涵盖5类语义（semantic）问题和9类语法（syntactic）问题。评价方法是，给出一组相关词对中的前3个词，嵌入后的词向量通过计算如果能够准确回答出第4个词则得分。回答正确的词对越多，则认为词嵌入的质量越高。

这种测试可以描述成“当a对应b时，c对应什么？”。例如，描述实体之间类比关系的语义问题：“当brother对应sister时，grandson对应什么？”，或描述时态、形态变化的语法问题：“当big对应biggest时，small对应什么？”。在一个理想的词嵌入空间中，上述问题通过向量的代数运算就可以回答。首先计算向量$X = vector(w_a) - vector(w_b) + vector(w_c)$，然后在$X$附近寻找余弦距离最近的词作为答案。

词向量计算的结果必须与测试集给出的第4个词完全匹配，同义词在本实验中不得分。另外，如果是自定义的语义-语法测试集，则需要注意测试集中的相关词对必须具有方向性，否则无法进行有效的向量运算。

4.3 实验结果

影响词嵌入质量的因素有很多，为了得到一个比较可信的HNS性能，对于一些通用的词嵌入训练参数，本次实验中将对其进行统一设置：上下文窗口大小为8，初始学习率为0.05，亚采样的概率为1e-4，迭代训练2次。大部分参数是word2vec工具的默认值，我们相信这些默认值可以带来一个相对优异的结果。对于HNS的参数，设置候选负样本个数为100，强负样本个数为15。

表1中我们比较了HNS方法与部分已经发布的预计算词向量在语义-语法测试数据上的准确率。基于HNS方法的CBOW模型称之为HNS-CBOW，基于HNS方法的Skip-gram模型称之为HNS-SG，HNS-CBOW和HNS-SG都训练了完整的维基百科26亿词例的语料库。其他模型结果：CBOW和SG的准确率结果来源于论文[9]，文中使用的测试集和本文相同；GloVe的词向量公开发布在斯坦福网站[0]，根据文档介绍该词向量在60亿词例的语料库上迭代训练了50次，其准确率由本文下载后测算得到。另外，我们控制所有词嵌入的词典大小都是由40万个最常出现的词组成，避免了词典大小对于准确率的影响。

结果表明，HNS模型使用更小的语料库和更少的迭代次数就能够达到比其他模型更高的整体准确率。当向量长度为100维和300维时，HNS-CBOW的语义准确率均最高，分别是74.4%和79.8%，同时，HNS-CBOW的整体准确率也最高，分别为64.8%和72.3%。但是，在语法准确率上，HNS方法表现地不够理想。

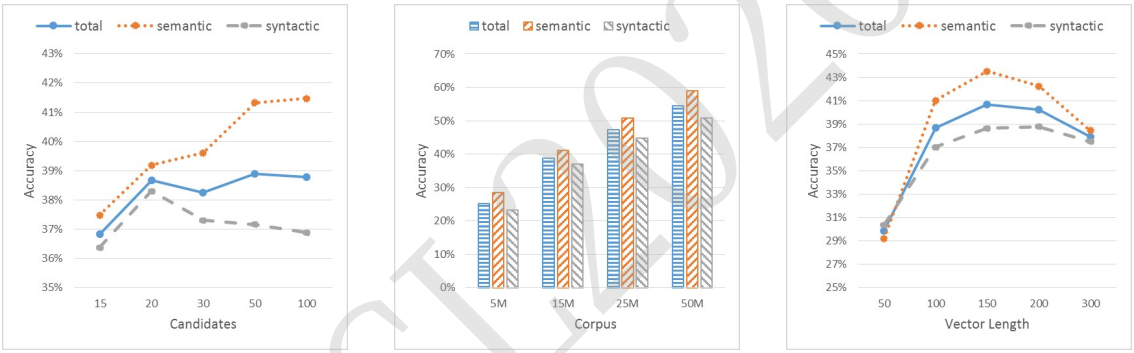
[0] https://nlp.stanford.edu/projects/glove/

Table 1: Accuracy of various word embeddings on the Semantic-Syntactic test set
表 1: 不同词嵌入在语义-语法测试集上的准确率

Model	Dim	Size	Semantic	Syntactic	Total
GloVe	100	6B	65.3%	**61.3%**	63.1%
HNS-CBOW	100	2.6B	**74.4%**	56.8%	**64.8%**
HNS-SG	100	2.6B	62.6%	46.1%	53.6%
CBOW	300	6B	63.6%	**67.4%**	65.7%
SG	300	6B	73.0%	66.0%	69.1%
GloVe	300	6B	77.4%	67.0%	71.7%
HNS-CBOW	300	2.6B	**79.8%**	66.1%	**72.3%**
HNS-SG	300	2.6B	74.6%	57.1%	65.1%

4.4 模型分析：参数分析

本节将讨论候选负样本集合大小、语料库大小、词向量长度等参数对于词嵌入质量的影响，包括语义准确率、语法准确率以及整体准确率。本节实验使用的训练数据是从维基百科语料库中随机抽取的5000篇文章，约1500万个词例组成。所有结果使用HNS-CBOW模型迭代15轮计算得到。



(a) (b) (c)

Figure 1: Semantic-syntactic accuracy on different params
图 1: 不同参数对语义-语法准确率的影响

在图1(a)中，我们展示了从不同大小的候选负样本集合中抽取10个强负样本对于词嵌入性能的影响。从图中可以看出，对于10个强负样本的采样要求，当候选集合从15增大到20时，总体、语义、语法3个准确率都有比较明显的提升；当候选集合从20增到100时，总体准确率就不再有明显变化，但此时语义准确率的提升还是比较明显的，不过其代价是降低了语法准确率。

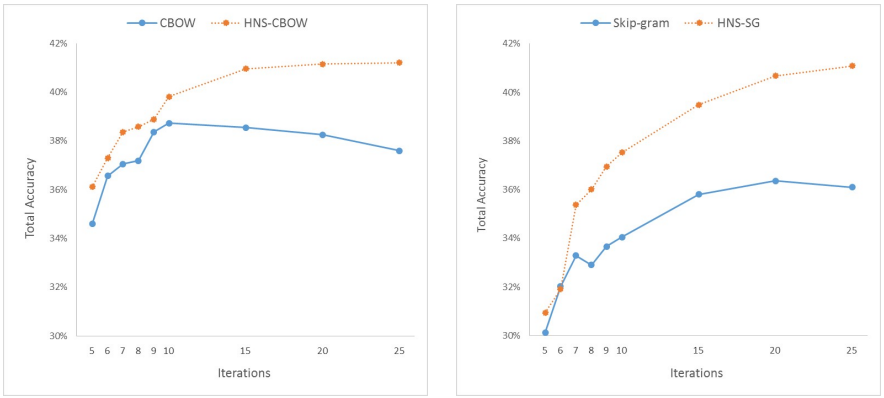
在图1(b)中，我们分别对拥有500万、1500万、2500万、5000万个词例的语料库进行了训练。意料之中的是，语料库越丰富，词嵌入的效果也越好。使用5000万词训练出的词向量比500万词的准确率提升了一倍以上。不过，随着语料库词例数量的增加，增大语料库对于词嵌入质量的提升效果是在逐渐降低的。

在图1(c)中，我们探索了在50到300的向量长度上，15轮迭代后能够达到的准确率。实验中使用1500万词例的语料库，当向量长度从50扩大到150时，准确率随着向量长度的增加而增大。但是当向量长度达到300时，准确率反而发生了下滑，这是因为，高维向量想要达到更高的准确率需要更多的迭代次数和更大的语料库来支撑，因此，在训练时间和语料库来源都受到限制的情况下，合理选择向量长度是必要的。

4.5 模型分析：与word2vec比较

为了严格比较HNS与word2vec原方法的区别，我们按照4.4的参数设置对上下文窗口、初始学习率、亚采样概率进行了控制，以使这些参数对最终准确率的影响降到最低。同时，设置候

选负样本个数为100，强负样本个数为15。作为对比，在使用word2vec工具时，选择通过负采样的方式进行训练，每次选取负样本的个数也设置为15个。另外，本次实验每一种方法都对相同的语料库进行了25轮迭代学习，语料库包含1500万个词例，并在每轮迭代后记录下词嵌入的整体准确率。



(a) (b)

Figure 2: Total accuracy on CBOW and Skip-gram
图 2: CBOW和Skip-gram方法的整体准确率

图2展示了不同迭代次数下基于HNS的词嵌入与word2vec工具在语义-语法测试集上的整体准确率变化。我们发现，在相同的参数下，不论CBOW还是Skip-gram框架，HNS方法几乎在每轮迭代后都表现出了更高的质量。同时，HNS方法也没有出现准确率下降的情况，如果不考虑训练时间的限制，HNS方法可以取得更好的结果。

5 结论

本文提出了一种在词嵌入训练过程中使用余弦距离衡量样本重要性的采样方法。该方法吸收了强采样的思想，结合word2vec的负采样方法，改善了词嵌入学习的的质量。本文使用公开的语义-语法测试数据，证明了HNS的有效性，改进后的HNS方法在CBOW和Skip-gram框架下比原方法都取得了更好的结果，相比于部分公开的预训练词向量也拥有更高的准确率。此外，本文探索了在HNS方法下不同参数对于词向量质量的影响，对比了不同候选负样本集合大小、语料库大小、词向量长度下语义-语法准确率的变化。结果表明，向量空间需要结合实际目标，针对特定的训练任务选择不同的参数组合。后续工作主要集中在两点，一是更高效的距离计算及排序方法，二是探索优化后的词向量对NLP下游任务的影响。

参考文献

Bansal M, Gimpel K, and Livescu K. *Tailoring Continuous Word Representations for Dependency Parsing* [C] // Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Short Papers). Baltimore, Maryland, USA: Association for Computational Linguistics, 2014: 809-815

Goldberg Y and Levy O. *word2vec Explained: Deriving Mikolov et al.'s Negative-Sampling Word-Embedding Method* [J]. arXiv preprint arXiv:1402.3722, 2014

Habibi M, Weber L, Neves M, et al. *Deep Learning with Word Embeddings Improves Biomedical Named Entity Recognition* [J]. Bioinformatics, 2017, 33(14): i37-i48

Ji Shihao, Satish N, Li Sheng, et al. *Parallelizing Word2Vec in Multi-Core and Many-Core Architectures* [J]. arXiv preprint arXiv:1611.06172, 2016

Katharina S. *Adapting word2vec to Named Entity Recognition* [C] // Proceedings of the 20th Nordic Conference of Computational Linguistics. Vilnius, Lithuania: Linköping University Electronic Press, 2015: 239-243

Lilleberg J, Zhu Yun, and Zhang Yanqing. *Support Vector Machines and Word2vec for Text Classification with Semantic Features* [C] // 2015 IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing (ICCI*CC). Beijing, China: IEEE, 2015

Mikolov T, Chen Kai, Corrado G, et al. *Efficient Estimation of Word Representations in Vector Space* [J]. arXiv preprint arXiv:1301.3781, 2013

Mikolov T, Sutskever I, Chen Kai, et al. *Distributed Representations of Words and Phrases and their Compositionality* [J]. arXiv preprint arXiv:1301.4546, 2013

Miyato T, Dai A, and Goodfellow I. *Adversarial Training Methods for Semi-Supervised Text Classification* [J]. arXiv preprint arXiv:1605.07725, 2017.

Pennington J, Socher R, and Manning C. *GloVe: Global Vectors for Word Representation* [C] // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar: Association for Computational Linguistics, 2014: 1523-1543

Schroff F, Kalenichenko D, and Philbin J. *FaceNet: A Unified Embedding for Face Recognition and Clustering* [C] // 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, MA, USA: IEEE, 2015: 815-823

Wang Ling, Dyer C, Black A, et al. *Two/Too Simple Adaptations of Word2Vecfor Syntax Problems* [C] // Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL. Denver, Colorado, USA: Association for Computational Linguistics, 2015: 1299-1304

Ye Xin, Shen Hui, Ma Xiao, et al. *From Word Embeddings to Document Similarities for Improved Information Retrieval in Software Engineering* [C] // ICSE '16 Proceedings of the 38th International Conference on Software Engineering. New York, NY, USA: ACM, 2016: 404-415

联合依存分析的汉语语义组合模型

陈圆梦，张玉洁†，徐金安，陈钰枫
北京交通大学 计算机与信息技术学院，北京 100044
† 通讯作者，E-mail:yjzhang@bjtu.edu.cn

摘要

在语义组合方法中，结构化方法强调以结构信息指导词义表示的组合方式。现有结构化语义组合方法使用外部分析器获取句法结构信息，导致句法分析与语义组合相互割裂，句法分析的精度严重制约语义组合模型的性能，且训练数据领域不一致等问题会进一步加剧性能的下降。对此，本文提出联合依存分析的语义组合模型，将依存分析与语义组合进行联合，一方面在训练语义组合模型时对依存分析模型进行微调，使其能够更适应语义组合模型使用的训练数据的领域特点；另一方面，在语义组合部分加入依存分析的中间信息表示，获取更丰富的结构信息和语义信息，以此来降低语义组合模型对依存分析错误结果的敏感度，提升模型的鲁棒性。我们以汉语为具体研究对象，将语义组合模型用于复述识别任务，并在CTB5汉语依存分析数据和LCQMC汉语复述识别数据上验证本文提出的模型。实验结果显示，本文所提方法在复述识别任务上的预测正确率和F1值上分别达到76.81%和78.03%；我们进一步设计实验对联合学习和中间信息利用的有效性进行验证，并与相关代表性工作进行了对比分析。

关键词： 句法分析；语义组合；联合学习；图注意力网络

Chinese Semantic Composition Model with Dependency Parsing

Yuanmeng Chen, Yujie Zhang† Jinan Xu, Yufeng Chen
School of Computer and Information Technology, Beijing Jiaotong University
Beijing 10004
†Corresponding Author, E-mail:yjzhang@bjtu.edu.cn

Abstract

In the semantic composition methods, the structural methods emphasize the combination mode of words' meaning representation guided by structural information. Existing structural semantic composition methods use external parser to obtain syntactic structure information, resulting in the separation of syntactic parsing and semantic composition. The accuracy of syntactic analysis will severely restrict the performance of semantic composition models, and the inconsistent training data fields will further aggravate the performance degradation. To solve this problem, this paper proposes a semantic composition model combined with dependency parsing. On the one hand, the dependency model is fine-tuned when training the semantic composition model, so that it can be more suitable for the domain characteristics of the training data used

国家自然科学基金（61876198,61976015,61976016）资助

by the semantic composition model. On the other hand, we add the intermediate information representation of dependency to the semantic composition part to obtain more abundant structural information and semantic information, so as to reduce the sensitivity of semantic composition model to erroneous results of dependency parsing and improve the robustness of the model. We take Chinese as the specific research object, apply semantic combination model to retelling recognition task, and verify the model proposed in this paper on CTB5 Chinese dependency parsing data and LCQMC Chinese retelling recognition data. The experimental results show that the prediction accuracy and F1 value of the method proposed in this paper reach 76.81% and 78.03% respectively in retelling recognition tasks. We further designed experiments to verify the effectiveness of joint learning and intermediate information utilization, and made comparative analysis with relevant representative work.

Keywords: Syntactic analysis , Semantic combination , Joint learning , Graphical attention network

1 引言

语义组合以一定的方式将句子中的词义表示进行计算合并，从而得到句子的语义表示。作为语义组合的重要组成部分，组合方式的选择对最终得到的语义表示的性能有着重要影响。目前主流的组合方式是序列化的语义组合方法，仅对句子进行序列化处理，忽视了句子的语法结构，导致获取的句子表示难以准确地反应句子的语义。汉语由于词序更加灵活，且缺乏表层变化信息等特点，因此在计算句子语义时需要句法结构信息的指导。依存句法信息由于与词义和语义关联更为密切，因此在汉语的语义表示研究领域，依存句法分析和语义组合的结合是未来主要的研究方向之一。

目前一些研究者尝试利用依存句法信息作为指导，构建树结构的语义组合方法，通过依存分析器预测句子的依存结构，然后根据依存树结构进行语义组合，在句子匹配等任务上取得了一定的成就(Mou et al., 2016)。但这类方法仍存在如下问题：（1）依存分析和语义组合相互割裂。现有方法直接使用外部依存分析器获得的依存句法信息，没有针对语义组合任务进一步优化依存分析模型，从而限制了最终获取的语义表示的精度。（2）数据领域不一致。依存分析与语义组合的训练数据可能来自不同领域，将会导致依存分析模型在应用于语义组合数据时精度降低，进而影响语义组合模型的性能。（3）信息利用不充分。使用外部依存分析器获取依存句法信息，仅能利用预测得到的依存句法树，而在依存分析过程中产生的结构信息和语义信息则未加利用，浪费了大量的中间信息。

针对上述问题，本文提出联合依存分析的语义组合模型。以依存句法树作为图注意力计算中的图，对每个节点的语义根据其孩子节点进行组合计算；然后提出依存分析中间信息的利用方法，将依存关系中作为头节点的语义信息引入语义组合模型，以降低依存分析的预测错误对语义组合模型带来的影响，提升语义组合模型的鲁棒性；最后通过依存分析与语义组合的联合学习，对依存分析模型进行领域自适应，提升依存分析模型的鲁棒性。我们将语义组合模型用于复述识别任务，在汉语复述识别数据集LCQMC上的预测正确率达到76.81%，F1值达到78.83%。

2 相关工作

目前语义组合方法主要可以分为两类：一种是将句子视为序列结构进行组合，将句子中各个词的信息进行加权整合，从而得到能够有效表达句子语义的表示；另一种则是利用句法结构作为语义组合的指导，根据句子中的结构关系对词义表示的组合顺序和方式加以限制，得到能更准确表达句子语义的表示。

对于如何通过组合词汇语义得到句子语义，一种朴素的思想是将词义表示相加得到句子语义表示，一般这种方式被称为加法组合（Additive Compositionality）(Mikolov et

al., 2013)。Hu et al. (2015)借鉴图像处理的技术，提出基于多层卷积操作的语义组合方法。Sutskever et al. (2014)和Cho et al. (2014)在他们提出的seq2seq（sequence to sequence）模型中，将RNN模型在最后一步的输出作为整个句子的语义表示,利用RNN类模型能够充分利用长距离信息的特点（以LSTM和GRU等变种为主），将句子信息进行有效地融合。该方法一度随seq2seq模型一道成为机器翻译、语音识别等生成任务中常用的语义组合方法。Chen et al. (2017)考虑到加法组合的语义组合方法会受到句子长度的影响，因此提出通过平均池化和最大池化相结合的方法，将句子长度的影响消去。该方法以其简单高效和对句子长度不敏感的特性，成为目前句子匹配任务中主流的语义组合方法。

鉴于目前句子结构主要被定义为树结构，因此结构化语义组合也以递归神经网络为基本模型。Zhu et al. (2015)对LSTM单元进行修改，提出针对二叉树句法结构的S-LSTM，利用LSTM能够进行长距离信息传递的特性，将各个词的信息经转化为二叉结构的短语句法树逐层传递至根节点，从而得到句子的语义表示。Mou et al. (2016)利用CNN能够轻松处理递归结构的特性，提出TBCNN（tree-based convolutional neural network），将依存句法树中每两层的语义信息加以融合，然后通过池化合并得到最终的句子语义表示。

序列化语义组合方法的优点是模型结构简单，能够快速得到句子表示，但由于忽视了句子内在的结构，对于序列差异较小的句子则难以区分。结构化语义组合方法能够更精确地表达句子的语义，但对依存分析的精度有较高的要求，同时模型更为复杂，时间消耗也更长。由于获取高质量句法分析标注难度较大，而使用外部句法分析器获取句法结构，可能会由于句法分析与语义组合数据领域不一致的问题，导致句法分析精度降低。本文针对现有结构化语义组合方法存在的问题，提出联合依存分析的汉语语义组合模型，将依存分析与语义组合计算进行联合，并提出依存分析中间信息的利用方法，从而得到更好的句子语义表示。

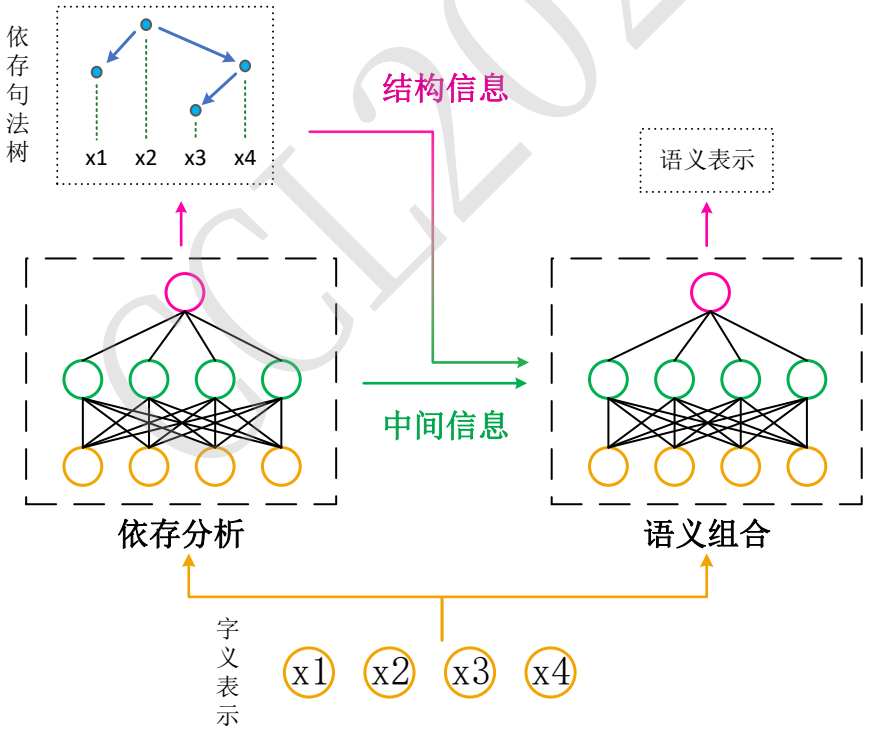


图 1: 联合依存分析的汉语语义组合计算模型

3 联合依存分析的汉语语义组合模型

针对现有结构化语义组合方法存在的问题，本文在Ma et al. (2018)的基础上，联合基于注意力的语义组合模型，提出联合依存分析的汉语语义组合模型。如图1所示，我们的模型主要包含依存分析和语义组合两大部分。依存分析部分对输入句子进行依存句法分析，并将分析过程中产生的中间信息和最终得到的依存句法树传递给语义组合部分；语义组合部分根据句子中每

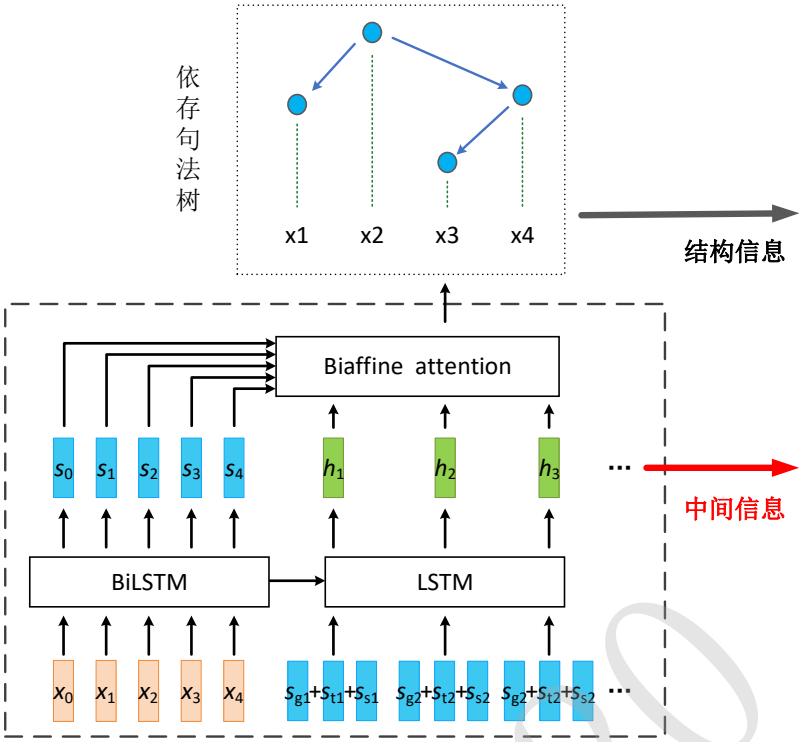


图 2: 联合模型中的依存分析部分

个字的字义表示，加入依存分析部分产生的中间信息，以依存句法树作为指导进行语义组合，从而得到句子的语义表示。

3.1 基于Stack-Pointer Networks的依存分析模型

Stack-Pointer Networks(StackPTR)是一个基于转移的依存分析模型。我们在StackPTR的基础上，针对汉语没有明显分词标记的特点，以每个词的最后一个字为根节点构建两层词内依存结构，以此进行字符级汉语依存分析。StackPTR模型大致框架如图2所示，其中每一步计算中得到的头节点表示h_i包含较为丰富的结构信息，因此我们将其作为依存分析的中间信息表示传递给语义组合部分。StackPTR中的具体细节请见Ma et al. (2018)。

3.2 基于注意力的语义组合模型

虑到每个字对句子语义的贡献程度不同，我们在语义计算部分提出基于注意力的语义组合模型，利用依存分析部分得到的结构信息作为指导，用注意力得分作为信息的权重，进行字义表示的组合。

如图3所示，模型主要分为字义编码层、字义组合层和句义输出层三个部分。字义编码层对每个字的语义表示进行编码；字义组合层以依存句法树作为语义计算的结构，将每个依存节点的信息传递给头节点；句义输出层将语义组合计算得到的每个字的结构化语义信息进行池化合并，得到表示句子语义的向量表示。

3.2.1 字义编码层

参考Hochreiter and Schmidhuber (1997)，我们使用双向LSTM进行字义表示的编码。依存分析模型的中间信息以字向量的形式，传递了丰富的结构信息和语义信息。我们将其作为额外的字义信息，对预训练字向量进行扩充。对于给定的句子$x = \{x_1, x_2, \ldots, x_n\}$，编码层首先将每个字的原始向量表示$x_i$与依存分析中间信息表示$h_i$进行拼接，得到字的向量表示$x_i'$。然后我们将输入双向LSTM，编码句子信息得到每个字在句子中的语义表示m_i。

此外，为了提升句子全局信息的利用，我们将双向LSTM两个方向的最后一步输出进行拼接，得到句子表示m_x，作为字义组合层中的额外信息输入。

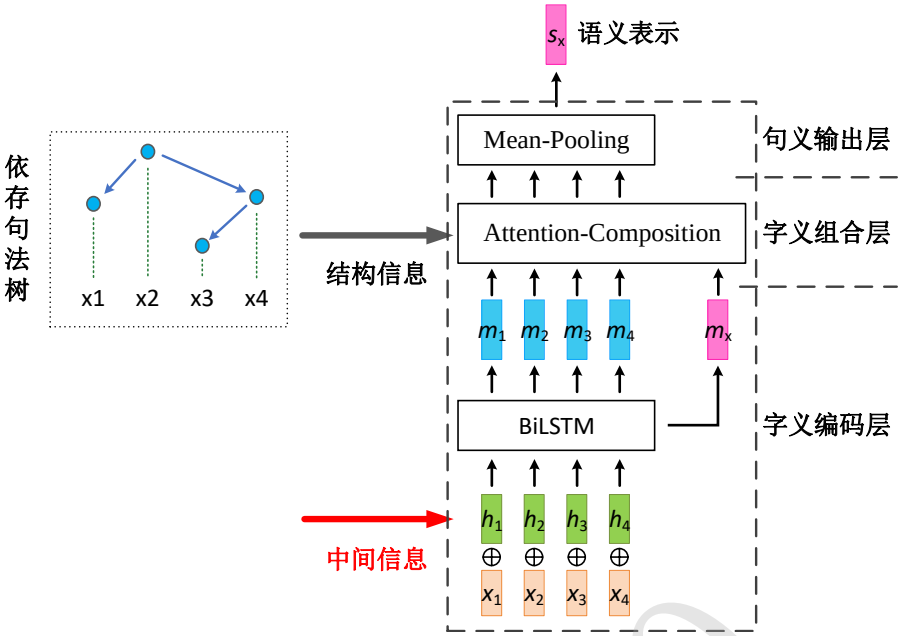


图 3: 基于注意力的语义组合模型

3.2.2 字义组合层

我们在语义组合计算层使用图注意力网络(Hochreiter and Schmidhuber, 1997)进行字义表示的组合计算，其中依存分析部分预测出的依存句法树作为指示节点相关性的有向图，对字义编码层输出的字义表示进行语义组合，其中每个节点在计算时仅考虑其依存节点，如图4所示。

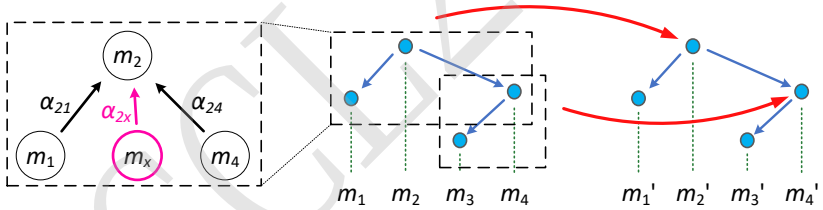


图 4: 语义组合计算层中的信息计算方式

我们将字义编码层中得到的句子表示m_x作为一个额外的节点，作为所有节点的依存节点参与图注意力网络的计算，使组合计算后每个字的表示都包含不同程度的句子全局信息。我们选择使用双线性变换作为注意力得分的计算机制，因此字义组合层的计算公式即为：

$$m_i^{'} = m_i + \sum_{j \in V(i)} m_i W m_j + m_i W m_x \times m_x \tag{1}$$

其中m_i和$m_i^{'}$分别表示第i个字在进行字义组合计算前后的语义表示；$j \in V(i)$表示节点i的所有依存节点，W为双线性变换的参数矩阵。

3.2.3 句义输出层

借鉴Mou et al. (2016)的工作，我们使用池化操作对字义组合计算后的字向量进行池化操作，获得最终的句子语义表示。为了使句子的语义表示不受句子长度的影响，同时尽可能保存更多的语义信息，我们选择使用平均池化对语义组合层输出的字向量表示进行合并。最终句子语义表示s_x的计算公式如下：

$$s_x = \frac{1}{n} \sum_{i=1}^{n} m_i^{'} \tag{2}$$

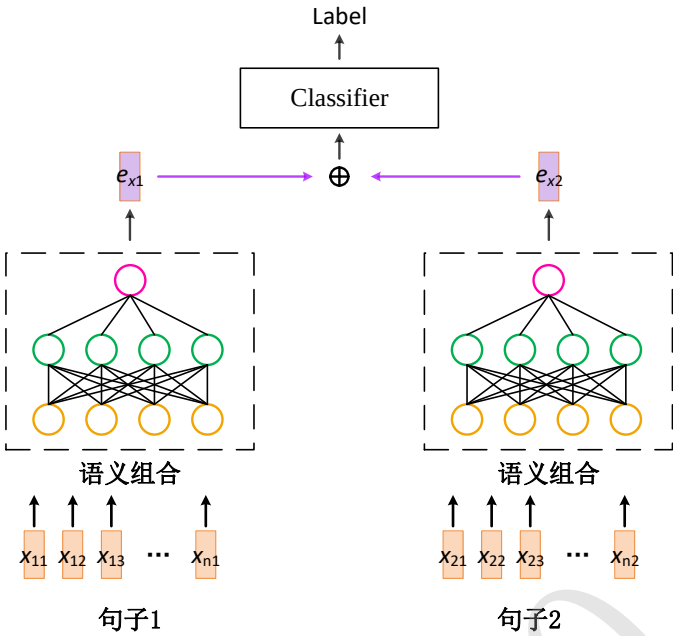


图 5: 复述识别模型

3.3 模型训练

3.3.1 复述识别任务

复述识别的主要目标是判断给定的两个句子是否表达相同（或相近）的语义。本文认为在所有相关任务中，复述识别任务与句子语义关联最为紧密，因此提出基于复述识别任务的语义组合计算的训练和评测方法。为了能够更直观地反应语义组合模型对复述识别性能的影响，我们未使用目前复述识别任务中最流行的交叉注意力机制(Veličković et al., 2017)和预训练语言模型(Wang et al., 2017)，而是借鉴早期常用的获得句子语义表示后进行关系预测的方式(Hu et al., 2015)，对每个句子独立进行语义组合计算，然后使用分类器进行复述的判别。

我们的复述识别模型如图5所示，使用本章提出的语义组合模型（依存分析部分未在图中显示）对两个输入句分别进行语义组合计算，得到它们的语义表示。然后将两个句子的语义表示拼接后输入一个分类器，判别它们是否互为复述。

3.3.2 联合模型训练方式

在进行联合模型的训练时，我们考虑两种训练方式：一种是将依存分析模型进行预训练，然后在训练语义组合模型时对依存分析模型的参数进行微调；另一种是直接将两个任务进行迭代训练。

预训练方式： 我们首先对联合模型中的依存分析部分的模型参数进行预训练，然后进行复述识别任务的训练，并在训练过程中对依存分析部分的模型参数进行微调。模型的目标函数通过最小化交叉熵损失定义：

$$\mathcal{L}(\theta) = \mathcal{L}_{par}(\theta_{com}, \theta_{dep}) = -logP_{par}(y|x, \theta_{dep}, \theta_{com}) \quad (3)$$

其中θ_{com}和θ_{dep}分别表示依存分析和语义组合模型的参数，表示复述关系标签。

迭代训练方式：在模型训练时，我们每次随机从两个任务中选择一个任务，并从对应的训练集中随机选取一批数据进行模型的训练。其中，在进行依存分析任务的训练时，仅对依存分析部分的模型参数进行学习；在进行复述识别任务的训练时，同时对联合模型中两个部分的参数进行学习。模型的目标函数通过最小化交叉熵损失定义：

$$\mathcal{L}(\theta) = \mathcal{L}_{dep}(\theta_{dep}) + \mathcal{L}_{par}(\theta_{com}, \theta_{dep}) = -logP_{dep}(A|x, \theta_{dep}) - logP_{par}(y|x, \theta_{dep}, \theta_{com}) \quad (4)$$

其中θ_{com}和θ_{dep}分别表示依存分析和语义组合模型的参数，A表示字节别依存句法树的边集，y表示复述关系标签。为了使模型训练过程更为稳定，我们首先对依存分析模型进行了100次训练。

4 实验

4.1 实验设置

本文使用的依存分析实验数据为宾州汉语树库CTB5，复述识别实验数据为语义相似度数据集LCQMC(Liu et al., 2018)。两个数据集的划分及统计数据如表1和2所示。

数据集	句子数	平均句长	词数	字数
训练集	18104	44.4	494k	805k
开发集	352	32.8	7k	12k
测试集	348	39.5	8k	14k

表 1: CTB5数据划分

数据集	句对数	平均句长	字数	正例占比
训练集	239k	10.9	5.2m	0.58
开发集	9k	12.5	0.2m	0.50
测试集	13k	9.7	0.2m	0.50

表 2: LCQMC数据详情

4.2 参数设置

我们使用word2vec工具在gigaword生语料上预训练字向量，字向量维度为100维；LSTM隐藏层维度为400，Dropout率为0.33。模型训练使用Adam（Adaptive Moment Estimation）优化算法，依存分析模型初始学习率设置为0.002，复述识别模型初始学习率设置为0.0001。对于预训练的模型训练方法，为了对依存分析部分的模型参数进行微调，我们对其设置了一个较小的学习率0.00001。

4.3 实验结果与分析

4.3.1 联合模型训练方式对比

我们分别对管道模型（pipeline）、预训练方法（pre-train）和迭代训练方法（alternate）进行实验，并在复述识别和一体化依存分析任务上进行了比较。其中，我们将管道模型中依存分析部分的参数学习率设置为0，用以模拟使用外部依存分析器提供结构信息的传统结构化方法。

复述识别任务：三个模型在复述识别任务上的对比结果如表3所示，其中预训练和迭代训练两种方式相较于管道模型均有明显的提升，且使用迭代训练方式的模型取得了最好的结果。

模型	类型	Acc(%)	F1(%)
Ours (pipeline)	结构化（管道）	72.64	75.74
Ours (pre-train)	结构化（联合）	74.01	76.86
Ours (alternate)		76.37	78.03

表 3: 联合模型训练方式在复述识别任务上的对比结果

一体化依存分析：三个模型在一体化依存分析任务上的对比结果如表4所示，其中两种训练方式得到的模型都较参数调整前的依存分析模型性能更低，且使用迭代训练方式的模型降低更为明显。

模型	分词(%)	词性标注(%)	依存分析(%)
Ours (pipeline)	98.25	95.13	85.44
Ours (pre-train)	97.85	94.35	83.04
Ours (alternate)	97.93	94.22	82.13

表 4: 联合模型训练方式在依存分析上的对比结果

总结分析：总的来说，虽然我们的联合模型在一体化依存分析任务上的精度有所降低，但在复述识别任务上的精度有所提升。我们根据表1和表2中两种数据平均句长的对比，以及进行数据分析后发现：我们所使用的依存分析数据（CTB5）为新闻领域的文本，句子较长且表达形式较为书面化；复述识别数据（LCQMC）为搜索引擎上收集的问句，句子较短且表达形式较为口语化。两种数据在领域和语言现象上存在较大的差异，因此我们做出如下推断：（1）使用CTB5上训练的依存分析模型，在对LCQMC中的句子进行的依存分析精度会有明显的降低，并因此导致语义组合模型在复述识别任务的精度较低；（2）我们的联合模型能够针对LCQMC的数据特点，对依存分析部分的参数进行适当地调整，虽然使其在CTB5上的一体化依存分析精度有所降低，但能够隐式地提升其在LCQMC数据上的依存分析精度，进而提升在复述识别任务上的精度。

4.3.2 依存信息利用的对比

我们对联合模型中的语义组合部分使用到的依存信息进行消融实验。分别对比了不使用依存结构信息（without-structure）和不使用依存中间信息（without- intermediate），实验结果如表5所示。

模型	ACC(%)	F1(%)
Ours (alternate)	76.37	78.03
Ours (without-structure)	75.70	76.78
Ours (without- intermediate)	75.86	77.07

表 5: 依存信息利用对比结果

从表中结果可以看出，去掉依存结构信息和去掉依存中间信息都会带来复述识别精度的明显下降。其中去除依存结构信息带来的性能降低较为明显，表明依存句法信息能够提升语义组合计算模型的性能；去除依存中间信息带来的性能降低表明，我们的语义组合模型能够有效利用依存分析过程中产生的语义表示，对汉字语义进行适当的补充，提升汉字表示包含。

4.3.3 语义组合方法对比

我们在本章所提的基于注意力的语义组合模型中，对字义组合层和句义输出层进行替换，实现了常见的语义组合计算方法，并与本章所提方法在复述识别任务上进行对比。对比的序列化方法包括平均池化（Mean）、基于CNN的方法（CNN）(Hu et al., 2015)和基于LSTM的方法（LSTM）(Sutskever et al., 2014; Cho et al., 2014)，结构化方法包括采用并列化处理的基于树卷积神经网络的方法（Tree-CNN）(Mou et al., 2016)。其中序列化方法将不使用依存分析模型提供信息，结构化方法将依存分析模型的参数学习率设置为0。对比结果如表6所示。

从对比结果可以看出，我们的模型较现有常见的语义组合计算方法，在复述识别任务上的预测准确率和F1值均有较明显的提升。其中较最好的LSTM方法分别提升了0.83%和1.68%。表明我们的方法能够有效地利用依存句法信息和依存中间信息，从而获取更为准确的语义表示，并反应在下游任务中。

此外，我们实现的结构化方法Tree-CNN和我们的管道模型在复述识别任务上性能接近，但较序列化方法的性能有较明显的下降。我们认为这是由于依存分析和复述识别任务的数据领域不一致的问题，对结构化语义组合方法带来的巨大影响，侧面反映了我们提出的联合模型能够有效降低数据领域不一致的问题。

模型	类型	Acc(%)	F1(%)
Baseline (Mean)	序列化	73.21	74.72
CNN		74.84	76.27
LSTM		75.53	76.31
Tree-CNN	结构化（管道）	72.93	75.46
Ours (pipeline)		72.64	75.74
Ours (alternate)	结构化（联合）	**76.37**	**78.03**

表 6: 语义组合计算方法对比结果

4.3.4 与现有复述识别模型的对比

我们与现在常用的复述识别模型(Wang et al., 2017; Devlin et al., 2019)在LCQMC数据集上的最好结果进行了比较，结果如表7所示。

模型	ACC(%)	F1(%)
BiMPM	83.4	**85.0**
Bert-large	**87.3**	-
Ours (alternate)	76.37	78.03

表 7: 复述识别性能对比

对比结果表明，我们的模型较现有复述识别方法有十分明显的差距。经过分析，我们认为这主要由以下原因导致：1）现在主流复述识别方法主要使用交叉注意力机制（Cross-Attention）进行字义表示的学习，即对两个句子中的字向量进行注意力机制的计算，这样能够对两个句子的相关信息进行利用，已有工作(Liu et al., 2018)表明交叉注意力机制对复述识别任务的性能能够带来显著提升。本章中复述识别任务的主要目的在于对语义组合计算模型的性能进行评价，因此未使用交叉注意力机制，仅对单句信息进行利用。2）以Bert为主的预训练语言模型能够显著提升字向量在具体句子中语义表示的精度，并且能够在大规模训练集中提取丰富的语言学知识，以此提升下游任务的性能。本章主要针对语义组合计算模型中的结构化信息利用和模型联合方法进行改进，仅使用word2vec预训练的静态字向量，且训练数据较小。今后可以尝试引入预训练语言模型，进一步验证我们所提方法的有效性。

5 总结

本文针对现有结构化语义组合计算方法的不足，提出联合依存分析的语义组合计算方法，在现有依存分析模型的基础上，使用图注意力网络根据依存句法树进行语义组合计算，并利用依存分析中间信息对语义组合计算的字义表示进行补充，提升模型的鲁棒性。今后我们将在现有模型中加入预训练语言模型，提升字义表示的性能，以此来进一步提升语义组合计算模型的性能。

参考文献

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1657–1668. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*

2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 1724–1734. ACL.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2015. Convolutional neural network architectures for matching natural language sentences. *CoRR*, abs/1503.03244.

Xin Liu, Qingcai Chen, Chong Deng, Huajun Zeng, Jing Chen, Dongfang Li, and Buzhou Tang. 2018. LCQMC:a large-scale Chinese question matching corpus. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1952–1962, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Xuezhe Ma, Zecong Hu, Jingzhou Liu, Nanyun Peng, Graham Neubig, and Eduard H. Hovy. 2018. Stack-pointer networks for dependency parsing. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1403–1414. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

Lili Mou, Rui Men, Ge Li, Yan Xu, Lu Zhang, Rui Yan, and Zhi Jin. 2016. Natural language inference by tree-based convolution and heuristic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. The Association for Computer Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. In Carles Sierra, editor, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 4144–4150. ijcai.org.

Xiao-Dan Zhu, Parinaz Sobhani, and Hongyu Guo. 2015. Long short-term memory over recursive structures. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1604–1612. JMLR.org.

基于对话约束的回复生成研究

管梦雨，王中卿*，李寿山，周国栋
苏州大学计算机科学与技术学院，苏州，中国
{20195227007}@stu.suda.edu.cn，{wangzq，lishoushan，gdzhou}@suda.edu.cn

摘要

现有的对话系统中存在着生成“好的”、“我不知道”等无意义的安全回复问题。日常对话中，对话者通常围绕特定的主题进行讨论且每句话都有明显的情感和意图。因此该文提出了基于对话约束的回复生成模型，即在Seq2Seq模型的基础上，结合对对话的主题、情感、意图的识别。该方法对生成回复的主题、情感和意图进行约束，从而生成具有合理的情感和意图且与对话主题相关的回复。实验证明，该文提出的方法能有效地提高生成回复的质量。

关键词： 对话生成；主题识别；情感识别；意图识别

Research on Response Generation via Dialogue Constraints

Mengyu Guan，Zhongqing Wang，Shoushan Li，Guodong Zhou
School of Computer Science and Technology, Soochow University, Suzhou, China
{20195227007}@stu.suda.edu.cn，{wangzq，lishoushan，gdzhou}@suda.edu.cn

Abstract

Existing dialogue systems tend to generate meaningless general replies such as ”OK” and ”I don’t know”. In daily dialogs, every utterance usually has obvious emotional and intentional tendencies. So this paper proposes a response generation model based on dialogue constraints. Based on the Seq2Seq model, it combines the recognition of utterances’ themes, sentiments and intentions. This method constrains the topics, emotions and intentions of the generated responses, generating responses with reasonable sentiment and intention tendencies and related to the topic of the conversation. Experiments show that the method proposed in this paper can effectively improve the quality of generated responses.

Keywords: Dialogue generation , Topic recognition , Sentiment recognition , Act recognition

1 引言

*通讯作者：wangzq@suda.edu.cn

人机交互(Human Computer Interaction, HCI)作为信息时代人类与计算机之间信息交流的基础技术，受到学术界和工业界的广泛关注。人机对话(Human-Machine Dialogue)是人机交互技术的核心领域，旨在最大限度地模仿人与人之间的对话方式，使得人类能够用更自然的方式与机器进行交流。其应用场景广泛，具有较高的研究价值和商业价值。构建一个较完备的人机对话系统涉及到NLP技术的很多方面，比如句法分析 (Fried et al., 2017)，命名实体识别 (Huang et al., 2015)等。本文主要研究多轮对话的回复生成。简而言之就是根据历史对话信息，自动生成自然合理的回复，在信息交互的过程中协助用户完成特定的任务。随着端到端框架在机器翻译 (Brown et al., 1993)任务上的良好表现，研究人员将其迁移应用于对话生成任务中。对话生成可以简化为输入输出的映射问题，即对对话的输入进行编码和解码从而得到应答。因为对话是有时序的，可以视为序列，所以端对端框架下的序列对序列模(Sequence-to-Sequence, Seq2Seq) (Sutskever et al., 2014; Cho et al., 2014)非常适合对话生成模型。

主题简要地表明了整段对话的内容，用于理解对话的含义；情感表明了话语的极性，用于识别说话者的观点；意图作为话语的语义标签，用于描述说话者的动作意图。原则上，对主题、情感和意图的识别有助于对话语内容的理解。表 1给出了一组对话示例，整段对话是关于“日常生活”的讨论且对话中的每句话都有明确的情感和意图倾向。我们发现，同一说话者的情感往往是保持不变的；说话者B作为对话中被动的一方，意图往往受到对话发起者A的意图的影响。因此本文通过识别对话的主题及每个话语的情感和意图，对生成的回复进行约束。若我们要生成的是第二轮中说话者B的回复，我们的方法将生成与对话主题相关、情感倾向于“中性”、意图倾向于“承诺”的回复，而不是我们通常得到的例如“好的”，“我不知道”等安全回复 (McKay and Piperno, 2014)。

以往的对话生成模型中往往忽视了主题、情感和意图识别的重要性，因此本文提出了基于对话约束的回复生成模型。具体来说，在训练阶段，一方面，编码器中我们采用单词级别的LSTM网络对对话历史信息中的每个子句进行特征抽取，得到每个上下文的特征向量。我们将所有的上下文的特征向量整合为一个固定维度的向量作为整组对话的中间语义向量。解码器中我们同样采用单词级别的LSTM网络，同时考虑中间语义向量和目标回复向量生成回复。另一方面，我们用上下文的特征向量和生成回复的特征向量来预测整段对话的主题和每个子句的情感和意图。在我们提出的方法中主题预测模型、情感预测模型、意图预测模型和对话生成模型共享了编码和解码阶段的参数。在编码阶段，通过对主题、情感、意图的识别，我们可以更好地理解对话中各子句的语义，生成更高质量的中间向量；在解码阶段，在更高质量的中间语义向量的基础上，对生成回复的情感和意图进行识别，使得我们生成的回复与对话信息相关且具有合理的情感和意图倾向。在生成阶段，我们仅需将历史对话信息作为模型输入，即可生成更高质量的回复。在DailyDialog数据集上的相关实验结果表明，我们提出的模型明显优于基线系统。

本文的组织结构如下：第一节介绍论文的研究背景和意义；第二节主要介绍了对话生成的相关工作；第三节主要描述了基于对话约束的回复生成模型；第四节是实验设置的介绍和实验结果分析；最后，在第五节中对研究工作进行总结，同时提出了下一步的研究方向。

A	The sun is beginning to shine. What a lovely summer day!（积极，陈述） 阳光正好，多么美好的夏日啊！
B	Yeah, clearly blue sky. But it is a bit too hot for me.（中性，陈述） 是的，显然是蓝天。但是对我来说有点热。
A	It’s not that hot. It’s cooler than yesterday. Let’s go swimming!(积极，指示) 没那么热。今天比昨天凉。我们去游泳吧！
B	If we don’t stay too long, we won’t get sun-burned. （中性，承诺） 如果我们不停留太久，我们不会被晒伤。

表 1. 对话示例

2 相关工作

随着社交网络中大量的聊天语料的积累和硬件计算性能的提升，通过深度学习 (LeCun

et al., 2015)技术自主提取对话特征，自动学习生成回复成为可能。Google的Vinyals and Le (2015)首先将机器翻译中的“序列-序列”模型应用于对话生成任务中，其将回复生成问题视作翻译问题，编码器与解码器均采用RNN提取特征，例如长短期记忆网络(Long Short-Term Memory Network, LSTM) (Hochreiter and Schmidhuber, 1997)、门控循环神经单元(Gated Recurrent Unit, GRU)等，在开放领域和特定领域的预料上进行训练，均能得到较理想的回复。对话通常是一个持续，动态的过程，生成回复是需要考虑当前的对话语境，即历史信息。Wen et al. (2015)在神经Seq2Seq对话生成模型中，针对对话任务的依赖历史信息的特征，改进了LSTM模型，提出了一种语义控制LSTM应用到对话生成任务中。Sordoni et al. (2015)提出了考虑历史对话信息，设计了多轮对话的模型，编码器采用多层前向神经网络(Multilayer Feed-Forward Neural Network)代替循环神经网络模型(Recurrent Neural Network, RNN)，将对话中的历史信息和用户对话信息一起进行编码。Serban et al. (2016)则提出采用层级神经网络(Hierarchical Neural Network, HNN)以解决上下文中多个句子的编码问题，从词级别对句子进行编码，并用句子编码对上下文进行编码。不同的上下文语境对生成回复内容有着不同的影响，Tao et al. (2018)引入了注意力机制(attention mechanism)，通过计算不同时刻上下文的权重，决定上下文内容在回复中的表达程度，取得了更好的效果。在对话生成过程中，不仅要结合语境还要考虑一定的常识和背景知识，才能得到更加合理的回复。Kumar et al. (2016)提出了动态记忆网络(Dynamic Memory Network, DMN)，在考虑历史对话信息的基础上，结合特定的背景和常识信息生成回复。通常，对话生成中的Seq2Seq模型使用极大似然估计为目标函数，容易生成符合语法，但是通用的回复，例如“我不知道”这样的回复。Li et al. (2016)提出了最大互信息的模型，通过改变目标函数加强对话消息和回复之间的相关性，以避免最大似然估计作为目标函数生成安全回复的问题，生成多样化的回复。Li et al. (2017a)还借鉴生成对抗网络(Generative Adversarial Network, GAN) (Gulrajani et al., 2017)的思想，提出使用对抗的思想生成回复，除了训练了对话生成模型，还训练了一个分辨人与机器回复的鉴别器模型，用鉴别的结果作为奖励，使得生成器不断朝更加自然的方向生成回复。Shao et al. (2017)在Seq2Seq模型中对生成端使用注意力机制，捕捉生成端的信息，并在模型的解码端使用一种基于分段的随机解码技术。Zhang et al. (2018)提出在Seq2Seq对话模型中使用一个高斯核函数层来指导模型以不同的特殊性生成回复。Xing et al. (2017)在Seq2Seq引入主题模型，并在编码过程中根据主题类别加入主题词，使得生成的对话内容中包含更多的语义信息。Xiong et al. (2016)在基于RNN的Seq2Seq模型的基础上，配合CNN抽取历史对话信息的主题信息特征和主题相关的特征。一些研究者通过加入关键词 (Mou et al., 2016)或使用Beam Search算法 (Vijayakumar et al., 2016)来丰富生成的回复，增加回复结果的合理性和趣味性。

基于上述工作的启发，为了解决生成安全回复的问题，本文提出了基于对话约束的回复生成模型。通过对对话的主题、情感和意图的识别生成更高质量的回复。

3 基于对话约束的回复生成模型

我们的任务旨在自动生成合理自然的对话回复。一组对话由两个对话者之间发起的$m/2$轮对话组成，可表示为对话序列$D = \{U_1, U_2, ..., U_m\}$，其中$U_x(x = 1, 2...)$称为对话的子句。对话生成模型的目的是在第$m/2$轮时，根据前面的$m-1$个子句$\{U_1, U_2, ..., U_{m-1}\}$计算在此情况下生成句子$U_m$的概率，即$P(U_m|U_1, U_2, ..., U_{m-1})$。每个句子$U_m$是可变长的单词序列，可表示为$U_m = \{w_{m,1}, w_{m,2}, ..., w_{m,N_m}\}$。其中$w_{m,n}$表示第m个句子中的第n个单词，$N_m$表示$U_m$中的单词个数。通过前m-1个子句和当前已经生成的单词来逐字预测下一个词，直到达到特定的句子长度或者生成结束符，预测结束，得到回复U_m。$P(U_m|U_1, U_2, ..., U_{m-1})$可表示为如下公式：

$$P(w_{m,1}, w_{m,2}, ..., w_{m,n}|U_1, U_2, ..., U_{m-1}) = \prod_{n=1}^{N_m} P(w_{m,n}|U_1, U_2, .., U_{m-1}, w_{<n}) \quad (1)$$

3.1 模型

本文采用的是基于LSTM网络的Seq2Seq模型。该模型包括3个部分:编码器(Encoder) 网络、解码器(Decoder)网络和连接Encoder-Decoder的中间语义向量C。编码器网络和解码器网络分别对应输入序列和输出序列的两个神经网络，本实验中编码器和解码器是基于LSTM网络处理输入序列和输出序列的。输入序列通过编码器对其进行编码，形成一个中间语义向量C，

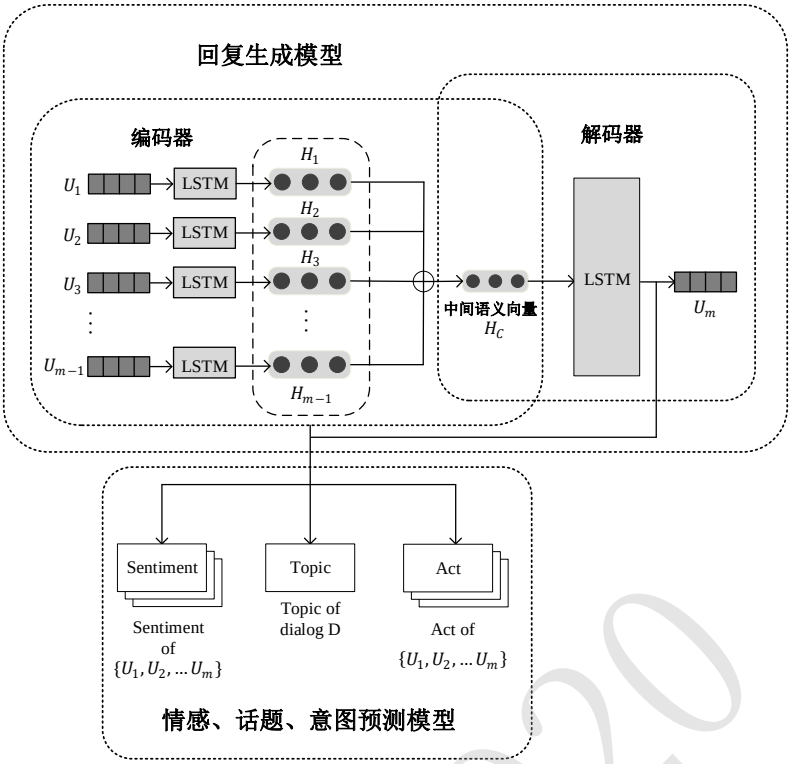


图 1. 模型网络结构图

把这个向量传递给解码器，通过中间语义向量C和递归隐藏状态，计算词汇表中每个词的概率分布，生成下一个词，并将其作为下一个时刻的输入，直至遇到结束符标志或达到指定的长度时结束解码。另外我们的模型(JTEA)在模型训练阶段，从编码器和解码器的LSTM网络中学习对话历史对话信息中每个子句和生成回复的特征表示。这些特征向量用于预测每个话语的情感和意图以及整段对话的主题。主题预测、情感预测和意图预测作为对话回复生成的辅助任务，只在模型训练阶段进行，帮助学习编码器和解码器中的共享参数，对生成回复的主题、情感和意图进行约束。模型网络结构图如图 1所示。

3.1.1 编码器

编码器是将输入序列编码成一个中间语义向量C。本实验中使每个子句分别经过LSTM神经网络，得到每个子句的特征向量，然后取这些特征向量的平均值，作为中间语义向量C。我们可以在两个层次上讨论话语序列：每组对话是由子句序列组成，而每个子句由单词序列组成。基于此，对于给定句子$U_m = \{w_{m,1}, w_{m,2}, ..., w_{m,N_m}\}$，经过LSTM网络，将最后一层的隐藏层的输出$h_{N_m}$作为该子句的语义向量。本文为历史信息中的每个子句分别构建一个LSTM模型，取所有LSTM模型输出的平均值作为整组对话的中间语义向量C。

3.1.2 解码器

解码器是将编码器中生成的中间语义向量C再转化成输出序列。在解码阶段，我们会用输出序列$\{y_1, y_2, ..., y_{t-1}\}$以及中间语义向量$C$来预测下一个输出的单词$y_t$,即

$$y_t = \text{argmax}\, P(y_t) = \prod_{t=1}^{T} P(y_t|y_1, y_2, ..., y_{t-1}, C) \tag{2}$$

通常情况下，会为解码器指定生成文本最大长度和结束字符。在解码过程中，只要符合上述两个条件之一，解码过程就会结束。解码器的输出并不是文本，而是一个向量，这个向量代表着当前这个神经单元输出对应词表的概率分布，通常选择概率最高的作为输出，但概率最高的往往是“好的”，“是的”等安全回复。因此，我们引入了主题、情感、意图识别帮助模型更好的理解对话内容，生成与对话主题相关且具有合理情感和意图的回复，增加生成回复的多样性。

3.1.3 主题预测模型

主题预测任务旨在预测整段对话的主题标签，用于约束生成回复的主题。本实验中取各个子句在编码器和解码器中经过LSTM网络得到的特征向量的平均值$\overline{h}$作为主题预测模型的softmax层的输入。给定特征向量H^t作为softmax层的输入：

$$P_s^t = \text{softmax}(W_s^t H^t + B_s^t) \tag{3}$$

这里W_s^t，B_s^t为模型参数，P_s^t是意图预测模型的输出，用于主题分类。

3.1.4 情感预测模型

情感分类任务旨在预测子句的意图标签，用于约束生成回复的情感倾向。本实验中将每个子句在编码器和解码器中经过LSTM网络得到的特征向量作为情感预测模型的softmax层的输入。给定特征向量H^s作为softmax层的输入：

$$P_s^s = \text{softmax}(W_s^s H^s + B_s^s) \tag{4}$$

这里W_s^s，B_s^s为模型参数，P_s^s是情感预测模型的输出，用于情感分类。

3.1.5 意图预测模型

意图分类任务旨在预测子句的意图标签，用于约束生成回复的意图倾向。本实验中将每个子句在编码器和解码器中经过LSTM网络得到的特征向量作为意图预测模型的softmax层的输入。给定特征向量H^a作为softmax层的输入：

$$P_s^a = \text{softmax}(W_s^a H^a + B_s^a) \tag{5}$$

这里W_s^a，B_s^a为模型参数，P_s^a是意图预测模型的输出，用于意图分类。

4 实验设置与结果分析

4.1 数据集

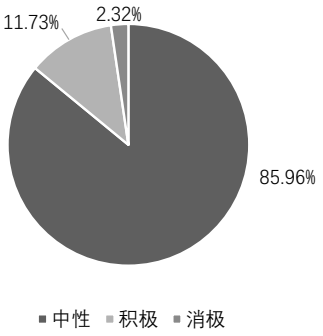
本实验使用DailyDialog对话语料 (Li et al., 2017b)，该语料收集于英语学习网站的对话练习。该语料的基本统计信息如表 2所示，共包含13118个多回合对话，平均每组对话轮数约为8,平均每句对话的单词数约为15，平均每组对话的单词数约为115。该数据集中的对话反映了我们的日常交流方式，涵盖了我们日常生活的各种话题。该语料中的每句话都标注了情感和意图类别，这些标注由3名语言专家共同完成，具有较高的可靠性。其中主题分为10类：校园生活(School Life)、工作(Work)、健康(Health)、日常生活(Ordinary Life)、人际关系(Relationship)、文化与教育(Culture & Education)、政治(Politics)、态度与情感(Attitude & Emotion)、旅游(Tourism)、金融(Finance)；情感分为7类，在本实验中我们为了更好的识别情感，将情感重新分为中性(Neutal)、积极(Positive)、消极(Negative)三类；意图分为4类：陈述(Inform)、询问(Question)、指示(Directive)、许诺(Commissive)。

对话总数	13118
平均每组对话轮数	7.9
平均每个子句的单词数	14.6
平均每组对话的单词数	114.7

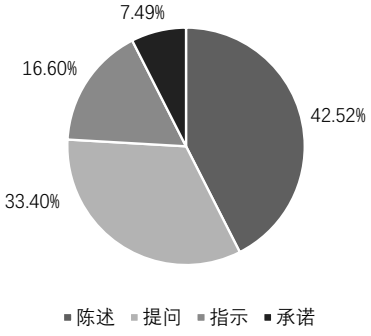
表 2. DailyDialog基本信息统计

本实验中，我们研究四轮对话，因此我们过滤了少于八句的对话，并截取大于等于八句对话中的前八句。在以上条件下，挑选5835组对话作为训练集，200组作为测试集。图 2中分别给出了过滤后的数据集中话题、情感、意图的类别概率分布。从图中我们发现，对话中情感为中性的话语占大多数，意图为陈述和询问的话语占比高于其他两种意图。

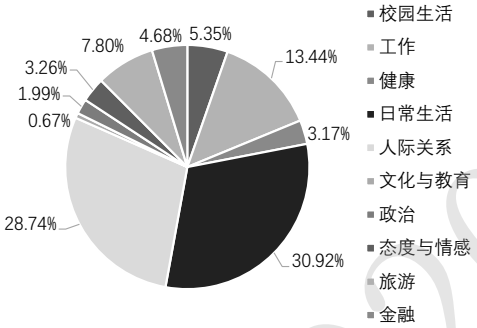
图 3展示了同一轮对话中前一句的意图确定时，后一句话的意图类别的概率分布。从图中我们发现对话中不同的角色之间的意图是相互影响。对话发生在前后两个角色之间，后者的意



(a) 情感类别分布



(b) 意图类别分布



(c) 话题类别分布

图 2. 情感、意图、话题类别概率分布

图常常会受到前者的影响，而前者是对话的发起方，处于主导地位。例如，提问和陈述往往是同时发生的，因为当有人向我们提问时，我们通常不会转移话题，而是礼貌地回复别人的问题。另外指示和承诺往往也是同时发生的，当有人向我们提出建议时，我们一般会对对方所提的建议做出回应。图 4显示了同一说话者前一句的情感确定的条件下，后一句的情感类别的概率分布。为了避免无情感话语的干扰，我们只统计了话语的情感为积极或者消极的情况。从图中我们看出相同情感同时出现的概率远远高于其他情感，这说明同一说话者的情感基调通常是保持不变的。基于上述分析，我们可以发现对对话的情感和意图识别在回复生成中是非常重要的。

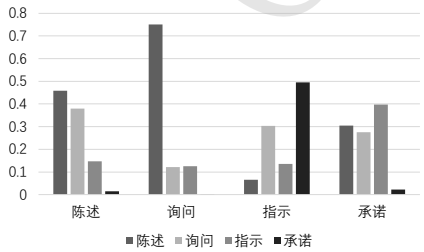


图 3. 同一轮对话中意图的影响

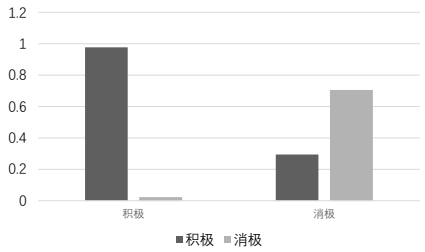


图 4. 同一说话者情感的影响

本文使用BLEU(Bilingual Evaluation Under-study) (Papineni et al., 2002)来衡量所产生的响应与地面真实响应之间的相关性。本文主要使用BLEU-1、BLEU-2和BLEU-3来评测实验效果。本文还使用词嵌入之间的余弦相似度来度量词之间的相似性，可分为三类：Average (Foltz et al., 1998)，Greedy (Rus and Lintean, 2012)和Extrema (Forgues et al., 2014)。Average计算句子级别的嵌入，而Greedy和Extrema则计算单词级别的余弦相似度。它们的区别在于，Greedy将句子中的平均单词向量作为句子的嵌入，而Extrema则采用这些单词向量的极值。

4.2 实验参数设置

为了获取最优模型，经调整本实验设置的相关参数如表 3所示。

参数	取值
Embedding层的输出维度	128
LSTM层的输出维度	128
话语最大长度	20
批次大小	64
迭代次数	20
dropout	0.2

表 3. DailyDialog基本信息统计

4.3 与基准模型比较

在前人关于对话回复生成的研究的基础上，我们选取了Seq2Seq，Multi，Dir-VHRED，ReCoSa，HRG五个模型进行对比实验。接下来，我们将分别介绍这五个模型。

- **Seq2Seq模型：** (Bahdanau et al., 2015)将前七句历史对话信息拼接成一个向量作为输入，为其构建一个LSTM模型，此LSTM模型的输出作为中间语义向量，然后解码生成对话回复。此模型为一个单输入模型。

- **Multi模型：**将前七句话分别作为输入，为每个子句构建LSTM模型，将所有LSTM的输出的平均值作为中间语义向量，然后解码生成对话回复。此模型为一个多输入模型。

- **Dir-VHRED模型：** (Zeng et al., 2019)使用Dirichlet分布来描述VHRED中的潜在变量。Dirichlet分布是贝叶斯统计中多项式分布的流行共轭先验，它可以是凹面或凸面，单调上升。

- **ReCoSa模型：** (Zhang et al., 2019)使用自注意力机制来更新上下文和被屏蔽的响应表示，并在解码过程中使用上下文和响应表示之间的注意力权重。

- **HRG模型：** (Zhang and Zhang, 2019)采用分层响应生成框架以自然和连贯的方式捕获对话意图。

模型名称	BLEU-1	BLEU-2	BLEU-3	Average	Greedy	Extrema
Seq2Seq	12.94	5.64	4.80	57.88	42.01	36.06
Multi	15.64	6.48	5.33	62.14	44.33	37.20
Dir-VHRED	10.90	3.82	2.26	59.92	41.26	32.70
ReCoSa	17.24	8.37	6.89	62.63	46.59	40.42
HRG	17.58	7.55	5.88	63.29	45.34	38.50
JTEA(Ours)	19.41	10.40	9.01	64.56	48.13	41.46

表 4. 与基线模型比较

表 4展示了我们的模型和基线模型的比较结果。从表中我们可以得出以下结论:

1) Multi模型比Seq2Seq模型表现好。因为Seq2Seq模型是将所有历史句子信息拼接作为输入序列，导致前面子句的语义信息被逐渐稀释掉，生成的中间语义向量不能充分提取历史信息中的特征。同时也证明了将历史信息中的每一个子句分别经过LSTM网络，取平均值作为中间语义向量更加合理。

2) 我们的模型相较于Multi模型，BLEU-1、BLEU-2、BLEU-3值分别提升了3.77、4.76、3.68个百分点，Average、Greedy、Extrema分别提升了2.42、3.8、4.26个百分点。而我们的模型是在Multi模型的基础上了引入了主题、情感和意图的识别，这证明了对对话回复进行约束可以有效地提高生成回复的质量。

3) 我们的模型的实验结果超过了所有的基准模型，充分说明了我们提出的模型能有效地提高生成的对话回复的质量。

4.4 不同因素的影响

为了验证我们模型的有效性，将我们的模型与分别单独考虑话题、情感、意图预测模型进行对比实验。实验结果如表 5所示。我们共涉及了5组对比实验。Multi模型在上一节中已经介绍，不再赘述；Joint Topic模型是在Multi模型的基础上加入对话意图的识别；Joint Senti模型是在Multi模型的基础上加入话语情感的识别；Joint Act模型是在Multi模型的基础上加入话语意图的识别。

模型名称	BLEU-1	BLEU-2	BLEU-3	Average	Greedy	Extrema
Multi	15.64	6.48	5.33	62.14	44.33	37.20
Joint Topic	18.01	8.87	7.60	63.34	45.32	39.26
Joint Senti	16.89	7.74	6.34	62.74	43.93	37.32
Joint Act	17.97	8.77	7.46	63.22	45.50	38.84
JTEA(Ours)	19.41	10.40	9.01	64.56	48.13	41.46

表 5. 不同因素的影响

从表中可以发现，所有具有约束条件的模型(Joint Topic, Joint Senti, Joint Act, JTEA)都优于基准Multi模型，这表明所有的约束条件对于回复生成都是有效的。另外，我们的模型即考虑所有的约束条件优于单独考虑每个约束条件的模型，这表明我们应该集成所有约束条件来生成更高质量的回复。我们还可以发现情感的约束相对于主题和意图的约束效果要稍差一点，是因为在我们的语料中绝大部分话语的情感都是中立的，情感对回复的影响相对较小。

4.5 案例分析

我们对基线Multi模型和我们的模型生成的对话回复进行对比分析，表 6给出了三组对话示例，我们截取了对话的主要内容且对话内容都已翻译为中文展示。

从第一组示例中我们可以看出，此轮对话的主题是校园生活。对于最后说话者A的提问，Multi模型没能充分的理解该组对话的语义信息，所以生成了“你可以确定”这样和对话内容毫无联系的回复。而我们的模型因为加入了对主题的识别，所以生成了与对话主题高度相关且自然合理的回复。在第二组对话中，我们发现最后说话者A的意图是“指示”。在同一轮对话中，当前一句的意图为“指示”时，后一句的意图最大概率为“承诺”。我们的模型因为加入了对意图的识别，生成了意图为“承诺”的回答，而multi模型则生成了意图为“陈述”的回答，显然我们生成的回复更加的合理。在第三组实例中，我们的模型生成的回复在意图上显然更加合理。另外，根据4.1小节中的分析，同一说话者的情感基调往往是不变的。我们的模型因为加入了对情感的识别，生成了带有积极情感倾向的回复。

5 总结

在大数据不断发展的今天，人机对话系统是人机交互领域一个非常重要的研究方向，开放域聊天机器人的研究受到了广泛关注。这使得提高对话回复的质量尤为重要。

考虑到在对话往往是围绕特定的主题且对话中的每个子句具有明显的情感和意图倾向，本文提出了一种基于对话约束的回复生成模型。以往的对话生成模型总是倾向于生成类似“好的”、“我不知道”的安全回复，虽然这样的回复看起来合理，但其实这样的模糊回复意义并不大，信息量很低。为了解决这个问题，我们在基于LSTM网络的Seq2Seq模型基础上引入了对对话的话题、情感和意图的识别，使得生成具有合理的情感和意

示例1	
A	我如何申请美国的大学?
B	您应该去图书馆查找有关美国大学的信息， 然后写信给招生办公室。（中性，指示）
A	处理所有程序需要多长时间? （中性，提问）
Multi生成的回复	你可以确定。（中性，陈述）
JTEA(Ours)模型生成的回复	如果一切顺利，需要3到6个月。（中性，陈述）
示例2	
A	嗨，米克尔。你怎么了? 你看起来很生气。（中性，询问）
B	不，我只是检查我的体重，我越来越胖。（消极，陈述）
A	你为什么不和我一起锻炼呢? 如果坚持下去，它会起作用。 来吧，走吧！（中性，指示）
Multi生成的回复	是的，我有一杯酒。（中性，陈述）
JTEA(Ours)模型生成的回复	好的，请帮助我更好的完成它。（中性，承诺）
示例3	
A	您好，我是ABC公司的露西。我打电话看你是否做出决定。 您是我们正在寻找的合适人选。（中性，提问）
B	是的，我决定与您的公司合作。我的荣幸。（积极，陈述）
A	好的，请下个星期一上午9点到我的办公室。（中性，指示）
Multi生成的回复	那其他男人呢? （中性，提问）
JTEA(Ours)模型生成的回复	没问题。谢谢你给我打电话。（积极，承诺）

表 6. 生成回复示例对比

图倾向且与对话主题相关的回复。实验结果表明：该方法与传统的Seq2Seq方法相比，在BLEU值、Average、Greedy、Exrema的评测上有了较大的提升。

由于本文对话数据仅限于标准的书面日常对话，有一定的局限性。因此，本文的下一步工作将进一步考虑更加专业的，更加贴近实际的对话文本。另外，研究历史信息中各个子句之间的相互影响，对对话中的句子结构进行建模，进一步提升情感、意图识别的效果。

参考文献

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Peter F. Brown, Stephen Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Comput. Linguistics*, 19(2):263–311.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1724–1734. ACL.

Peter W Foltz, Walter Kintsch, and Thomas K Landauer. 1998. The measurement of textual coherence with latent semantic analysis. *Discourse processes*, 25(2-3):285–307.

Gabriel Forgues, Joelle Pineau, Jean-Marie Larchevêque, and Réal Tremblay. 2014. Bootstrapping dialog systems with word embeddings. In *Nips, modern machine learning and natural language processing workshop*, volume 2.

Daniel Fried, Mitchell Stern, and Dan Klein. 2017. Improving neural parsing by disentangling model combination and reranking effects. *arXiv preprint arXiv:1707.03058*.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pages 5767–5777.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991.

Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, and Richard Socher. 2016. Ask me anything: Dynamic memory networks for natural language processing. In *International conference on machine learning*, pages 1378–1387.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature*, 521(7553):436–444.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In Kevin Knight, Ani Nenkova, and Owen Rambow, editors, *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 110–119. The Association for Computational Linguistics.

Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017a. Adversarial learning for neural dialogue generation. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2157–2169. Association for Computational Linguistics.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017b. Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv preprint arXiv:1710.03957*.

Brendan D. McKay and Adolfo Piperno. 2014. Practical graph isomorphism, II. *J. Symb. Comput.*, 60:94–112.

Lili Mou, Yiping Song, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. 2016. Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. In Nicoletta Calzolari, Yuji Matsumoto, and Rashmi Prasad, editors, *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 3349–3358. ACL.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Vasile Rus and Mihai Lintean. 2012. An optimal assessment of natural language student input using word-to-word similarity metrics. In *International Conference on Intelligent Tutoring Systems*, pages 675–676. Springer.

Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Louis Shao, Stephan Gouws, Denny Britz, Anna Goldie, Brian Strope, and Ray Kurzweil. 2017. Generating high-quality and informative conversation responses with sequence-to-sequence models. *arXiv preprint arXiv:1701.03185*.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. *arXiv preprint arXiv:1506.06714*.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Chongyang Tao, Shen Gao, Mingyue Shang, Wei Wu, Dongyan Zhao, and Rui Yan. 2018. Get the point of my utterance! learning towards effective responses with multi-head attention mechanism. In *IJCAI*, pages 4418–4424.

Ashwin K. Vijayakumar, Michael Cogswell, Ramprasaath R. Selvaraju, Qing Sun, Stefan Lee, David J. Crandall, and Dhruv Batra. 2016. Diverse beam search: Decoding diverse solutions from neural sequence models. *CoRR*, abs/1610.02424.

Oriol Vinyals and Quoc V. Le. 2015. A neural conversational model. *CoRR*, abs/1506.05869.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *arXiv preprint arXiv:1508.01745*.

Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. 2017. Topic aware neural response generation. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Kun Xiong, Anqi Cui, Zefeng Zhang, and Ming Li. 2016. Neural contextual conversation learning with labeled question-answering pairs. *arXiv preprint arXiv:1607.05809*.

Min Zeng, Yisen Wang, and Yuan Luo. 2019. Dirichlet latent variable hierarchical recurrent encoder-decoder in dialogue generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1267–1272.

Bo Zhang and Xiaoming Zhang. 2019. Hierarchy response learning for neural conversation generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1772–1781.

Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, Jun Xu, and Xueqi Cheng. 2018. Learning to control the specificity in neural response generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1108–1117.

Hainan Zhang, Yanyan Lan, Liang Pang, Jiafeng Guo, and Xueqi Cheng. 2019. Recosa: Detecting the relevant contexts with self-attention for multi-turn dialogue generation. *arXiv preprint arXiv:1907.05339*.

多模块联合的阅读理解候选句抽取*

吉宇[1,‡] 王笑月[1,‡] 李茹[1,2,*†]郭少茹[1,‡] 关勇[1,‡]
[1]山西大学　计算机与信息技术学院
[2]山西大学　计算智能与中文信息处理教育部重点实验室
[‡]{jiyu0515, wangxy0808,guoshaoru0928,guanyong0130}@163.com
[*]liru@sxu.edu.cn

摘要

机器阅读理解作为自然语言理解的关键任务，受到国内外学者广泛关注。针对多项选择型阅读理解中无线索标注且涉及多步推理致使候选句抽取困难的问题，本文提出一种基于多模块联合的候选句抽取模型。首先采用部分标注数据微调预训练模型；其次通过TF-IDF递归式抽取多跳推理问题中的候选句；最后结合无监督方式进一步筛选模型预测结果降低冗余性。本文在高考语文选择题及RACE数据集上进行验证，在候选句抽取中，本文方法相比于最优基线模型F1值提升3.44%，在下游答题任务中采用候选句作为模型输入较全文输入时准确率分别提高3.68%和3.6%，上述结果证实本文所提方法有效性。

关键词： 机器阅读理解 ； 候选句抽取 ； 递归抽取

Evidence sentence extraction for reading comprehension based on multi-module

Yu Ji[1,‡] Xiaoyue Wang[1,‡] Ru Li[1,2,*] Shaoru Guo[1,‡] Yong Guan [1,‡]
[1]School of Computer and Information Technology，Shanxi University
[2]Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education,Shanxi University,Taiyuan,China
[‡]{jiyu0515, wangxy0808,guoshaoru0928,guanyong0130}@163.com
[*]liru@sxu.edu.cn

Abstract

As a key task of natural language understanding, machine reading comprehension has been widely concerned by scholars at domestic and foreign. In order to solve the problem of multiple choice reading comprehension, which is difficult to extract evidence sentences due to the absence of clue annotation and questions involve multi-hop reasoning, we proposes a model of evidence sentence extraction based on multi-module combination. Firstly,we use some labeled data to fine-tune the pre-training model; secondly, the evidence sentences in the multi-hop reasoning problem are extracted recursively through TF-IDF; finally, the unsupervised method is combined to further filter the model prediction results to reduce redundancy. This paper is verified on the Chinese Gaokao and the RACE data set. In the extraction of evidence sentences, compared with the optimal baseline model, the F1 value of the method in this paper is increased by 3.44%. The accuracy of using evidence sentences as model input in

* 基金项目：国家重点研发计划重点专项(No.2018YFB1005103);国家自然科学基金(No.61772324)
† 通讯作者.

downstream answer tasks is 3.68% and 3.6% respectively higher than that of full text input. The above results confirm the effectiveness of the proposed method.

Keywords: Machine reading comprehension , Evidence sentence extraction , Recursive extraction

1 引言

随着自然语言处理技术的发展，国内外对于机器阅读理解的研究不断深入。本文重点关注机器阅读理解中的多项选择题任务(Mostafazadeh et al., 2016; Lai et al., 2017)，即：给定文章、问题和选项，要求根据文章回答问题，从多个选项中选择最佳选项。

对于该任务，研究者通常将整篇文章、问题及选项作为输入(Wang et al., 2018; Ran et al., 2019)并在三者之间两两交互，进行信息整合继而选出最佳选项。然与片段抽取式阅读理解不同，多项选择的答案难以直接从给定的文章中提取(Wang et al., 2019)，（如在RACE(Lai et al., 2017)中87%的问题不能直接从文章中找到答案），且并非全文都与当前选项有关。若将全文信息编码将引入噪声从而影响模型预测。特别对于文章较长的数据集（如高考语文阅读理解多项选择题，其平均长度为1,134.15字/篇），输入全文将导致模型很难提取出与问题及选项相关的“重要信息”，且答题缺乏可解释性。因此，为提取问题所需的“重要信息”并提升模型性能，本文针对多项选择阅读理解任务中的候选句抽取问题进行研究。

针对以上问题，Zhang et al. (2019)提出DCMN+，在对文章编码前加入候选句筛选工作，利用余弦相似度评估文章与选项间关联程度，以此缩短文章范围。但多项选择题的候选句通常为多句(Wang et al., 2019)，存在某些候选句与选项之间重叠度较低的情况，如图1所示，结合问题可得出S24包含判断选项A正误的关键信息，但从表面看S24与选项A关联度较低。若通过余弦相似度、模式匹配的方式查找，该类候选句很难抽出且会对后续答题造成影响。鉴于此，Trivedi et al. (2019)将候选句判断视为文本蕴含任务(Korman et al., 2018)，文章中句子视为前提、选项视为假设，判断两者之间是否存在蕴含关系。由于缺少标注数据，采用SNLI(Bowman et al., 2015)对模型进行微调之后进行预测。但该种方法未考虑候选句之间信息冗余的情况，如图1所示，通过语义相似度计算，可在文中抽取出与选项A的关联句S1与S15,但这两句所含的信息相同，对答案预测并无提升作用且增加了计算量。对此，Yadav et al. (2019)提出ROCC，从候选句集对选项及问题的信息覆盖度、候选句与选项及问题之间的语义相关性以及候选句之间冗余性三方面计算ROCC得分，进一步筛选抽取结果。在一定程度上缓解冗余现象，有效提高了候选句抽取的精确性。

上述方法的提出虽使模型性能获得巨大提升，但仍存在一些挑战。（1）直接将选项与问题拼接，忽视其拼接结果是否为一个完整的陈述句或存在语法错误，会对模型句意理解造成影响，如图1所示；（2）通常数据集中候选句在全文所占比例较小而无关信息占比较大，存在正负样本不均衡的情况；（3）对于需要多步推理的问题（即：A推B，B推C），判断选项正误与否的候选句与选项之间并不存在直接关联，需寻找选项候选句的候选句。

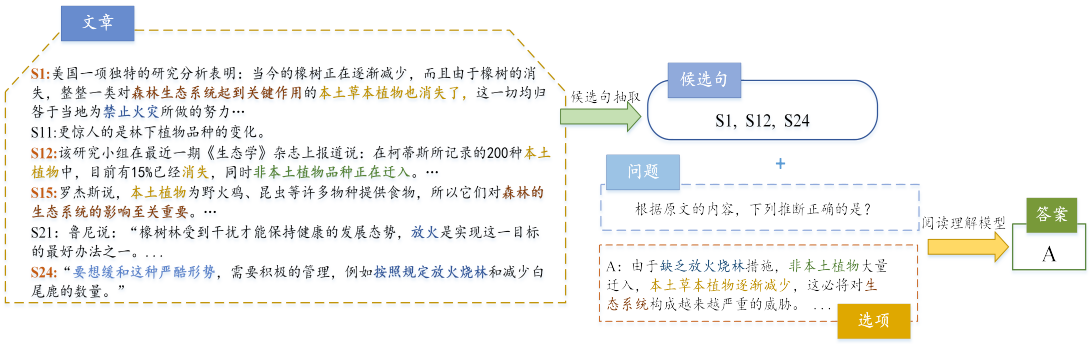


图 1. 高考阅读理解选择题候选句示例

面对上述挑战，本文在RACE和高考语文数据集（见5.2节）上进行实验。通过对数据集研究发现，文章候选句与选项之间的关联度较低。若仅用句对间关联较为明显的SNLI等数据集

训练或采用无监督方式，都无法较准确完整地将候选句抽出。故本文人工标注部分候选句对。考虑到选项信息不完整对候选句抽取的影响，本文将所有问题与选项拼接改写确保其不含语法错误；之后，使用构造的数据集对BERT(Devlin et al., 2018)进行微调；针对正负样本不均衡现象，采用FocalLoss(Lin et al., 2017)作为损失函数，在训练时推动模型更加关注于困难样本，降低简单负例的学习度，从而在整体上提高候选句抽取的F1值，基于此得到初步候选句集；对于多步推理问题导致候选句难以直接抽取的现象，本文提出基于TF-IDF的递归式抽取方法，进一步提升模型召回率；为保证候选句抽取结果的精确性，减少候选句之间的冗余，采用ROCC(Yadav et al., 2019)过滤重复信息，提升精确率。

为进一步评估候选句抽取质量及所提方法对后续答题的帮助，本文将抽取出的候选句集拼接，采用BERT与co-matching模型分别在RACE、高考语文阅读理解选择题数据集上进行实验，实验结果表明采用候选句集作为输入相比全文在高考及RACE数据集上分别提升了3.68%及3.6%。在候选句抽取上，本文所提方案相比于基线F1值进一步提升了3.44%及3.95%。

2 相关工作

候选句抽取工作，依据训练方式可划分为四种类型。（1）使用无监督方法为候选句抽取提供了指导，同时减省人工标注的消耗(Yadav et al., 2019)；（2）有监督方法通过标注数据训练模型，从而实现下游任务中自动抽取候选句的目的。Trivedi et al. (2019)使用文本蕴含语料(Bowman et al., 2015; Williams et al., 2018)为训练候选句抽取模型。对于不提供候选句标注的数据集，研究者从结构化知识库(Speer et al., 2016)中选取相关线索知识，训练模型(Hao et al., 2017; Lukovnikov et al., 2017)；（3）使用信息检索进行候选句抽取工作，通过强化学习(Geva and Berant, 2018)或pagerank(Surdeanu et al., 2008)学习如何在缺少明确训练数据的前提下进行候选句抽取。或是使用注意力机制在文本与选项及问题之间交互，使文章中与选项和问题相关部分的注意力权重更大(Ran et al., 2019; Tang et al., 2019)；（4）通过人工定义规则，抽取出含有噪声信息的候选句，使用弱监督方式训练模型(Min et al., 2018)。上述工作，各有其贡献之处与意义，推动了模型在相应下游任务上的性能表现。本文所提工作着重在对上述工作疏漏之处进行强化，综合使用有监督与无监督方式，使抽取结果可评价并且提高抽取结果的精确性也减省了数据标注工作量。同时，对上述模型中未能考虑到的选项信息缺失问题以及正负样本不均衡也进行了相应处理。此外，本文针对多步推理问题提出了一种多步信息抽取方式，进一步提升了模型抽取效果。并在下游任务中验证了模型的有效性。

3 候选句抽取模型

本文提出一种新的候选句抽取模型，模型整体架构如图2所示。其主要包含四部分：（1）选项改写模块：融合选项与问题所涵盖的信息，确保其结果无语法错误；（2）候选句抽取模块：从文章中初步筛选出与判断选项正误有关的句子集合；（3）TF-IDF递归抽取模块：在前一步的基础上，使用TF-IDF作为引导，抽取多步推理问题候选句，避免关键信息遗漏；（4）筛选模块：在所得句子集合上进一步筛选，提高候选句抽取精确率，降低信息冗余。

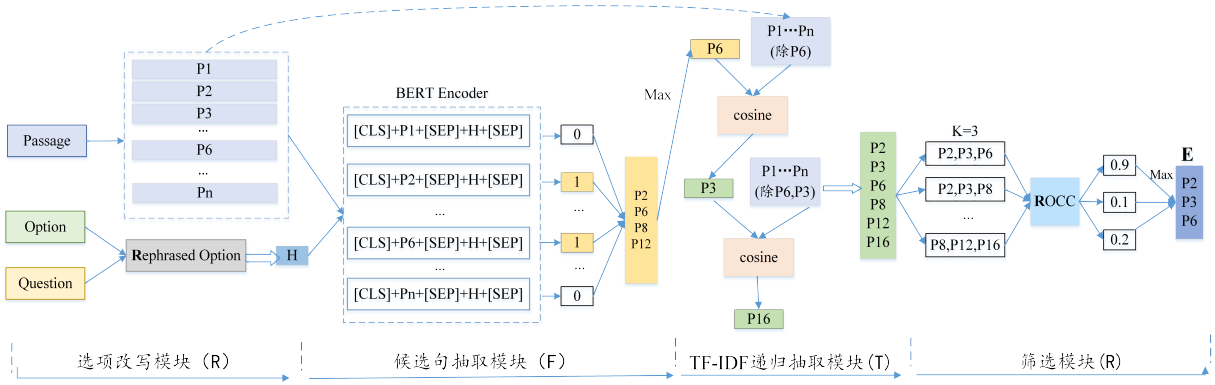


图 2. 候选句抽取模型

3.1 选项改写模块

通过对高考阅读理解及RACE数据集分析后发现，如图3所示，当问题为“下列说法符合（不符合）文意的一项是？（或其同义表述）,该类问题所蕴含的信息量较少，选项信息完整，无需对选项改写；而当问题为“下列对‘国外媒体关注点’的理解，不正确（正确）的一项是？”，选项内容为“科技竞争力”，若仅使用选项内容，其涵盖信息量过少，抽取对应候选句会较为困难；而若将问题与选项直接拼接，所得结果不符合语法规则。故需提取问题的关键信息，并将其与选项信息融合，形成一条完整的句子。

针对上述两种情况，首先采用正则表达式进行选项内容改写，使其形成完整陈述句$H = \{H_1, H_2, \ldots, H_m\}$其中$m$为选项改写句的长度；之后将文章切分为句子$P = \{P_1, P_2, \ldots, P_n\}$其中$P$为文章，$n$为文章中句子数量。

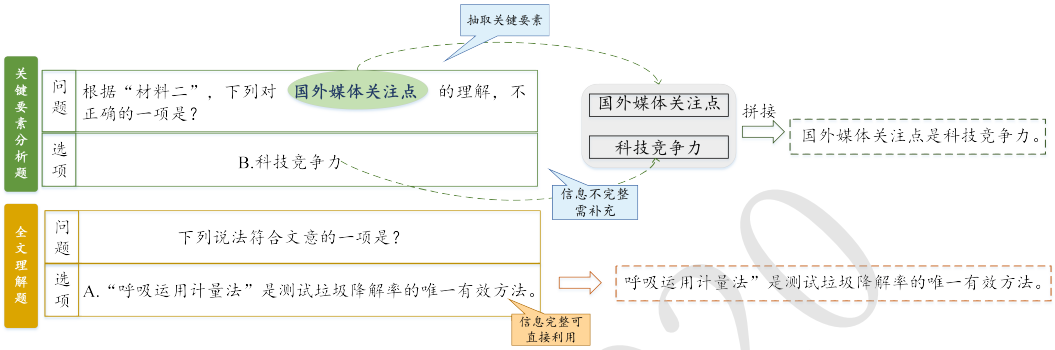


图 3. 选项改写示例

3.2 候选句抽取模块

该模块通过计算P_i与选项改写句H的关联度，初步抽取出候选句。本文在BERT基础上进行实验，首先将$[CLS]$,句子P_i,$[SEP]$,选项改写H,及$[SEP]$拼接后输入模型中，其中$[SEP]$为BERT中的片段分隔符，$[CLS]$为特殊字符（输入整体表示）。编码后，取$[CLS]$的编码结果$O_i \in R^d$进行分类，d为BERT隐藏层维度。

3.2.1 Focal Loss

由于候选句数据集中存在正负样本不均衡现象（RACE候选句数据集中，正负样本比为1：10）。本文采用FocalLoss作为损失函数，使模型聚焦于正样本的学习,缓解样本类别不均衡带来的风险。

输入的候选句对为(P_i, H),模型预测结果为P=[p_0,p_1],真值为Y=[y_0,y_1]。对于传统的交叉熵损失而言，其表示为：$CE = -(y_0 \log(p_0) + y_1 \log(p_1))$,显然，当负样本占比较大时，模型的训练会被负样本占据，使得模型难以从正样本中学习。

$$L_{fl} = \begin{cases} -\alpha(1-y')\log'_y & y = 1 \\ -(1-\alpha)y^{\gamma}\log_{1-y'} & y = 0 \end{cases} \tag{1}$$

FocalLoss在原有的基础加入权重系数γ及α,γ减少易分类样本的损失使模型更关注于困难的、错分的样本；α用于平衡正负样本本身数量比例不均,由此缓解了正负样本不均衡的现象。

3.3 TF-IDF递归抽取模块

由于阅读理解多项选择题中存在多步推理问题，如图1所示，该情况难以直接使用文本蕴含方式将选项对应候选句全部抽出。考虑到多步推理问题中存在链式关系，故基于上一步所得结果$E = \{E\}$，首先选出与选项改写句关联度最高的句子作为第一跳候选句$hop1$。继而，计算其与文章句子（除本身之外）的相似度，取相似度最高的作为第二跳候选句$hop2$。之后，计算文章句子中与$hop2$之间的关联度（$hop1$与$hop2$除外），并取关联度最高句子作为第三跳候选句，以此类推，重复K次（K值视具体情况设定）。将所得的句子与候选句集合E合并。

3.4 候选句筛选模块

为提升抽取结果精确性，降低无关及冗余信息比重。本文使用ROCC对结果进一步筛选。首先，使用上一步的抽取结果E并对其进行全组合$\binom{n}{m}$，生成候选句集合G，其中n为抽取结果的总共句子数，m为组合单位（可依据具体情况调节大小）。之后，对每组集合分别从(1)集合内部的信息冗余度(2)集合对选项的信息覆盖率(3)集合与选项之间信息相关性三个角度计算得分。

3.4.1 冗余度

通过计算给定集合中句对间信息重合度，来确保候选句的多样性和信息互补性。得分越低的句子集合，信息冗余度越低。

$$O(G) = \frac{\sum_{g_i \in G} \sum_{g_j \in G} \frac{|t|g_i| \cap t|g_j||}{\max(t|g_i|, t|g_j|)}}{\binom{|G|}{2}} \tag{2}$$

其中G表示给定句子集合，g_i与g_j分别表示集合中的某一条句子，$t(g_i)$表示g_i所包含的词集合（去重后），$|t|g_i| \cap t|g_j||$表示g_i与g_j的共有词数量。

3.4.2 覆盖率

该模块用于衡量给定集合G对选项改写句H的词汇覆盖率,由H与集合G之间的共有词的IDF值加权平均得到。Coverage值越大，意味该集合包含选项改写句的信息越多。

$$C_t(H) = \bigcup_{g_i \in G} t(H) \cap t(g_i) \tag{3}$$

$$C(H) = \frac{\sum_{t=1}^{|C_t(H)|} IDF\left[C_t(H)\left[t\right]\right]}{|t\left(H\right)|} \tag{4}$$

其中$C_t(H)$表示选项改写句与集合之间的共有词。

3.4.3 相关性

使用BM25(Robertson and Zaragoza, 2009)计算给定集合G与选项改写句H的相关度。计算公式如下：

$$R(H, G) = \sum_{i}^{n} w_i \cdot R(h_i, G) \tag{5}$$

从上述三个角度分别计算出给定集合得分后，综合得分计算集合的ROCC值。

$$S(G) = \frac{R}{\varepsilon + O(G)} \cdot R(\varepsilon + C(H)) \tag{6}$$

如式6中所示，R为集合与选项改写句的relevance得分，O为集合的overlap值，$C(H)$为集合对选项改写句的coverage值，为避免计算中出现分子或分母为0的情况，添加ε作为平滑项，实验中设ε值为1。之后，选ROCC得分最大集合作为最终的候选句集合E_2。

4 答题模型

得到候选句集合后，将其句子拼接为文章C。之后同问题Q，选项O_i一起作为答题模型的输入。

$$A_i = f(C, Q, O_i) \tag{7}$$

$$L(A_t|C,Q) = -\log \frac{\exp(W^T \cdot A_t)}{\sum_{j=1}^{m} \exp(W^T \cdot A_j)} \tag{8}$$

式7中，$f(.)$表示模型编码过程，所得$A_i \in R^d$为文章，问题，选项的最终表示，其中d为模型维度。式8中，$W \in R^{d\times 4}$ 为参数矩阵，A_t为问题的正确选项。

5 数据集

5.1 候选句数据集

高考候选句数据集：由于缺少中文阅读理解候选句语料，本文从数据集中随机抽取500道题,对每个选项人工标注其候选句。标注规则为：对应每个选项，文章中与判断其正误有关句子标注为1，反之，标注为0。为确保数据标注质量，本文采取交叉验证的标注方式：将数据二等分，由四个同学两两一组进行标注，各组内同学标注的数据相同。标注后两组同学交换进行两轮校验，针对标注结果中不一致数据，由仲裁者仲裁进行第三轮校验，剔除无法确定的数据，若无异议，经三轮验证后，将所得标注结果确定为最终候选句集合，包含45，311句对。其中训练集，验证集，测试集包含数据量分别为：36,254，4,528，4,529。

RACE候选句数据集：本文采用Wang et al. (2019)标注的500道RACE mid-challenge部分候选句对，共34，736句对，其中训练集，验证集，测试集分别为27,790，3,473，3,473 。由于初，高中试题难度有所区别，在验证候选句抽取对答题的影响时，本文仅使用RACE数据集中的初中部分进行测试。

5.2 阅读理解多项选择题数据集

本文采用RACE数据集中mid-challenge部分进行实验，共收集18，364道问题，按8：1：1方式将数据划分给训练、验证、和测试集；此外本文同时收集了2005-2019年高考语文阅读理解选择题共7886道，与RACE采用同样方式划分。

6 实验设计与结果分析

6.1 模型评价指标

候选句抽取评价指标： 实验采用F1值、P(精确率)、R(召回率)来评估候选句抽取效果，计算公式如下：

$$P = \frac{TP}{TP+FP} \times 100\% \quad P = \frac{TP}{TP+FN} \times 100\% \quad F1 = \frac{2\times P\times R}{P+R} \times 100\% \tag{9}$$

答题模型评价指标:对于答题部分，采用accuracy作为模型性能评价指标。

6.2 参数设置

针对不同数据集的实验参数设置如表1所示。

实验	DataSet	epoch	max_length	batch	learning rate	K	m
候选句抽取	高考	3	128	32	3e-5	3	4
	RACE	3	128	32	3e-5	3	2
答题模型	高考	6	450	40	1e-5	—	—
	RACE	3	320	32	5e-5	—	—

表 1. 模型参数设置

6.3 实验结果与分析

6.3.1 基线模型性能比较

表2中展示各模型在高考及RACE候选句数据集上的效果，对于高考候选句数据集，从表中可看出BERTwwm的P值，R值及F1值均高于BERT-base；ALBERT-base抽取候选句虽P值较高，但R值相比于BERTwwm低17.71个百分点。故针对高考数据集，本文以BERTwwm为基

础进行改进，结果表明结合RFTR(本文方法)后，模型效果在P值上提升5.41个百分点，R值提升1.99个百分点，F1值3.44个百分点。对于RACE数据集基线模型中BERT-wwm取得最优效果，故在此基础上结合RFTR后，效果提升了3.95个百分点。以上所述验证了所提方法的优越性。

	高考			RACE		
Baseline Model	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
ALBERT-base	81.50	46.13	58.91	76.34	59.08	66.61
BERT-base	73.59	61.84	67.21	77.30	61.10	68.25
BERT-wwm	73.78	63.84	68.45	78.03	60.78	68.33
BERT-wwm+OR	77.29	65.34	70.81	83.10	63.77	72.16
BERT-wwm+OR+FL	76.94	65.81	70.94	82.58	64.10	72.18
RFTR-ROCC	76.86	65.87	70.94	81.40	64.85	72.19
RFTR	79.19	65.83	71.89	84.36	63.23	72.28

表 2. 基于高考语文和RACE的候选句抽取结果

6.3.2 候选句抽取消融实验

为进一步研究所提方案对实验结果的影响，在高考及RACE候选句数据集上分别进行了消融实验。如表2所示,改写选项后（即表中OR）在两数据集上模型F1值相比基线分别提升2.36及3.83个百分点，并且P值和R值也均有提升，表明改写选项使信息更完整，语义更通顺，这对模型的语义学习有很大帮助；之后针对数据集中正负样本不均衡现象，使用Focal Loss进一步使模型效果在F1值上分别提升0.13和0.02个百分点，其中对于高考R值提升0.47个百分点，表明更换损失函数后，模型对正样本学习的偏向性增强；使用TF-IDF相似度计算抽取需多步推理问题的候选句使模型召回率分别提升了0.06和0.75，表明该方案可以有效缓解多步推理问题的信息损失；最后，使用ROCC从候选句集之间的冗余度，候选句集对选项信息覆盖率和候选句与选项相关性三方面考虑，进一步对结果进行筛选，在两数据集上P值分别提升2.33个百分点和2.96个百分点。

6.3.3 候选句抽取效果验证

本文在高考阅读理解选择题与RACE数据集上进行验证，将抽出的候选句拼接作为新文章输入模型，效果如表3所示，其中EV(RFTR)表示使用候选句作为文章的方法，该实验结果证明了候选句抽取的有效性。

	高考		**RACE**	
模型	**dev(%)**	**test(%)**	**dev(%)**	**test(%)**
BERT(base)	32.61	30.33	55.91	57.66
BERT+EV(RFTR)	36.29	31.34	59.51	59.87
co-matching	35.27	32.36	47.54	42.22
co-matching+EV(RFTR)	36.29	35.39	50.65	46.40
ALBERT	29.06	28.05	65.26	67.08
ALBERT+EV(RFTR)	32.11	29.57	68.92	69.30
DCMN	30.83	30.24	48.16	49.53
DCMN+EV(RFTR)	32.64	32.12	50.53	52.19

表 3. 高考语文与RACE数据集答题模型对比结果

6.3.4 K,m对候选句抽取实验结果的影响

为比较实验中TF-IDF递归抽取模块(见3.3节)中K值及候选句筛选模块（见3.4节）全组合中m值对候选句抽取的影响，本文进行了参数对比实验。实验结果如图4,5所示：

由图4可知，在高考数据及RACE数据集中，随着跳数的增加，候选句的召回率逐渐提高，当K为3时召回率与F1值达到最优，表明当跳数为3时可有效缓解多步推理问题的信息损

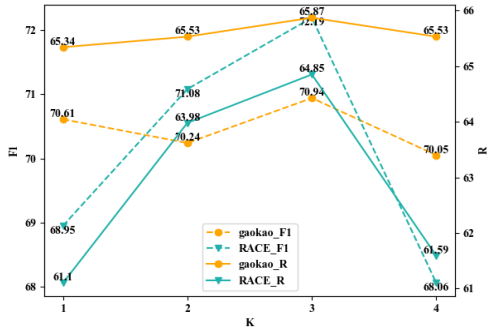


图 4. 递归抽取中K值变化

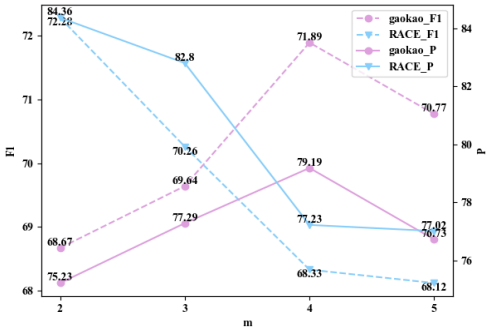


图 5. 筛选模块中m值变化

失；而当跳数为4时召回率下降，说明跳数过多也会引入一定的噪声。由图5可知，在高考数据及RACE中m值分别为4和2时精确率达到最优，说明ROCC可有效筛选冗余信息，最大限度地整合相关信息,从而剔除一部分无关句或冗余句。

6.3.5 错误抽取示例分析

本文选取了测试集中50条错误数据进行了分析，表4为列举的错误数据。

候选句	选项	预测结果	真实结果
蜉蝣这种生物大多数时间生活在水里，以藻类为食，当它们准备好繁殖，便爬出水面，在水边的植物上蜕皮，成为有翅的成虫。	蜉蝣有翅后即升空飞行。虽然飞行时间不长，但由此实现了生命的延续。	0	1
无论我们如何看待鲁迅，如何评价鲁迅先生的毕生之问和他为此所做的一切，现在，我们都依然得和他一起,承受一个各人心底诚信与爱都尚有不足的时代。	鲁迅的时代过去了，但那个时代的国民劣根性今天依然存在，为此我们要呼唤鲁迅，不要漠视鲁迅的存在。	0	1
从印刷的基本需求来看，排字机的字库通常要收7000多字。而从一般书报的需求来说，字体就有书版宋、报版宋、标题宋、仿宋、楷体、黑体...等十多种。	因字形字体的制约，汉字排版繁复。	0	1

表 4. 候选句抽取错误示例

由表可知，错误原因主要有以下三点：（1）指代问题，需辨别表中的“它们”指代“蜉蝣”，才可知“繁殖”与“生命延续”蕴含。（2）归纳概括问题：如“我们都依然得和他一起，承受一个各人心底的诚与爱都尚有不足的时代。”尚有不足的言外之意是：“但那个时代的国民劣根性今天依然存在”，然其表述差异性较大，导致计算机无法“理解”。（3）涉及归纳与知识融合：如需使模型知道“书版宋、报版宋、标题宋、仿宋等”即为“字形字体”。

7 结论与展望

本文针对多项选择阅读理解候选句抽取任务，以有监督方式抽取为基础，针对选项语义不完整、数据集正负样本不均衡、及抽取结果信息冗余等方面进行改进。在高考及RACE数据集上进行实验，证实了该方法的有效性。同时，还验证了候选句抽取对多项选择答案预测的帮助。此外，从表2中可看出，候选句抽取仍存在较大提升空间。结合错误分析，下一步计划挖掘阅读理解中更深层次的线索（如句间指代关联），提升候选句抽取效果，进一步提高答案预测的准确率。

参考文献

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, September. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Mor Geva and Jonathan Berant. 2018. Learning to search in long documents using document structure. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 161–176, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Yanchao Hao, Yuanzhe Zhang, Liu Kang, Shizhu He, and Jun Zhao. 2017. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Daniel Z. Korman, Eric Mack, Jacob Jett, and Allen H. Renear. 2018. Defining textual entailment. *Journal of the Association for Information Science Technology*.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations.

Tsung Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. 2017. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*.

Denis Lukovnikov, Asja Fischer, Jens Lehmann, and Sren Auer. 2017. Neural network-based question answering over knowledge graphs on word and character level. In *International World Wide Web Conference 2017*.

Sewon Min, Victor Zhong, Richard Socher, and Caiming Xiong. 2018. Efficient and robust question answering from minimal context over documents.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California, June. Association for Computational Linguistics.

Qiu Ran, Peng Li, Weiwei Hu, and Jie Zhou. 2019. Option comparison network for multiple-choice reading comprehension.

Stephen E. Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations Trends® in Information Retrieval*, 3(4):333–389.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2016. Conceptnet 5.5: An open multilingual graph of general knowledge.

Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. 2008. Learning to rank answers on large online QA collections. In *Proceedings of ACL-08: HLT*, pages 719–727, Columbus, Ohio, June. Association for Computational Linguistics.

Min Tang, Jiaran Cai, and Hankz Zhuo. 2019. Multi-matching network for multiple choice reading comprehension. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:7088–7095, 07.

Harsh Trivedi, Heeyoung Kwon, Tushar Khot, Ashish Sabharwal, and Niranjan Balasubramanian. 2019. Repurposing entailment for multi-hop question answering tasks.

Shuohang Wang, Mo Yu, Jing Jiang, and Shiyu Chang. 2018. A co-matching model for multi-choice reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 746–751, Melbourne, Australia, July. Association for Computational Linguistics.

Hai Wang, Dian Yu, Kai Sun, Jianshu Chen, and Dan Roth. 2019. Evidence sentence extraction for machine reading comprehension. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana, June. Association for Computational Linguistics.

Vikas Yadav, Steven Bethard, and Mihai Surdeanu. 2019. Quick and (not so) dirty: Unsupervised selection of justification sentences for multi-hop question answering. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

Shuailiang Zhang, Hai Zhao, Yuwei Wu, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. 2019. Dual co-matching network for multi-choice reading comprehension.

基于层次化语义框架的知识库属性映射方法

李豫
华中师范大学/ 计算机学院
liyu@mails.ccnu.edu.cn

周光有
华中师范大学/ 计算机学院
gyzhou@mail.ccnu.edu.cn

摘要

面向知识库的自动问答是自然语言处理的一项重要任务，它旨在对用户提出的自然语言形式问题给出精炼、准确的回复。目前由于缺少数据集、特征不一致等因素，导致难以使用通用的数据和方法实现领域知识库问答。因此，本文将“问题意图”视作不同领域问答可能存在的共同特征，将“问题”与三元组知识库中“关系谓词”的映射过程作为问答核心工作。为了考虑多种层次的语义避免重要信息的损失，本文分别将“基于门控卷积的深层语义”和“基于交互注意力机制的浅层语义”两个方面通过门控感知机制相融合。我们在NLPCC-ICCPOL 2016 KBQA数据集上的实验表明，本文提出的方法与现有的基于CDSSM和BDSSM相比，效能有明显的提升。此外，本文通过构造天文常识知识库，将问题与关系谓词映射模型移植到特定领域，结合Bi-LSTM-CRF模型构建了天文常识自动问答系统。

关键词： 知识库 ； 属性映射 ； 深层语义

Property Mapping in Knowledge Base Under the Hierarchical Semantic Framework

Yu Li
School of Computer /
Central China Normal University
liyu@mails.ccnu.edu.cn

Guangyou Zhou
School of Computer /
Central China Normal University
gyzhou@mail.ccnu.edu.cn

Abstract

KBQA is an important task in natural language processing. It aims to give refined and accurate responses to natural language questions raised by users. At present, due to the lack of data sets and inconsistent features, it is difficult to use common data and methods to implement domain-specific. Therefore, this paper focuses on the core work of KBQA by identifying the intent of users' questions and attempting to map the questions and the predicates in the knowledge base. To avoid the loss of the important information, we combine the "deep semantics based gate convolution" and "shallow semantics based on interactive attention mechanism" into a unified framework using the gated perception mechanism. We conduct experiments on NLPCC-ICCPOL 2016 KBQA dataset. The results show that our proposed method significantly outperforms the existing CDSSM and BDSSM. Besides, we also construct a commonsense knowledge base under the domain of astronomy. Furthermore, we build a commonsense automatic question answering system by applying the proposed model and Bi-LSTM-CRF into the astronomy domain.

Keywords: Knowledge base , Property mapping , Deep semantics

1 引言

问答任务（Question Answer，QA）是人工智能的核心研究之一。与传统搜索引擎相比，自动问答的便捷性和高效性增强了用户信息获取的体验，也使更多的学者开始对问答系统进行深入的研究。大规模知识库的迅速发展为实现自动问答目标提供了丰富有效的资源支撑，这使得面向知识库的自动问答（Knowledge Base Question Answer，KBQA）在工业界和学术界均受到了广泛的关注。知识库问答的目的就是根据用户提出的自然语言问题找到知识库中与之相关的知识，最后返回一个简洁、准确的答案。KBQA任务的核心工作是建立起问题到知识库的关系映射，而如何让机器理解自然语言问题与知识库三元组之间的语义等价关系是一个具有挑战性的难点。因此，本文探索的知识库属性映射方法可以作为KBQA系统中由问题关联到知识库的一种有效途径。

目前，随着如DBPedia (Auer et al., 2007)，Freebase (Bollacker et al., 2008)，Yago2 (Hoffart et al., 2011)，WikiData (Vrandečić and Krötzsch, 2014)等比较成熟的大型知识库相继涌现，自动问答的学术研究热度也在这些典型知识库基础上日益升温。知识库问答的实现有两大类主流方法，一种是语义解析（semantic parsingbased，SP-based），另一种是信息检索（information retrieve-based，IR-based） (Dong et al., 2015)。基于语义解析的方法是将问句分析成特有的表达形式或查询语句，如SPARQL、SQL语句等从知识库中搜索出答案；基于信息检索的方法是对候选答案通过特定方式进行排序得到最佳回答。早期针对小规模的知识库，多以语义解析方法为主，但这类方法往往会耗费大量精力去标注逻辑规则，也难以扩展到大规模知识库。Liang等 (2013)利用问答对语料，使用弱监督学习方法对问题进行过语义解析研究。Berant等 (2013)开发过一种语义解析器，可以训练无注释的逻辑形式，也可以扩展到大型知识库。Berant等 (2014)提出了一种基于释义的学习语义解析器的新方法以利用知识库中未涵盖的大量文本，但是其中一些工作仍依赖于手工标注和预定义规则，人工和时间成本较高。信息检索的方法则侧重于特征抽取以及对候选项的匹配和排序模型研究，其基本步骤是：“主题实体抽取”和“问题与关系谓词映射”。Yao等 (2014)使用句法分析技术，获得问句中的关键实体以及查询图。其他一些研究 (Zettlemoyer and Collins, 2012; Kwiatkowski et al., 2010)使用基于嵌入的模型来学习问题词和知识库构成的低维向量，并使用这些向量的总和来表示问题和候选答案，但是忽略了词序信息。Lai等 (2016)人使用主语谓语抽取算法，通过基于词向量的相似度结合分词技术实现属性映射，并利用人工定义的模板和规则取得了很好的效果。Wang等 (2016)使用分类器判断三元组中谓词与问题的映射。Yang等 (2016)使用了基于短语-实体字典的主题短语检测模型来检测问题与主题短语，之后使用排序模型对候选者进行排名。周博通等 (2018)在知识库问答属性映射问题上采用双向LSTM结合两种不同的注意力机制计算问谓相关度，在问题与谓词的映射测试上取得了91.77%的准确率。Xie等人 (2016)将CDSSM（Convolutional Deep Structured Semantic Models）与BDSSM（Bi-LSTM Deep Structured Semantic Models）相结合并利用余弦相似度计算问题与知识库关系谓词的匹配分数，但是其中采用的余弦距离是一个无参的匹配公式，并且仅使用深层神经网络可能会丢失一些重要的浅层词向量语义信息。赵小虎等 (zhao et al., 2020)通过将问题和知识库中三元组整体进行语义和字符的多特征匹配，并使用有参的全连接层计算相似分数，但是尚未考虑到浅层词向量的直接影响。

综合以上工作来看，问题与知识库的属性映射在KBQA任务中十分重要，同时也存在进一步改进的空间。本文着重关注问题与知识库谓词之间的映射方法，从表征和匹配两个角度改进前人所提到的CDSSM“问谓属性映射模型” (Xie et al., 2016)。在表征层中，首先针对问题的表述通过增设卷积门来过滤问题中与谓词无关的词级噪声，再使用两种共享的语义获取模型得到待匹配项的深层语义与浅层语义，最后利用门控机制平衡两种不同层次的语义得到层次化待匹配向量。在匹配层中，本文获取问题与关系谓词之间的多种联系，再由多层感知机融合，经池化操作获取最终的语义匹配得分。本文在NLPCC-ICCPOL 2016发布的中文问答数据集上进行属性映射实验，实验结果表明了该方法的有效性。另外，由于问题与谓词的映射是一种较为通用的问题意图识别过程，例如时间、地点、概念、因果、人物等通用询问意图在其他领域问答中也多有涉及，因此适合迁移到其他领域的问答。依照这种思路，本文构建了中文天文常识知识库，将天文命名实体识别作为基础任务，将面向知识库的“问谓属性映射”作为重点研究内容，构建了天文常识自动问答系统。综上，本文的贡献如下：

（1）针对问题表达，通过增设卷积门，适当过滤问题中与谓词无关的词级噪声。

（2）采用交互注意力机制获取浅层词向量全局语义。通过门控感知机制在表征层面有效地融合了层次化语义信息，既考虑了深层语义又防止浅层语义信息的丢失。

（3）最后，本文通过构建天文常识知识库以及将上述“问谓属性映射”方法迁移到特定领域知识问答中，与Bi-LSTM-CRF模型相结合构建天文常识自动问答系统。

2 面向知识库的属性映射

2.1 数据来源

由于中文知识库和相关问答语料较为欠缺，所以在中文知识库问答方面一直鲜有研究。在2016年和2017年NLPCC-ICCPOL发布了中文知识库以及问答对语料后，许多学者都开始围绕此项语料数据展开研究工作。为了扩展问答意图的范围，本文也选用了NLPCC-ICCPOL 2016年评测中公开的基于知识库的问答数据。这类数据来源于百科信息栏三元组，将其运用到领域知识库的问答中会具有较为全面的覆盖度。数据集的原始数据格式是：<问题，三元组，答案>三元组，例如：“机械设计基础这本书的作者是谁？”；“机械设计基础|||作者|||杨可桢，程光蕴，李仲生”；“杨可桢，程光蕴，李仲生”。在实际深度学习之前，需要针对所要用到的方法对原数据集进行修正和加工。

2.2 数据处理

数据重新处理的过程中，保留问题以及对应三元组中的关系谓词，如上例中的问题和谓词“作者”，对数据集中的全体谓词构造谓词词典，在词典中随机抽取9个谓词负例与正确谓词合并作为谓词候选集。随机初始化标签顺序，生成对应的谓词标签，其中正确谓词对应的标签为1，错误谓词对应的标签是0。在数据本身的标准性上，由于数据存在大量谓词中间空格现象，如“作者”这一谓词的原数据格式为“作 者”，需要去除空格保持数据一致性。另外，在数据预处理的过程中，本文严格控制谓词候选集中不存在重复项，从而防止训练和测试产生误差。同时，通过人工核查尽可能避免候选谓词中存在与正确谓词是同义词的情况，如“俗名”和“俗称”，“位置”和“地域所属”等，以免在准确率上出现偏差。为了探究主题实体在不同方法中产生的影响，我们将数据集分为掩盖主实体和不掩盖主实体两类。未掩盖主实体的数据集中，问句不做任何处理。在掩盖主实体的数据集中，首先根据每个问句对应的知识库三元组找到具体的主题实体，再将这些实体在问句中用特殊符号掩盖。例如问句“波色-爱因斯坦凝聚态有哪些比喻？”，经过掩盖后为：“E有哪些比喻？”；问句“大熊座47c多长时间一个周期？”掩盖后为：“E多长时间一个周期？”。转换完毕的数据集新格式为：<问题，谓词候选集，标签>。

2.3 属性映射框架

首先介绍本文提出的问题与关系谓词的属性映射网络架构GHSMM（Gate Hierarchical Semantic Match Model），结构图如图1所示。

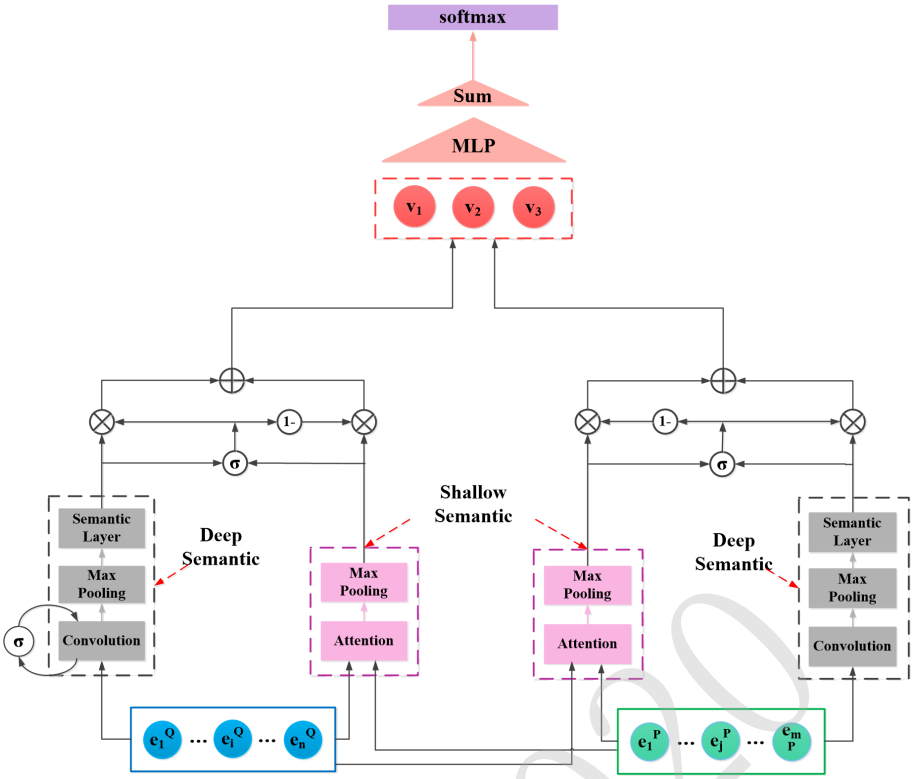


图 1: 融合门控感知的层次化语义匹配框架GHSMM

问句与候选谓词集合在初始状态下均以Word2vec词向量矩阵表示。在使用模型时，首先将问题与关系谓词的向量矩阵同时输送至“融合卷积门的深层语义模块”与“基于交互注意力机制的浅层语义模块”中得到两种层次语义向量，之后由“基于门控感知的层次化语义融合模块”将深层和浅层语义有效融合，经过“匹配层”和“决策层”得到候选谓词集与问题的匹配概率分布。

相较于传统的CDSSM模型，我们在获取深层语义的同时还考虑了上下文的浅层语义，同时，我们也改进了匹配层中简单的余弦夹角计算，利用多层感知机以及池化操作对问题和谓词的交互信息进行打分，在匹配效果上得到了一定提升。

2.3.1 融合卷积门的深度语义模型

问题语句较长且存在大量与谓词无关的噪声将会影响匹配，因此本文在CDSSM基础上增设卷积门，对问句进行门控过滤。

卷积神经网络（Convolutional Neural Network，简称CNN）可以有效地提取出矩阵的局部特征并在此之上进行全局的预测。给定句子序列的词向量$E = \{e_1, e_2, \cdots, e_n\}$，其中表示第$i$个词的词向量表示。通过设置卷积核的大小，使用卷积操作矩阵w对向量矩阵E进行卷积操作，得到结果$c = \{c_1, c_2, \cdots, c_{|E|-w+1}\}$。其中$\otimes$代表卷积运算，$f_x$为非线性激活函数，$b_c$是偏置项。

$$c_i = f_x(w \otimes E_{(i-w+1):i} + b_c) \tag{1}$$

为了进一步过滤问句中与谓词无关的词级噪声，本文采用卷积门控制问句的卷积输出，而对谓词则不使用门控机制。谓词的卷积结果c_p经过Relu激活函数直接输出。问句的卷积结果c_q由问句向量E_q通过两个卷积网络得到，其中一个是原始的CNN，另一个在sgmoid函数激活下生成门控向量。

$$c_p = \text{ReLu}(w \otimes E_p + b_1) \tag{2}$$

$$c_q = \text{ReLu}(w \otimes E_q + b_1) \odot sigmoid(v \otimes E_q + b_2) \tag{3}$$

我们采用最大池化操作提取特征图中的重要信息，将通过不同大小卷积核得到的卷积输出分别进行池化和拼接，最后再通过一个全连接层与tanh非线性函数将问句或谓词投影到一定维度大小的语义空间。

$$h_d = \tanh([\max(c^{(1)}); \max(c^{(2)}); ...; \max(c^{(win)})] \cdot W_d + b_d) \tag{4}$$

2.3.2 基于交互注意力的浅层语义表示

浅层语义可以反映全局的文本信息，我们使用交互注意力机制为词向量添加注意力信息，如图2所示。经过交互后，问句的每一个序列都对应一个待匹配谓词的全局语义向量；同样，谓词的每一个序列也对应着一个问句的全局语义向量。经过最大池化操作获取浅层语义信息。

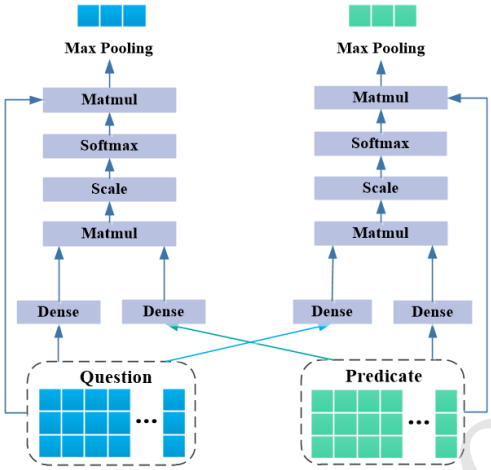


图 2: 基于交互注意力机制的浅层语义表示

注意力机制 (Vaswani et al., 2017)目前已广泛应用于自然语言处理领域，用来增大重要信息的权重系数，使模型关注重要的部分。公式(1)是注意力机制的计算过程。Q，K，V分别是输入向量与三个权重矩阵W^Q,W^K,W^V相乘的结果，各自代表了查询（query）、键（key）、真实值（value），其中Q与K的输出维度相同。Q与K的交叉乘积除以d_k的平方根是为了防止内积过大而影响梯度，经过softmax函数归一化后得到注意力权重。最后将V在Q的每一个位置上的向量进行一次加权求和，得到$Attention(K, Q, V)$，表达了对各个词的注意力程度。

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{5}$$

在使用注意力机制时，问句的目标语句是待匹配的谓词，谓词的目标语句是待匹配的问句，value值均为句子本身的表示（即Word2vec词向量）。E_q 与E_p分别表示问句q与谓词p的Word2vec词向量矩阵，最后得到问句和谓词的交互注意力语义向量S_q、S_p。

对于问句q，注意力运算如下：

$$S_q = Attention(E_p \cdot W^Q, E_q \cdot W^K, E_q) \tag{6}$$

对于谓词p，注意力运算如下：

$$S_p = Attention(E_q \cdot W^Q, E_p \cdot W^K, E_p) \tag{7}$$

为了获取全局语义信息，我们采用最大池化的操作对上述得到的注意力信息进行池化过滤，得到添加了注意力机制的词向量全局语义信息h_s。

$$h_s = \max(S) \tag{8}$$

2.3.3 基于门控感知的层次化语义融合

交互注意力机制可以融合问句与谓词之间的长期依赖信息，卷积神经网络可以较好地提取词与词之间的短距离特征，我们增设门控感知机制，将深层语义特征h_d与浅层词级特征h_s进行融合。

g表示它们生成的门控向量。⊙表示逐点乘积操作。

$$g = sigmoid(h_d \cdot W_g^{(1)} + h_s \cdot W_g^{(2)} + b_g) \tag{9}$$

门控机制将层次化语义信息进行平衡，其中g中每一个元素代表将多少浅层语义特征替换为深层语义特征，从而得到最终的层次化语义向量y。

$$y = g \odot h_d + (1 - g) \odot h_s \tag{10}$$

2.3.4 交互匹配层

CDSSM中采用了余弦距离计算问句与谓词之间的相似性，但余弦计算是一种简单无参数匹配方法。为了更充分地度量问谓相似程度，我们一方面将问谓间的多种交互特征进行融合，另一方面通过多层感知机获取问题与谓词之间多个方面上的相似得分，之后采用加和池化得到整体相似度。

这里我们选用三种方式对问题与谓词进行语义交互。v_1代表全局语义向量之和，v_2表示向量间的绝对差，v_3是向量之间的逐个点积运算。

$$v_1 = y_q + y_p \tag{11}$$

$$v_2 = \mid y_q - y_p \mid \tag{12}$$

$$v_3 = y_q \odot y_p \tag{13}$$

将v_1，v_2，v_3三者拼接形成一个扁平的向量，并输入到一个两层的全连接层中，将输出结果投射到m维作为评价相似度的m个方面，即$s = \{s_1, s_2, \cdots, s_m\}$，$w_s$、$w_h$是多层感知机中学习的权重，$b_s$、$b_h$是学习的偏置项。

$$s = w_s \cdot \tanh(w_h \cdot [v_1; v_2; v_3] + b_h) + b_s \tag{14}$$

在进行计算匹配得分前，需要将上一步得到的结果进行池化操作，这里我们选择加和池化计算匹配分数，并用sigmoid激活函数把匹配值压缩到0到1之间。将问题q与谓词p_i（其中$i = 1, 2, 3 \cdots k$，k为谓词集合的元素数目）的语义向量按照本文提出的方式进行一对一匹配，每一个问句对应得到k个语义相关性$SMS(p_i|q)$。

$$SMS(p_i \mid q) = sigmoid(\sum_{i=1}^{m} s_i) \tag{15}$$

2.3.5 决策层

将语义匹配得分送入softmax分类器中来预测最终的正确匹配项，并计算添加了正则项后的交叉熵目标函数。

$p(p_i|q)$是问题q与第i个谓词p_i相匹配的概率。其中，P是问题q的一组候选谓词，包括几个否定谓词样本和一个肯定谓词样本。p'代表候选谓词集P中的任意谓词元素。

$$p(p_i \mid q) = \frac{\exp(SMS(p_i \mid q))}{\sum_{p' \in P} \exp(SMS(p' \mid q))} \tag{16}$$

训练语义模型以最大化肯定谓词的可能性为训练目标，L是目标损失函数。其中，q_r代表R个问题中的第r个问题，p^+代表正确的谓词，$p(p^+|q_r)$是第r个问题中给定正谓词的条件概率，λ是L_2正则化参数，θ是模型的参数。最后使用优化算法对目标函数进行优化，此过程采用误差反向传播的方式更新各层权重和偏置值。

$$L(\theta) = -\log \prod_{r}^{R} p(p^+ \mid q_r) + \lambda \parallel \theta \parallel^2 \tag{17}$$

3 实验

3.1 实验设置与评测指标

实验采用NLPCC-ICCPOL公开数据集，其中训练集包含14609个问句，测试集包含9870个问句。本文从训练集中选取3000句作为开发集，剩下的11609个问句作为实际的训练集。实验环境：本实验的环境为tensorflow框架，编程语言为Python3.5，重要超参数设置情况：选取300维的Word2Vec词向量模型，batch size大小为50，学习率为0.005，卷积核窗口大小为1，2，3，卷积核数目为100，卷积步幅为1。为防止梯度爆炸使用了梯度裁剪。实验选择Momentum优化器。评测指标采用通用的准确率。

3.2 属性映射实验

	模型	测试集Acc(%) 无主实体	有主实体
对比模型	BiLSTM_AC1 (周博通等人, 2018)	86.74	-
	BiLSTM_AC12 (周博通等人, 2018)	87.64	-
	BiLSTM_AC12_Overlap (周博通等人, 2018)	91.77	-
	BDSSM	91.81	91.42
	CDSSM	92.15	91.49
本文模型	Attention+MLP（浅层语义匹配）	93.49	93.57
	CNN+MLP（无卷积门深层语义匹配）	93.68	94.01
	GCNN+MLP（融合卷积门深层语义匹配）	93.78	94.07
	HSMM（未引入平衡门的层次化语义匹配）	93.51	94.22
	GHSMM (综合模型：融合门控机制的层次化语义匹配)	**93.99**	**94.68**

表 1: 各种不同方法的属性映射实验结果

本文选取前人提出的神经网络与简单文本特征的结合模型、BDSSM模型以及CDSSM模型作为对比模型，进行属性映射实验。通过表1可以看出，本文的浅层语义匹配方法和深层语义匹配方法与对比模型相比效果均有提高，而将浅层语义和深层语义融合后得到GHSMM综合模型，比单独使用这两种语义时取得了更优的效果，证明了本文层次化语义匹配方法的有效性。

另外，实验显示，保留问句中的主题实体提高了所有本文改进模型的准确率，但没有对CDSSM、BDSSM起到提升作用。说明加入主实体更有利于本文模型的匹配。

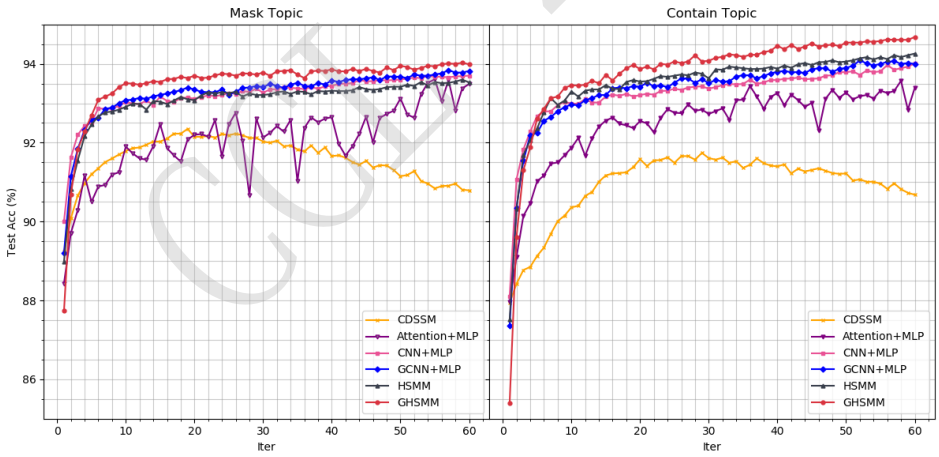


图 3: 本文各项改进方法在不同迭代次数下的效能比较

为了更好地说明本文在原CDSSM模型基础上所做的各项改进模块的有效性，我们对模型做了消融实验，即比较了不同迭代次数下各项改进模块的准确率。图3分别展示了基线模型（CDSSM）、基于交互注意力机制的浅层语义匹配（Attention+MLP）、无卷积门的深度语义匹配（CNN+MLP）、有卷积门的深度语义匹配（GCNN+MLP）、通过简单加和实现的层次化语义匹配（HSMM）以及组合起来的融合语义平衡门的层次化语义匹配（GHSMM）在一次训练中各迭代轮数下的性能。实验分别在包含主题实体的数据集（图3右部分）与掩盖主题实体的数据集（图3左部分）上进行。从图中可以看出，在实体被掩盖时有卷积门和无卷积门两者在每次迭代中的实际效能差距很小，说明此时卷积门对问句的过滤作用还比较微弱，而在保留主题实体时有卷积门的过滤效果相对更明显一些；单纯使用注意力机制得到的浅层语义匹配在

整体训练效果上具有较大的起伏，总体上弱于其他改进模型；HSMM将深层语义与浅层语义通过简单加和的方式相结合，效果要弱于GHSMM，这进一步证明了加入语义平衡门能使两种语义自适应地达到更好的结合效果。整体上看，层次化语义模型训练趋势平稳，且在两组实验数据集中呈现的总体效果为最优，这表明本文将浅层语义与深层语义成功地融合在一起，减少了语义信息的损失。

3.3 天文问答系统示范性应用

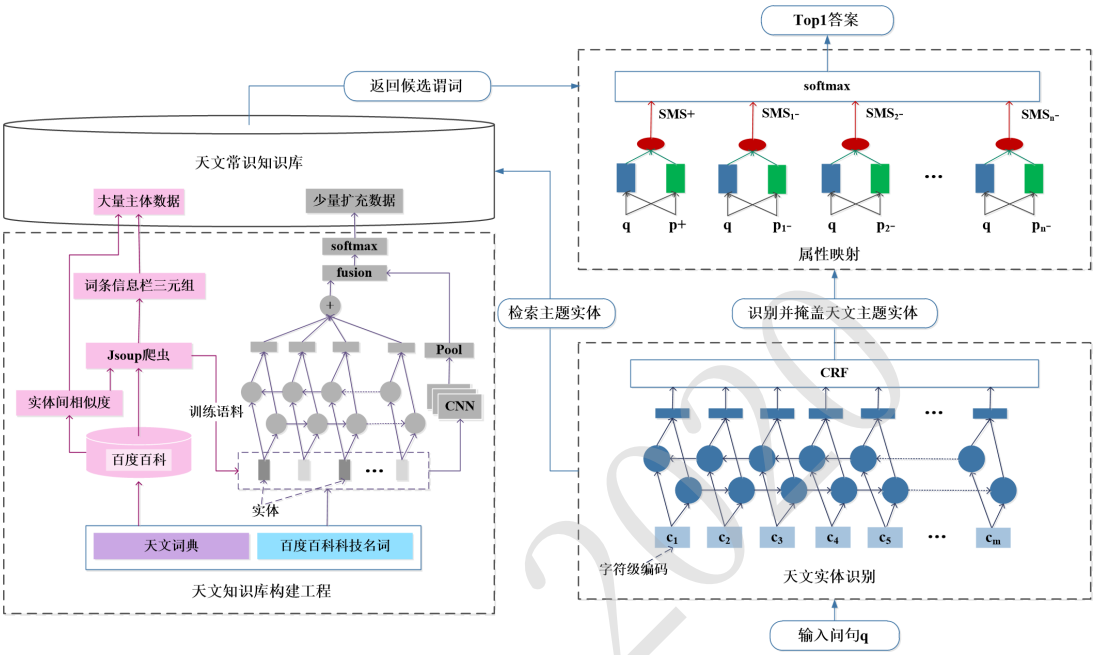


图 4: 天文问答系统构建

本文将提出的基于层次化语义框架的知识库映射方法应用到实际的领域知识库问答中，实现了天文问答系统的示范性应用。图4是整个天文问答系统的构建，包含“天文知识库构建”和“问答实现”两个大模块。

3.3.1 天文常识知识库的构建

目前，已经开放的天文知识库和相关成型的语料十分罕见，而百科知识语料属于互联网上开放的知识文本数据，具有规模庞大、持续更新扩展的特点，如著名的DBPedia知识库就是从维基百科中抽取构建的。因此，本文选择从百度百科网络资源中获取天文知识语料并进一步展开结构化信息抽取工作。百度百科具有较强的专业性和全面性，而且在词条页面中附带结构化的数据信息，这使得后期的信息抽取工作变得方便有效。本文首先下载了天文词典以及爬取百度百科“科技名词栏”中的天文术语，构建出了一份天文实体集，该实体集一共包含24376个天文学名词。本文依照实体集爬取了对应的百度百科信息框，将这些信息以三元组的格式存储，例如：{“木卫四”，“外文名”，“Dione”}。对于某些特定关系的尾实体（如“类别”、“别称”、“成分”等）要进行字符串的切割操作，将含多个并列成分的句子分开，例如：将{“日冕”，“结构”，“内冕、中冕和外冕3层”}这一个三元组拆分成：{“日冕”，“结构”，“内冕”}、{“日冕”，“结构”，“中冕”}、{“日冕”，“结构”，“外冕”}三个三元组。由于基于上述爬虫方法获取到的大多数是“实体-属性-属性值”类别的三元组，缺少“实体-关系-实体”这类三元组，因此为了进一步扩展知识库，本文事先回召了已有三元组中特定关系的关系句子，并作为语料通过基于注意力机制的Bi-LSTM+CNN网络训练关系抽取模型，再使用基于链入词条的TF-IDF的相似度计算获取关联度高的实体对，选择包含这些实体对的关系句子送入模型进行关系标签预测，经过人工评估后作为少量扩充数据入库。最终构建的知识库包含53975个三元组，以及10258个关系度较高的实体对和它们的关系程度数值。

3.3.2 天文问答的实现

上述方法构建的天文常识知识库与NLPCC-ICCPOL发布的知识库存在类似的问题，例如许多三元组宾语是以字符串形式表示而非知识库中的实体，因此难以形成像知识图谱这种网络拓扑结构；其次，对于同一个主实体可能存在多个几乎同义的关系谓词。很多在Freebase等外文知识库的研究并不适合直接应用在此类中文的数据集中，也无法处理需要多个三元组进行回答的复杂问题，而比较适合用来用来回答单实体单关系类型的问题。综合这些因素，本文将天文问答分为Bi-LSTM-CRF天文命名实体识别步骤与GHSMM属性映射步骤。考虑到大多数天文主实体较为生僻，因此使用主实体掩盖方法，使问句更接近自然语法结构。首先将问句经过实体识别层找出问句中对应的天文主实体，再将实体替换为特殊符号与检索出的相关候选谓词一并输送到属性映射层进行匹配。由于难以获取符合条件的数据集和标签，我们训练Bi-LSTM-CRF模型的语料是由爬取的天文百度百科文本经过分句、分字以及采用字符串匹配算法对照天文实体集添加字标签获取的，标签采用的序列格式为BIOES。在属性映射层中，本文模型可以处理不同数目的关系谓词，系统将匹配概率最高的谓词作为该问题的核心意图，联合主题实体和关系谓词，返回对应的尾实体作为最终答案。

3.4 样例分析

为了显示证明模型理解问题语义与选取关系谓词的有效性，本文抽取了一个天文问句例子，从已构建的天文知识库中检索相关实体和全体关系谓词，经过上述问答系统的识别后展示候选谓词分数经过归一化后得到的概率分布。

贝利珠名字的由来是什么?					
	外文名	命名原因	出现时间	人物	学科
CDSSM	**0.25513**	0.25404	0.16021	0.18954	0.14108
GHSMM	0.19977	**0.20090**	0.19977	0.19979	0.19977

表 2: 样例分析

如表2所示，该样例包含一个有关主题实体“贝利珠”的问题，并且询问的意图与“名字”有关。针对“贝利珠”主题实体，从知识库中检索出来的候选谓词有表中所示的5个。输入问句“贝利珠名字的由来是什么？”，在CDSSM中，带有“名字”意义的两个关系谓词“外文名”与“命名原因”在谓词候选集中的匹配概率都很高，模型最终错误地选择了与“名字”语义接近的“外文名”。同时，我们发现CDSSM中其他候选谓词也随着本身语义的相关度差异显示不同的概率，例如“人物”这种类型的词从某种程度上也经常和“名字”同时出现，语义相关度较高，故概率值也偏高一些。

而经过本文提出的GHSMM处理后，正确谓词匹配概率相对较高，而其他错误候选项的概率值接近一样，降低了其他词语在语义远近上的影响，例如在上述问句中与名字有关的“外文名”与不相关谓词“出现时间”、“学科”等概率值几乎相同。

上述分析再次表明本文提出的GHSMM比CDSSM具有更好的性能，在一定程度上避免了词语之间语义相近导致的错误匹配情况。

4 结语

基于知识库的自动问答大多数做法是通过主题实体识别和问句与关系谓语的属性映射，由于领域知识库缺少问答数据集，本文以NLPCC-ICCPOL数据进行问谓属性映射训练并迁移到天文领域知识库中实现自动问答。在属性映射研究过程中，本文将前人的CDSSM模型进行了改进，在特征抽取步骤中利用门控感知机制融合层次化语义信息，使最后的向量表示同时具备浅层的和深层的语义；在匹配层中我们将CDSSM中原有的无参余弦匹配改进为融合多种交互信息的多层感知机，通过池化等操作得到最终语义匹配分数，实验证明最终改进的属性映射方法比CDSSM与BDSSM等模型的匹配效果有明显提高。但是本文所实现的领域知识库自动问答系统仍然存在许多限制和不足，在下一步工作中，笔者希望将模型进一步改进，并关注问句中主题实体的提取环节，以实现更高性能的特定领域自动问答应用。

致谢

本文的工作作为毕业论文的一部分，受到国家自然科学基金（No. 61972173）支持。感谢匿名评审专家对我们工作提出的建设性修改意见。

参考文献

Auer S, Bizer C, Kobilarov G, et al. *Dbpedia: A nucleus for a web of open data*[M]. The semantic web. Springer, Berlin, Heidelberg, 2007: 722 735.

Bollacker K, Evans C, Paritosh P, et al. *Freebase: a collaboratively created graph database for structuring human knowledge*[C]. Proceedings of the 2008 ACM SIGMOD international conference on Management of data. ACM, 2008: 1247 1250.

Berant J, Chou A, Frostig R, et al. *Semantic parsing on freebase from question answer pairs*[C]. Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. 2013: 1533-1544.

Berant J, Liang P. *Semantic parsing via paraphrasing*[C]. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2014, 1: 1415 1425.

Dong L, Wei F, Zhou M, et al. *Question answering over freebase with multi-column convolutional neural networks.* // The Association for Computational Linguistics. Beijing, 2015: 260–269.

Hoffart J, Suchanek F M, Berberich K , et al. *YAGO2: exploring and querying world knowledge in time, space, context, and many languages*[C]. Proceedings of the 20th international conference companion on World wide web. ACM, 2011: 229 232.

Kwiatkowski T, Zettlemoyer L, Gold water S, et al. *Inducing probabilistic CCG grammars from logical form with higher order unification*[C]. Proceedings of the 2010 conference on empirical methods in natural language processing. Association for Computational Linguistics, 2010: 1223 1233.

Lian g P, Jordan M I, Klein D. *Learning dependency based compositional semantics*[J]. Computational Linguistics, 2013, 39(2): 389 446.

Lai Y, Lin Y, Chen J, et al. *Open domain question answering system based on knowledge base*[M]. Natural Language Understanding and Intelligent Applications. Springer, Cham, 2016: 722 733.

Vrandecic D, Krotzsch M. Wikidata. *a free collaborative knowledgebase*[J]. Communications of the ACM, 2014, 57(10): 78 85.

Vaswani A, Shazeer N, Parmar N, et al. *Attention is all you need*[C]. NIPS 2017: Advances in Neural Information Processing Systems, 2017: 5998 6008.

Wang L, Zhang Y, Liu T. *A deep learning approach for question answering over knowledge base*[M]. Natural Language Understanding and Intelligent Applications. Springer, Cham, 2016: 8 85 892.

Xie Z, Zeng Z, Zhou G, et al. *Topic enhanced deep structured semantic models for knowledge base question answering*[J]. SCIENCE CHINA: Information Sciences,2017,60(11):28-42.

Yao X, Van Durme B. *Information extraction over structured data: Question answering with freebase*[C]. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2014, 1: 956 966.

Zettlemoyer L S, Collins M. *Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars*[J]. arXiv preprint arXiv:1207.1420, 2012.

Yang F, Gan L, Li A, et al. *Combining deep learning with information retrieval for question answering*[M]. Natural Language Understanding and Intelligent Applications. Springer, Cham, 2016: 917 925.

周博通，孙承杰，林磊等. 基于*LSTM* 的大规模知识库自动问答[J]. 北京大学学报（自然科学版）, 2018, 54(2): 286 292.

赵小虎，赵成龙. 基于多特征语义匹配的知识库问答系统[J/OL]. 计算机应用:1-6[2020-06-09].http://kns.cnki.net/kcms/detail/51.1307.TP.20200519.1403.004.html.

面向垂直领域的阅读理解数据增强方法

吕政伟，杨雷，石智中，梁霄，雷涛，刘多星
汽车之家/ 中国，北京
{lvzhengwei, yanglei, shizhizhong,
liangxiao12030, leitao, liuduoxing}@autohome.com.cn

摘要

阅读理解问答系统是利用语义理解等自然语言处理技术，根据输入问题，对非结构化文档数据进行分析，生成一个答案，具有很高的研究和应用价值。在垂直领域应用过程中，阅读理解问答数据标注成本高且用户问题表达复杂多样，使得阅读理解问答系统准确率低、鲁棒性差。针对这一问题，本文提出一种面向垂直领域的阅读理解问答数据的增强方法，该方法基于真实用户问题，构造阅读理解训练数据，一方面降低标注成本，另一方面增加训练数据多样性，提升模型的准确率和鲁棒性。本文用汽车领域数据对该方法进行实验验证，其结果表明该方法对垂直领域阅读理解模型的准确率和鲁棒性均能有效提升。

关键词： 阅读理解 ； 数据增强 ； 问答系统

Method for reading comprehension data enhancement in vertical field

Zhengwei Lv,Lei Yang,Zhizhong Shi,Xiao Liang,Tao Lei,Duoxing Liu
Autohome Inc. / Beijing,China
{lvzhengwei,yanglei,shizhizhong,
liangxiao12030,leitao,liuduoxing}@autohome.com.cn

Abstract

Reading comprehension question answering system uses natural language processing technologies such as semantic understanding to analyze unstructured documents and generate answers, which has important theory value and vast application prospect. However, the costs of obtaining training samples for reading comprehension model are expensive and the user questions are complex and diverse in the vertical field, which leads to the poor accuracy and robustness. In response to the problem, this paper proposes a data enhancement method for reading comprehension question answering in the vertical field, which constructs training samples based on real user questions. So that it can reduce the cost of annotation and increase the diversity of training data. The experiments are carried out with the data in the automobile field and the results show that the method can effectively improve the accuracy and robustness of reading comprehension model in the vertical field.

Keywords: Reading comprehension , Data enhancement , Question answering system

1 引言

随着近几年智能问答的高速发展，阅读理解问答作为其重要发展方向之一，也逐渐成为了各领域的研究和应用热点。不同于传统问答系统中利用知识表示和检索方式获取答案(Qu et al., 2018; 安波et al., 2018)，基于阅读理解的问答利用模型直接对非结构化文档进行认知，从而获取给定问题的答案(Wang et al., 2017; Chen et al., 2017; Yu et al., 2018)。这种方式减少了知识的收集和表示过程，具有重要研究和应用价值。

阅读理解问答根据答案的产生方式，分为选择式、抽取式、生成式等类型，其中抽取式阅读理解根据问题从文档中抽取一个连续关片段作为答案，不用考虑答案的序列生成问题，答案获取方式直接，标注相对方便，难度适中，因此对抽取式阅读理解的研究相对较多。同时一系列大规模高质量评测数据集的发布，如SQUAD数据集(Rajpurkar et al., 2016; Rajpurkar et al., 2018)、DuReader数据集(He et al., 2017)、CMRC2018数据集(Cui et al., 2018)等，进一步促进了阅读理解问答的研究。但是这些数据集偏向于通用领域或百科知识，内容广而泛，针对垂直领域专业性的知识少，因此，面向垂直领域的抽取式阅读理解数据集标注和应用研究是十分必要。

在抽取式阅读理解数据集的标注过程中，标注人员提出的问题容易出现标注数据模式化，其表达方式单一、多样性不足，从而导致在应用中造成模型的准确性和鲁棒性较差。数据增强通常被用来解决这一问题，其原理是通过无监督、半监督或者有监督的方法构造新的训练样本，对原始的训练数据进行扩充，增加训练数据的量级和多样性，从而提升模型的准确性和鲁棒性。在机器阅读理解中，常见的数据增强方法有以下几种。

（1）远程监督方法，利用外部知识库自动对语料进行标注(白龙et al., 2019)，构造训练数据，增加数据的量级。然而，这种方法会引入很大的噪声，影响模型的语义理解，如图1，当问题的答案“日本”在一篇文档中多次重复出现，答案的标注位置不能很好的确定，将会影响模型整体的语义理解；

（2）问题生成方法，利用模型生成标注数据中问题的同义复述(Kim et al., 2019; Zhao et al., 2018)，实现增加问题表达的多样性。但是，目前序列生成技术相对不够成熟、且缺乏适当的评测指标，生成数据的质量难以控制，最终会造成构造数据的误差大，阅读理解模型效果差。

（3）完全生成方法，给定未标注文档，首先利用模型从文档中获取适合作为答案的片段，再根据文档内容和该片段生成相关问题，这种方法不需要已有阅读理解标注数据即可构造数据，能极大的提升构造数据的量级和覆盖范围。但是该方法引入的误差较大，除了问题生成环节的误差，在答案片段选取环节、问题和答案相关性等方面也会引入误差，形成误差的累积，最终影响构造数据的质量。

上述这些方法都是针对通用领域的研究，忽略了数据增强与实际应用数据的结合，造成构造数据与应用数据之间的语义偏差，影响模型应用效果。另外，在垂直领域中，领域术语多，问题更为专业，衍生出的表达方式更多样化，用远程监督或模型生成方式构造的数据，很难满足专业性和多样化性，容易造成模型应用中准确率低、鲁棒性差。

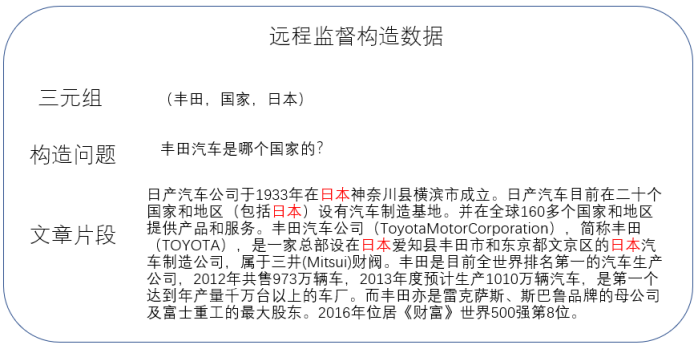


图 1. 远程监督方法构造训练数据

针对以上问题，本文提出了一种垂直领域中基于真实用户问题的数据增强方法，该方法也是对训练数据中的问题产生复述，以增加数据多样性，但不采用序列生成的方式，而是基于用户问题的表达形式进行构造，避免了序列生成模型的训练，增加数据的可控性，同时构造数据是基于真实数据产生的，增加了数据的一致性。该方法首先通过实体识别构建问题的语义原型库，并利用相似度计算获取当前问题的相似原型，然后对相似原型进行语义泛化，构造出包含真实语义结构的同义问句，增加问题的多样性，从而实现增加整个训练数据的量级和多样性。我们将本文提出的方法在真实汽车领域数据上进行实验，其结果表明该方法有效的提升了问答模型的准确率和鲁棒性。综上所述，本文的主要贡献包括：（1）提出了一种垂直领域中基于真实问题的数据增强方法，提升了模型的准确率和鲁棒性。（2）在汽车领域数据上对多个模型和数据增强方法进行了对比实验，实验结果证明了该方法的有效性。

2 相关工作

SQUAD等大规模评测数据的出现，引起学术界和工业界对抽取式阅读理解的深入研究，R-Net(Wang et al., 2017)、DrQA(Chen et al., 2017)、QANET(Yu et al., 2018)等一大批深度学习模型被相继提出。随着BERT(Devlin et al., 2018)、Roberta(Liu et al., 2019)、Albert(Lan et al., 2019)等预训练模型的提出，抽取式阅读理解取得了突破性进展，多种基于预训练模型的方法，在SQUAD2.0数据集上的评价指标超过了人类水平。本文根据各种深度学习模型和预训练模型，对垂直领域抽取式阅读理解的数据增强方法进行研究，以提升各种模型在垂直领域中的准确率和鲁棒性。在机器阅读理解任务中，常用的数据增强方法有远程监督方法、问题生成方法和完全生成方法三种。

远程监督的方法利用外部知识库自动对语料进行标注，构造训练数据，如Chen等(2017)利用QA问答对作为知识库，通过检索得到相关文档片段，构造训练数据。Zhang等(2018)通过知识三元组(E_1, R, E_2)，用实体E_1和关系R构造问题，实体E_2作为答案，用问题和答案检索无标注文档，构造训练数据，从而增加数据量级，提升模型性能。

问题生成的方法利用模型生成新的问题，构建训练数据，包括生成相关问题和生成同义复述问题两种。Zhu等(2019)通过生成相关但不可回答的问题，提升模型的语义理解能力，在SQUAD2.0数据集上取得1.9个F1点的提升。Gan等(2019)提出引导式的生成方法，利用seq2seq模型生成同义问题，增加问题的多样性，提升模型的准确性和鲁棒性。

完全生成的方法是给定文档，直接利用模型根据文档内容生成相关问题和答案，构造训练数据。如Subramanian等(2017)利用模型先从文档中提取关键短语，并以该短语为参考答案生成相关的问题，从而构造训练数据。Puri等(2020)先用BERT从文档中提取答案片段，再将答案和文档进行拼接，利用GPT2(Radford et al., 2019)模型生成相关问题，构造训练数据。

以上几种数据增强方法都是针对通用领域的研究，忽略了数据增强与实际应用数据的结合，会造成在垂直领域应用中构造数据与实际数据之间的语义偏差，从而影响模型应用效果。另外，远程监督的方法容易引入数据噪音，问题生成的方法其数据质量难以控制并且需要训练序列生成模型，同时垂直领域中数据专业性程度高，领域实体数量多，表达更多样，因此以上方法在垂直领域中不适用。借鉴其它自然语言处理任务中利用替换的方式进行数据增强的思想(Wei and Zou, 2019; Fadaee et al., 2017)，本文提出了一种垂直领域中基于真实用户问题的数据增强方法。该方法利用真实用户数据，对训练数据中的问题产生复述，以增加数据多样性，避免了序列模型的训练，增加数据的可控性，同时构造数据是基于真实数据产生的，增加了数据的一致性。最后在汽车领域数据集上，本文通过实验证明该方法对模型的准确率和鲁棒性均能有效提升。

3 数据增强方法

本文提出的数据增强方法是基于真实用户问题，该数据来源于问答系统的日志记录。该方法首先通过实体识别对用户问题进行处理，构建语义原型库；然后利用相似度计算方法，从原型库中获取当前问题的若干相似原型；最后对相似原型进行语义原型泛化，构造出包含真实用户问题语义结构的同义问题。

3.1 问题预处理

问题预处理，是将用户问题进行实体识别，从而获取问题语义原型的过程。将问句抽象为字符序列$Q = (c_1, c_2, c_3, \ldots, c_{(n-1)}, c_n)$，对序列Q进行实体识别，得到序列$Q^T = [c_1, \ldots E_1(c_i, c_{(i+1)}), \ldots E_2(c_{i+k}), \ldots c_n]$，其中$E_i$为识别出的实体，$c_i$为问句中的字符，$Q^T$称为问句的语义原型。将真实用户数据进行预处理，构建问句的语义原型库，从而可以获取大量的表达多样的语义原型数据来构造训练数据的同义问题。

问句预处理过程中的实体识别是指将文本中具有特定含义的文字片段作为一个整体识别出来。在通用领域，实体的类型主要有人名、地名、机构名称、专用名词等，在汽车垂直领域，实体的类型有车系、车型、品牌、车身参数、配置等等。

3.2 语义原型相似度计算

从原型库中找到与训练数据问题相似的问句原型，是构造同义问题的重要环节，相似语义原型的挑选既要考虑问句中已识别的实体序列的相关性，也要考虑字符序列的语义相关性。假设语义原型Q_1^T,Q_2^T，其相似度计算方法如式1：

$$P(Q_1^T, Q_2^T) = w_1 R_1 + w_2 R_2 + w_3 R_3 \tag{1}$$

其中w_1、w_2、w_3为权重参数；R_1为实体类型相关因子，代表两个原型实体类型的相关性；R_2为实体顺序相关因子，代表两个原型实体类型的先后顺序一致性；R_3为语义相关因子，代表两个原型的语义相关性。

R_1为实体类型相关因子，来衡量两个原型实体类型的相关性。定义：函数$E_set(Q^T)$为语义原型中实体类型的集合，$|E_set(Q^T)|$为实体类型个数，$|E_in| = |E_set(Q_1^T) \cap E_set(Q_2^T)|$为两个原型实体集合交集的实体个数。则$R_1$的计算方法如式2所示：

$$R_1 = 2(\frac{|E\_in|}{|E\_set(Q_1^T)|}\frac{|E\_in|}{|E\_set(Q_2^T)|})/(\frac{|E\_in|}{|E\_set(Q_1^T)|} + \frac{|E\_in|}{|E\_set(Q_2^T)|}) \tag{2}$$

R_2为实体顺序相关因子，顺序一致为1，否则为0。

R_3为两个原型的语义相似度值，本文语义相似度的计算采用SBERT(Reimers and Gurevych, 2019)模型，首先将原型中的实体词替换为实体名称，得到新的问题表示，利用孪生网络对问题中的字符进行向量化表示，通过计算向量的余弦值得到问题的相似度。网络的训练和推理如图2，训练阶段，问题1和问题2输入到BERT模型，经过平均池化，输出得到向量u和v。向量u、v及两个向量内部元素的差值$|u - v|$进行拼接，输入到Softmax分类器中进行训练。在推理阶段，直接计算u和v的余弦值，得到R_3值。

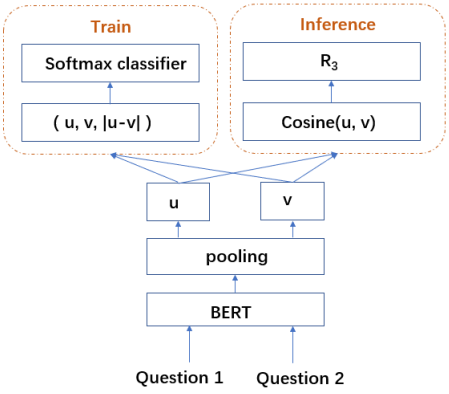


图 2. 语义相似度计算网络

3.3 语义原型泛化

语义原型泛化是对相似原型进行处理获取同义问题的过程，利用问题原型中的实体内容，替换相似原型对应的实体内容，改变了相似原型问句表达的内容主体，但是相似原型的语义结构保持不变，从而构造出主体内容一致，但表达形式多样的同义问题，从而能有效增强构造数据中问题的多样性表达。

语义原型的泛化过程如图3所示，通过对当前问题进行处理，从原型库选取与当前问题语义原型相似的若干原型，用当前原型的实体，替换相似原型中同类别实体，例如：用“宝马X3”替换“奥迪Q3”，用“价格”替换“钱”等，保留相似原型中的其它字符不变，从而得到当前问题的同义问题。

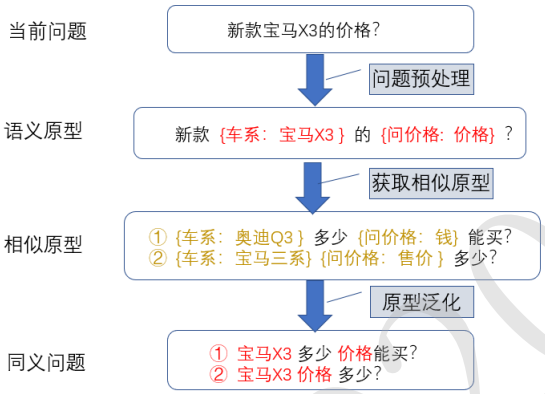


图 3. 基于语义原型的同义问题构造

4 实验

4.1 实验数据

本文将提出的数据增强方法在汽车领域数据集上进行验证，该数据集通过人工标注获取，对给定的每篇资讯文章提出3至5个相关问题并标出答案位置。该数据集共包含905篇汽车类资讯文章和2746个相关的问题，分为训练集和测试集两部分。同时为了验证鲁棒性，对测试集中的问题进行人工复述，每个问题生成若干个同义表达，产生鲁棒性测试集，共包含2312个同义表达的问题，数据样例见表1，具体细节见表2。

数据样例	文档	..... 今年11月昂希诺纯电动正式在国内上市，补贴后售价为**17.28-19.88万元**，共三款车型，今天我们测试的是它的顶配车型TOP悦享版。昂希诺纯电动的电池容量为64.2kWh，电芯为宁德时代NCM配比为523的方壳三元锂电芯，NEDC续航里程为500km。.....
	问题	北京现代昂希诺纯电动补贴后售价是多少?
	答案/索引	17.28-19.88万元/(*start_index*: 713)
鲁棒性样例	人工复述1	北京现代昂希诺纯电动补贴后多少钱?
	人工复述2	现代昂希诺纯电动版补贴后的价格是多少?

表 1. 标注数据样例

本文在3.1节问题预处理部分，实体识别是用汽车领域专用的实体识别算法，能够识别出车系、车型、品牌、车身参数、配置等领域实体。在3.1节语义原型相似度计算部分，SBERT语义相似度模型需要数据进行训练，为了避免人工标注，本文在网络上爬取百度知道中的提问和相关提问数据，用汽车领域的关键词进行筛选，最终得到约20万组相关问题，约100万条数据。同组问题组合标记为正样本，不同组数据组合标记负样本，构造模型的训练数据和测试数据，训练SBERT语义相似度模型。

数据集	文档数目	问题数目	问题长度	答案长度	文档长度
Train	610	1998	19	8	832
Test	295	748	20	12	920
Robust_test	2312	1998	20	11	920

表 2. 实验数据

4.2 对比实验

为了验证本文提出的数据增强方法的有效性，本文用BERT_base模型作为基准模型进行实验，其中Batch_size为6，Epoch为4，其它超参数保持不变，对比以下各种数据增强方法：

简单数据增强方法EDA(Wei and Zou, 2019)：对原始训练数据集中的问题进行处理（同义词替换、插入、删除、交换位置）得到新问题，随机抽出新问题与原始训练数据中的文档进行组合，构造训练数据。

远程监督增强方法DS(Zhang et al., 2018)：将汽车领域新闻资讯文章按段落进行切分，构建Elasticsearch索引，用汽车领域知识图谱中3万个知识三元组数据进行搜索，将检索到的段落作为文档D，用知识三元组(E_1, R, E_2)中的实体E_1和关系R构建问题Q，实体E_2作为答案A，构建训练数据(Q, D, A)。

语义原型泛化增强方法（PG）：本文所提数据增强方法。

以上三种方法在测试集和鲁棒性测试集上的实验结果如图4和图5所示。横坐标N_{aug}表示添加构造数据的数量，$N_{aug} = 0$表示没有添加构造数据，$N_{aug} = 1$表示添加了原始训练数据1倍数量的构造数据。实验结果表明，在汽车领域数据测试集和鲁棒性测试集中PG方法效果要优于其他两种方法。

如图4在测试集中，PG方法构造的数据对测试集的EM和F1值均有2个点以上的提升，N_{aug}在2至8时，效果最好，N_{aug}超过16时，提升效果有所下降；其他两种方法效果相当，对测试集几乎没有提升效果。对于远程监督方法，汽车领域知识三元组数据量大，但是种类相对较少，构造出来的数据形式相对单一，另外数据构造过程中也会引入较大的噪声，这些因素可能对构造数据质量产生影响，从而影响实验结果；对于EDA构造方法，形式上相对简单，在分类任务中有效果，在阅读理解任务中表现不明显。

如图5在鲁棒性测试集中，三种方法对F1指标均有提升效果，PG的提升效果明显高于其他两种方法，EDA的方法略高于DS的方法。对于EM指标，PG和EDA方法优于DS方法，并且DS方法随着数据量的增加，EM指标呈下降趋势，有此可以看出DS方法构造的数据引入的噪声相对较大，对原始训练数据造成了干扰。

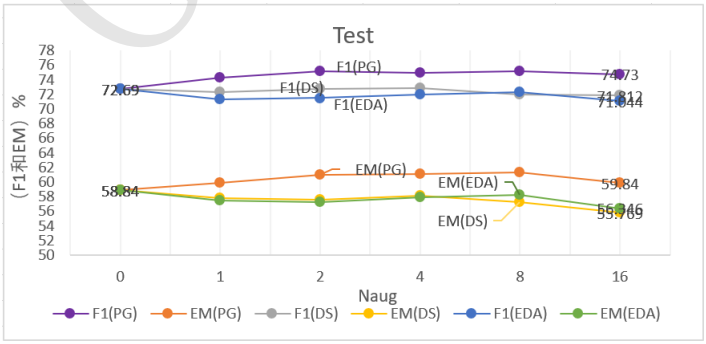


图 4. 在测试集上三种方法对比实验。PG方法提升效果明显优于EDA和DS方法；PG方法在EM和F1指标上均有2个点以上的提升；N_{aug}大于16时，三种方法效果均有下降趋势。

为进一步分析各种方法构造出的训练数据的区别，本文使用原始数据量4倍的构造数据，分别按比重$(0, 0.2, 0.4, 0.6, 0.8, 1)$加入原始训练数据，进行实验。实验结果如图6（a-d），在测试集和鲁棒性测试集中，PG和EDA效果相当，仅使用构造数据就能达到与训练数据接近的效果。DS方法随着原始训练数据的增加，效果逐步提升。DS方法构造的数据完全没有使用原始训练数据，PG和EDA方法构造的数据是在对原始训练数据微调的基础上获取的，因此在仅使

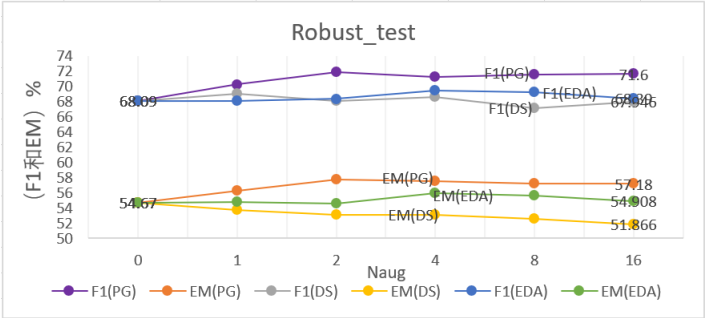
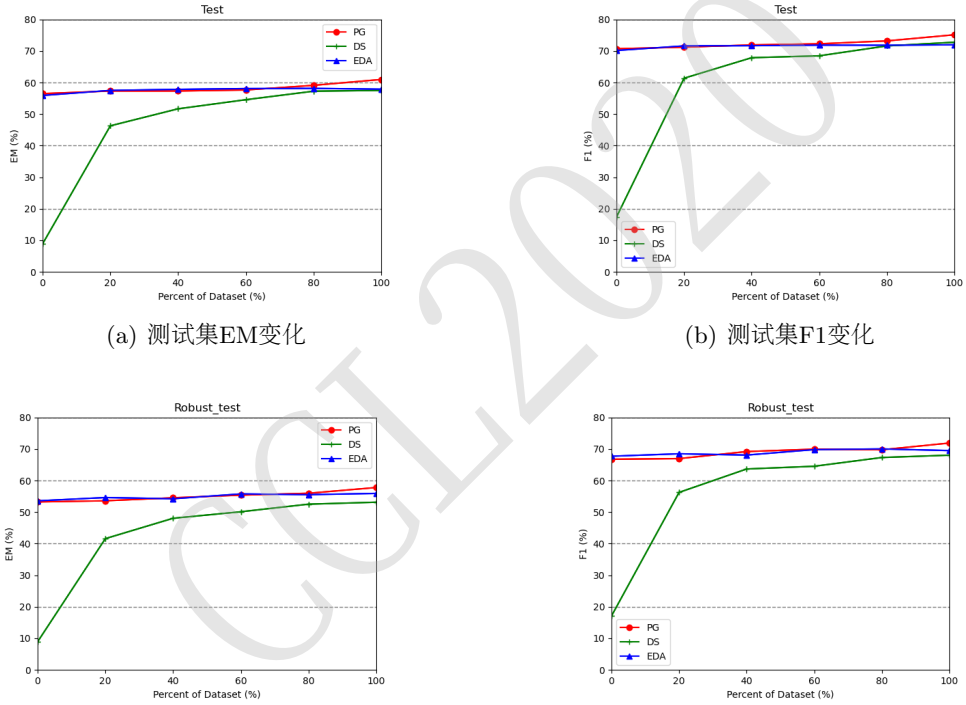


图 5. 在鲁棒性测试集上三种方法对比实验。三种方法对F1值均有提升效果，PG的提升效果明显高于EDA和DS；对于EM指标，PG和EDA方法要优于DS方法，并且DS方法随着数据量的增加，EM指标呈明显下降趋势。

用构造数据时，DS效果明显低于PG和EDA。



(a) 测试集EM变化　(b) 测试集F1变化

(c) 鲁棒性测试集EM变化　(d) 鲁棒性测试集EM变化

图 6. 训练数据占比变化图。图a、b是在测试集上随着训练数据比重增加F1和EM指标的变化，图c、d是在鲁棒性测试集上随着训练数据比重增加F1和EM指标的变化。在测试集和鲁棒性测试集中，PG和EDA效果相当，仅使用构造数据就能达到与训练数据接近的效果。DS方法随着原始训练数据的增加，效果逐步提升；在仅使用构造数据时，DS效果明显低于PG和EDA。

4.3 多模型验证实验

为了验证本文提出的数据增强方法在各种模型上的通用性，本文选择近期在阅读理解任务中表现突出的多个模型进行实验。

BERT模型：BERT模型在阅读理解任务取得突破性的成绩，它采用多层Transformer结构堆叠而成，层数的不同，模型大小不同，本文采用层数为12的BERT_base模型进行微调，验证方法的有效性，其中Batch_size为6，Epoch为4，其他参数不变。

Albert模型：Albert模型在BERT模型基础上进行了改进，通过词嵌入矩阵的分解和隐藏层参数共享，减小模型的参数，提升模型的性能。本文选择与BERT_base模型参数量相当

的Albert_xlarge模型进行实验，其中Batch_size为6，Epoch为4，其他参数不变。

DrQA模型：DrQA模型是一个完整的端到端的阅读理解问答系统，包含文档检索和文档阅读两个模块，本文仅使用文档阅读模块，验证方法的有效性。在实验中，数据预处理采用CoreNLP(Manning et al., 2014)进行分词和实体识别，使用腾讯中文词向量(Song et al., 2018)进行词嵌入，训练参数与原模型一致。

如表3，实验结果表明本文提出的数据增强方法在三个模型上均有效果，测试集和鲁棒性测试集的F1和EM指标都有2个点以上的提升。从模型之间的对比可以看到：Bert、Albert预训练语言模型在阅读理解任务中表现突出，DrQA是非预训练模型，没有经过大量无监督数据的预训练，因此效果较差；在参数量相当的时候，经过改进的Albert模型效果比Bert更好。

模型	数据集	数据增强	F1	EM
Albert	Test	N	74.12	59.01
		Y	76.97	62.30
		Δ	2.85	3.29
	Robust_test	N	70.56	55.79
		Y	73.49	58.80
		Δ	2.93	3.01
BERT	Test	N	72.69	58.84
		Y	74.94	61.08
		Δ	2.25	2.24
	Robust_test	N	68.09	54.67
		Y	71.19	57.49
		Δ	3.10	2.82
DrQA	Test	N	61.80	44.55
		Y	66.00	49.62
		Δ	4.20	5.07
	Robust_test	N	56.45	39.19
		Y	60.35	43.71
		Δ	3.90	4.52

表 3. 数据增强方法在多个模型上的实验结果，其中N表示不使用数据增强，Y表示使用数据增强，Δ表示增加量。

5 结束语

本文提出了一种垂直领域中基于真实用户问题的数据增强方法，该方法对真实用户问题的语义原型进行泛化，构造同义表达问题，从而增强问题的多样性，同时提升构造数据和应用场景中数据的一致性，从而提升模型的准确率和鲁棒性。该方法结合了垂直领域的数据特点和相关技术方法，如：领域实体识别技术，在汽车领域数据集上，验证多种模型，F1和EM指标均能取得2至5个百分点的提升。本文面向垂直领域的数据增强方法对其它各垂直领域都有借鉴作用，具有很大的普适性，下一步将结合本方法，在通用领域数据上进行分析和研究。

参考文献

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*.

Yiming Cui, Ting Liu, Wanxiang Che, Li Xiao, Zhipeng Chen, Wentao Ma, Shijin Wang, and Guoping Hu. 2018. A span-extraction dataset for chinese machine reading comprehension. *arXiv preprint arXiv:1810.07366*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2017. Data augmentation for low-resource neural machine translation. *arXiv preprint arXiv:1705.00440*.

Wee Chung Gan and Hwee Tou Ng. 2019. Improving the robustness of question answering systems to question paraphrasing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6065–6075.

Wei He, Kai Liu, Jing Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, et al. 2017. Dureader: a chinese machine reading comprehension dataset from real-world applications. *arXiv preprint arXiv:1711.05073*.

Yanghoon Kim, Hwanhee Lee, Joongbo Shin, and Kyomin Jung. 2019. Improving neural question generation using answer separation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6602–6609.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

Raul Puri, Ryan Spring, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. 2020. Training question answering models from synthetic data. *arXiv preprint arXiv:2002.09599*.

Yingqi Qu, Jie Liu, Liangyi Kang, Qinfeng Shi, and Dan Ye. 2018. Question answering over freebase via attentive rnn with similarity matrix based cnn. *arXiv preprint arXiv:1804.03317*, 38.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. 2018. Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 175–180.

Sandeep Subramanian, Tong Wang, Xingdi Yuan, Saizheng Zhang, Yoshua Bengio, and Adam Trischler. 2017. Neural models for key phrase detection and question generation. *arXiv preprint arXiv:1706.04560*.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 189–198.

Jason Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension. *arXiv preprint arXiv:1804.09541*.

Hongzhi Zhang, Xiao Liang, Guangluan Xu, Kun Fu, Feng Li, and Tinglei Huang. 2018. Factoid question answering with distant supervision. *Entropy*, 20(6):439.

Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. 2018. Paragraph-level neural question generation with maxout pointer and gated self-attention networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3901–3910.

Haichao Zhu, Li Dong, Furu Wei, Wenhui Wang, Bing Qin, and Ting Liu. 2019. Learning to ask unanswerable questions for machine reading comprehension. *arXiv preprint arXiv:1906.06045*.

安波, 韩先培, and 孙乐. 2018. 融合知识表示的知识库问答系统. 中国科学:信息科学, 48(11):59–70.

白龙, 靳小龙, 席鹏弼, and 程学旗. 2019. 基于远程监督的关系抽取研究综述. 中文信息学报, 33(10):10–17.

融入对话上文整体信息的层次匹配回应选择

司博文
苏州大学
计算机科学与技术学院
20185227065@stu.suda.edu.cn

孔芳*
苏州大学
计算机科学与技术学院
kongfang@suda.edu.cn

摘要

对话是一个顺序交互的过程，回应选择旨在根据已有对话上文选择合适的回应，是自然语言处理领域的研究热点。已有研究取得了一定的成功，但仍然存在两个突出的问题。一是现有的编码器在挖掘对话文本语义信息上尚存在不足；二是只考虑每一回合对话与备选回应之间的关系，忽视了对话上文的整体语义信息。针对问题一，本文借助多头自注意力机制有效捕捉对话文本的语义信息；针对问题二，整合对话上文的整体语义信息，分别从单词、句子以及整体对话上文三个层次与备选回应进行匹配，充分保证匹配信息的完整。在Ubuntu Corpus V1和Douban Conversation Corpus数据集上的对比实验表明了本文给出方法的有效性。

关键词： 回应选择；多头自注意力机制；交叉注意力机制；对话整体信息

Learning Overall Dialogue Information for Dialogue Response Selection

Bowen Si
School of Computer
Science and Technology
Soochow University
20185227065@stu.suda.edu.cn

Fang Kong *
School of Computer
Science and Technology
Soochow University
kongfang@suda.edu.cn

Abstract

Dialogue is a sequential interactive process. Response selection aims to select the appropriate response based on the existing dialogue, which is a research hotspot in the field of natural language processing. Existing researches have achieved some success, but there are two problems. First, the existing encoders still have deficiencies in mining the semantic information of the dialogue text; second, the existing research only considers the relationship between each round of dialogue and the alternative response, ignoring the overall semantic information above the dialogue. For problem one, this article uses the multi-head self-attention mechanism to capture the semantic information of the dialogue text effectively; for problem two, integrating semantic information above the dialogue, then matching context and response from the words, sentences, and the overall dialogue levels which can fully guarantee the completeness of matching information. Experiments on the Ubuntu Corpus V1 and Douban Conversation Corpus datasets exhibit the effectiveness of the method presented in this article.

基金项目：国家自然科学基金面上项目（61876118）；国家自然科学基金人工智能应急管理项目(61751206)
*通信作者：kongfang@suda.edu.cn

Keywords: Response selection , Self-attention , Cross-attention , Overall Dialogue Information

1 引言

人机对话（human-computer conversations(Saygin and Cicekli, 2002)），旨在促进人类与机器自然交流，是自然语言处理（natural language process,NLP(Manning et al., 1999)）领域的关键任务。该任务要求模型依据已有的对话上文，产生与对话上文相匹配的回应。当前，有两种产生回应的方法：生成式和检索式。生成式又称回应生成（response generation(Ritter et al., 2011)），旨在使用自然语言生成技术生成与对话上文相匹配的回应；检索式又称回应选择（response selection(Rowe et al., 2000)），旨在从备选回应中为对话上文选出合适的回应。这两种方法产生的回应都与对话上文密切相关，因此如何更好的整合对话上文的信息一直是这两个任务的研究重点。本文主要关注回应选择任务，因为，回应选择任务存在信息量丰富和对话内容流畅等优势。许多工业界的产品采用的都是检索式对话系统，例如微软的小冰(Zhou et al., 2020)、阿里的小蜜(Li et al., 2017)等。

已有的回应选择方法存在对话历史的语义挖掘不充分，以及只注重每一回合对话与备选回应之间的关联信息，从而忽视了完整对话上文信息的问题。针对这两个问题，本文提出了融入整体对话上文信息的回应选择方法，记为IODIRS（Integrate the Overall Dialogue Information for Response Selection），具体而言，首先，使用多头自注意力机制（Multi-head attention Mechanism(Vaswani et al., 2017)）挖掘对话文本的潜在语义信息；其次，计算每一回合对话与备选回应在单词和回合级别的关联信息，并将每个回合和备选回应的关联信息拼接到一个矩阵中；接着，使用交叉注意力机制（Cross-attention Mechanism(Hao et al., 2017)）从上到下计算对话上文的整体语义信息，挖掘整体关联信息的同时保证对话历史的序列特性；之后，将对话上文的整体语义信息与备选的回应进行关联，并将关联的信息拼接到上述矩阵中；最后整合各类信息进行最终的回应选择。该方法在深度挖掘每一回合对话文本语义信息的同时将对话上文的整体信息融入该任务中，从而提升回应选择的性能。

本文的贡献包括如下几点：（1）提出融入对话上文整体语义信息的回应选择模型。（2）使用多头自注意力机制挖掘每一回合对话的语义信息。（3）使用交叉注意力机制从上到下挖掘对话上文之间的关联信息，并将其整合，保证对话上文序列特性的前提下，整合对话上文的整体信息，并通过Ubuntu Corpus V1(Lowe et al., 2015)和Douban Conversation Corpus(Wu et al., 2017)数据集上的实验验证模型的有效性。

本文的组织结构如下：第二节介绍相关工作；第三节阐述IODIRS模型；第四节介绍具体的实验设置，并对实验结果进行分析；第五节对本文进行总结，并给出下一步的研究计划。

2 相关研究

随着深度学习的发展，建立一个数据驱动的人机对话系统越来越受到关注。已有的研究可以划分为生成式对话和检索式对话。生成式对话又称回应生成，旨在使用自然语言生成技术生成与对话上文相关的回应。检索式对话又称回应选择，旨在从备选回应中选出与对话上文最相关的回应。相较于生成式对话，检索式对话的信息更丰富，过程更流畅，所以本文重点研究检索式对话。

早期对话回应选择的研究主要集中于短文本的单回合对话。Wang (2015)提出了相关数据集和一个基于向量空间和语义匹配的方法。Ji(2014)提出使用深度神经网络来匹配对话上文和回应之间的语义信息的方法。Wu(2016)提出了一个用于短文本回应选择的主题感知卷积神经网络框架。这些方法在单回合回应选择任务取得了一定成果，但是无法解决多回合问题。

当前的研究主要集中于多轮对话的回应选择任务。该任务更具挑战性，因为模型需要整合整个对话上文中信息。目前的研究主要有以下三种方法。

基于编码的方法，Lowe (2015)使用RNN编码对话上文和备选回应的方法，此类方法又被称为平行编码方法。不久，Kadlec (2015)研究了不同种类编码器在平行编码网络上的性能。Yan(2016)采用了另一种做法，他们使用一个CNN计算对话上文和备选回应的匹配分数。Zhou(2016)采用了两个并行的编码器，一个处理单词级别的信息，另一个处理话语级别的信息。这些方法处理信息相对简单，没有充分挖掘对话上文和回应之间的深层语义信息。

基于匹配的方法，Wang(2016)提出MV-LSTM模型，通过基于LSTM的注意力加权句子表示法来提升模型的性能。Tan (2015)提出的QA-LSTM，使用一个简单的注意力机制与LSTM编码器相结合的方法。这些方法在挖掘文本信息方面取得一定的成果，但是将对话上文拼接成一个长文的做法使得文本变得过长，现有的编码器处理文本的能力有限，很难学习到有效的信息，同时基于匹配的方法没有将得到的信息进行充分整合，回应选择任务要求模型选出合适的回应，模型必须能够充分整合出对话上文和备选回应之间的关联信息。

层次匹配的方法，Wu(2017)提出顺序匹配网络（Sequential Matching Network，SMN），将备选回应分别与对话上文中的每一次对话匹配，Zhang(2018)提出深度话语汇聚网络（deep utterance aggregation network，DUA），该网络细化处理对话，同时使用自注意力机制来寻找每次对话中的重要信息。Tao(2019)提出多种表示聚合模型（multi-representation fusion network，MRFN），该模型将对话文本在多个粒度表示，之后将每个粒度上的信息进行聚合。以上做法主张将每一次对话与备选回应进行交互，之后将交互信息进行聚合。虽然使用了对话前文信息，但是忽视了对话上文的整体语义信息。实际中，对话之间存在很多指代和省略。忽视这些信息很难有效捕捉到完整的对话上文信息。

现有的工作取得了一定成果并有效推动了多回合对话回应选择任务的发展，但已有研究存在挖掘文本语义信息能力不足和忽视对话上文整体语义信息的问题。对此，本文提出IODIRS模型，借助多头自注意力机制有效挖掘对话文本的潜在语义信息，借助交叉注意力机制从上至下挖掘对话之间的关联信息，最后将关联信息进行整合进而得到对话上文的完整语义信息。

3 IODIRS模型

多回合对话回应选择任务要求模型在备选回应池中为对话上文选择合适的回应。本文将多回合对话回应选择任务转化为分类任务，要求模型在得到对话上文和备选回应的情况下判断备选回应是否是对话上文的合适回应。

3.1 任务描述

给定数据集$D = \{(y_i, C_i, r_i)\}_{i=1}^{N}$，其中$C_i = \{u_{i1}, u_{i2}, ..., u_{iL}\}$表示对话上文，其中对话上文有$L$回合的对话，$r_i$ 表示备选回应，$y_i \in \{0, 1\}$表示标签，$y_i = 1$ 表示r_i 是C_i 合适的回应，$y_i = 0$ 表示r_i 不是合适的回应。检索式对话的目标是根据数据集D 训练出一个匹配模型$s(.,.)$。对于任意一个对话上文-回应对(C, r)，匹配模型都能给出对话上文C和回应r 之间的匹配分数。

3.2 模型概述

图1为IODIRS模型框架图，该模型主要由以下三个部分组成：

1）编码层：使用多头自注意力机制对每一回合的对话进行编码，挖掘其潜在语义信息。

2）聚合层：首先，使用交叉注意力机制从上到下计算对话上文的整体语义信息，同时计算每一回合对话与备选回应在单词和回合级别的相似矩阵。之后，使用CNN整合每一回合对话的单词和回合级别相似矩阵获取其匹配信息，最后，将每一回合对话与备选回应的匹配信息与对话整体与备选回应的匹配信息进行整合，得到最终的匹配信息。

3）输出层：融合得到的语义信息，并对结果进行预测。

3.3 编码层

诸多研究证明，注意力机制在自然语言处理领域是有效的。Vaswani(2017)提出一个基于注意力机制的生成网络Transformer。相比RNN模型，Transformer不仅取得了更好的生成结果，而且其训练速度也非常快，因为其注意力计算可以并行。已有工作显示注意力机制在挖掘文本潜在语义信息上的卓越性能。因此，我们使用多头自注意力机制对文本进行编码。

任务中，对话上文可以表示为：$C = (u_1, ..., u_L)$ ，其中，$u_i = (u_{i1}, ..., u_{im})$ ，备选的回应可以表示为：$r = (r_1, ..., r_n)$，L表示对话上文的回合数，m表示每一回合对话文本的长度，n表示备选回应的长度。首先，我们使用预训练词向量$R^{d_e \times V}$将每个词转为其对应的词向量。特别的，如果某个词在表中不存在，则赋予一个随机值。此时，对话上文C中的一回合对话u_i和回

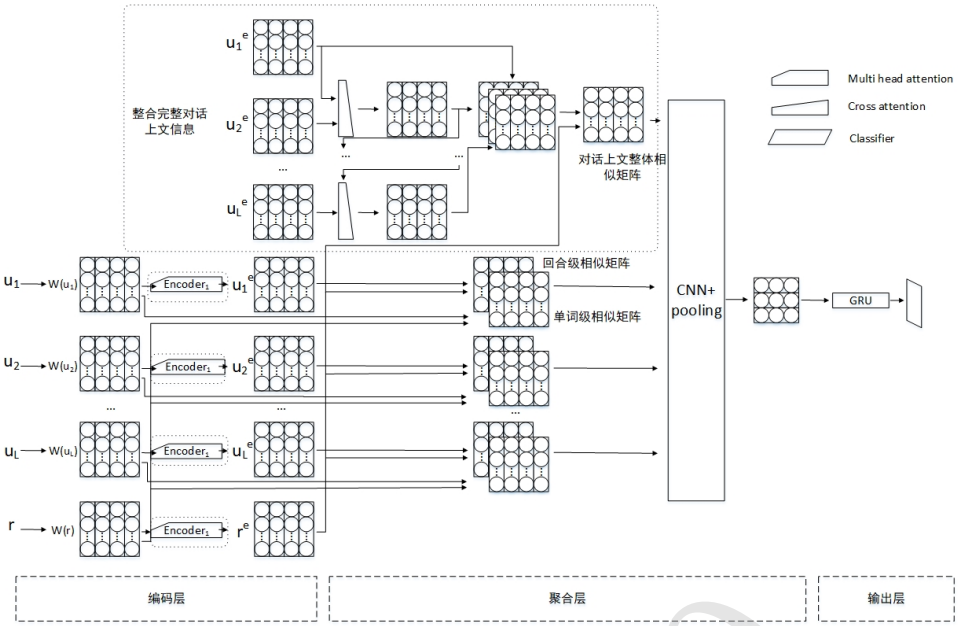


图 1 IODIRS模型

应r变成词向量序列$W(u_i) = [W(u_{i1}), W(u_{i2}), ..., W(u_{im})]$，$W(r) = [W(r_1), W(r_2), ..., W(r_n)]$。

为了得到对话上文中每一回合对话和备选回应的语义向量u_i^e，r^e，本文将对话上文和回应向量序列传入$MultiHead$，具体计算如式（1）～（2）。

$$u_i{}^e = MultiHead_1(W(u_i), W(u_i), W(u_i)) \tag{1}$$

$$r^e = MultiHead_1(W(r), W(r), W(r)) \tag{2}$$

其中，i表示对话上文中的第i个回合的对话文本，式（1）和式（2）共享一个$MultiHead_1$。

3.4 聚合层

回应选择任务的重点在于如何有效的整合对话上文的信息，并准确找到对话上文与备选回应之间的联系。正确的回应一定与对话上文中的相关信息密切相关。因此，模型能否有效整合出对话上文和备选回应之间的联系是确定回应是否是正确回应的关键步骤。已有的层次编码方法采用将每一回合对话与备选回应进行交互，并把交互的信息进行整合的方法。这样做虽然能够捕捉到每一回合对话与备选回应之间的联系，但是忽视了对话上文的整体语义信息。实际生活中，对话之间存在较多的省略和指代现象，倘若忽视这些现象，模型很难捕捉到所有关键信息。为充分挖掘对话上文与备选回应之间的语义联系，本文从对话上文整体、每一回合对话单词级别和每一回合对话整体语义三个层面将对话上文与备选回应进行交互。

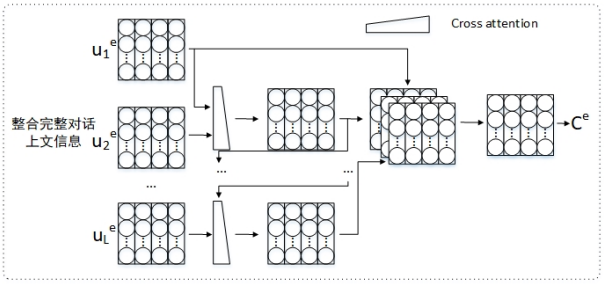


图 2 整合完整对话上文信息

3.4.1 对话上文整体语义挖掘

对于对话上文整体语义信息的挖掘，本文使用交叉注意力机制从上到下捕获对话之间的关联信息，并把关联到的信息进行整合，进而得到完整的对话上文语义信息E(C)。具体如图2所示具体计算如公式（3）～（4）所示:

$$u_{i+1}{}^{c} = CrossAttention(u_i^c, u_{i+1}^e) \tag{3}$$

$$C^e = mean_1([u_1^c; u_2^c; ...; u_L^c]) \tag{4}$$

其中$u_1^c = u_1^e$，[.] 表示向量的拼接，L表示对话上文中对话的回合数。

交叉注意力机制工作原理如下图3所示:

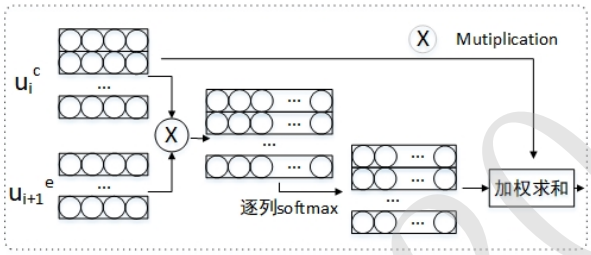


图 3 交叉注意力机制

交叉注意力权重的计算如式（5）。

$$e_{jk} = (u_{ij}^c)^T u_{(i+1)k}^e \tag{5}$$

本文使用软对齐获取对话回合之间的局部相关性，其通过式（8）中的注意力矩阵$e \in R^{m\times m}$计算得到。$u_{(i+1)}^e$中第k个单词的隐藏层向量$u_{(i+1)k}^e$与u_i^c中相关的语义部分被标识为向量$u_{(i+1)k}^l$，称之为$u_{(i+1)k}^e$的对偶向量，这个向量表示所有相关信息的加权和。具体计算如式（6）。

$$\beta_{jk} = \frac{exp(e_{jk})}{\sum_{i=1}^{m} exp(e_{ik})}, u_{(i+1)k}^l = \sum_{j=1}^{m} \beta_{jk} u_{ij}^c \tag{6}$$

其中，$\beta \in R^{m\times m}$表示标准化注意力权重矩阵。

为了融合已经得到的所有相关信息，本文使用启发式匹配方法处理得到的语义向量。具体如式（7）～（9）。

$$u_{(i+1)k1}^l = u_{(i+1)k}^e - u_{(i+1)k}^l \tag{7}$$

$$u_{(i+1)kM}^l = u_{(i+1)k}^e \cdot u_{(i+1)k}^l \tag{8}$$

$$u_{(i+1)k}^c = F([u_{(i+1)k}^e; u_{(i+1)k}^l; u_{(i+1)k1}^l; u_{(i+1)kM}^l]) \tag{9}$$

其中，[.]表示向量的拼接操作，F是使用$ReLU$ 降低维度的单层前馈神经网络。

3.4.2 三个层次信息与备选回应的交互

为了整合有效信息，本文从对话上文整体语义，每一回合对话单词级别和每一回合对话整体语义三个层面将对话上文与备选回应进行交互。

首先，我们先计算每一回合对话文本与备选回应在单词和回合文本级别的相似信息，并将结果进行整合，得到该回合文本与备选回应的匹配向量ul_i。具体如公式（10）～（13）所示。

$$uwl_i = W(u_i) \cdot (W(r))^T \tag{10}$$

$$usl_i = u_i^e \cdot A \cdot (r^e)^T \tag{11}$$

$$ulc_i = F(Conv2d(Stack(uwl_i, usl_i))) \tag{12}$$

$$ul_i = Maxpool(ulc_i) \tag{13}$$

其中，T表示矩阵的转置，$A \in R^{m \times m}$ 表示线性变化的参数，$Stack$ 表示矩阵的堆叠，$Conv2d$(Chudanov et al., 1999)表示卷积神经网络，F 表示层前馈神经网络，$Maxpool$表示池化。

接着，我们使用同样的方法计算对话上文整体与备选回应之间的相似信息。具体如公式（14）～（16）所示。

$$Csl = C^e \cdot A \cdot (r^e)^T \tag{14}$$

$$Clc = F(Conv2d(Stack(Csl))) \tag{15}$$

$$Cl = Maxpool(Clc) \tag{16}$$

其中，$Stack$表示矩阵的堆叠，$Conv2d$表示卷积神经网络，F表示层前馈神经网络，$Maxpool$表示池化。

最后我们将每一回合的匹配信息以及对话上文整体匹配信息整合。具体如公式（17）所示。

$$match = Stack(ul_1, ul_2, ..., ul_L, Cl) \tag{17}$$

其中，$Stack$表示矩阵的堆叠，L 表示对话上文的回合数。

3.5 输出层

获取到对话上文与备选回应之间匹配信息$match$ 之后，输出层的主要工作是对上述匹配信息进行进一步整合，得到最终的匹配信息，并得到最终的匹配结果。

我们采用GRU(Dey and Salem, 2017)对$match$进行进一步编码，并使用最后一层结果作为最终的匹配向量。具体操作如公式（18）～（19）所示。

$$last = GRU(match) \tag{18}$$

$$label = Sigmoid(W_1 \cdot last + b_1) \tag{19}$$

其中，$label$ 表示预测的结果，W_1 和b_1分别表示$Sigmoid$层的权重和偏置。

3.6 优化策略

在模型训练的过程中，本文选择二分类交叉熵误差作为损失函数，具体计算如公式（20）～（21）：

$$Loss(y\prime, y) = \sum_{i=1}^{S} l_i \tag{20}$$

$$l_i = -w_i[y_i log y\prime_i - (1 - y_i) log(1 - y\prime_i)] \tag{21}$$

其中，y表示真实答案，$y\prime$是模型预测的概率，S是训练样本的总数。同时，本文使用$Adam$（Adaptive Moment Estimation(Kingma and Ba, 2015)）算法优化模型参数。

4 实验设置与结果分析

4.1 实验数据集

本文使用公开数据集Ubuntu Corpus V1和Douban Conversation Corpus数据集验证所提出的方法。Ubuntu Corpus V1数据集中的对话主要是关于Ubuntu系统故障排除的多回合英文对话。Douban Conversation Corpus数据集是从豆瓣中获取的开放域对话，其构建方式与Ubuntu Corpus V1相似。两个数据集的具体分布如表1所示。

	Ubuntu			**Douban**		
Name	Train	Val	Test	Train	Val	Test
样本数	1M	500K	500K	1M	50K	10K
备选回应数	2	10	10	2	2	10
正例数	1	1	1	1	1	1.18
回合数	10.13	10.11	10.11	6.69	6.75	6.45
回合长度	11.35	11.34	11.37	18.56	18.50	20.74

表 1　Ubuntu Corpus V1和Douban Conversation Corpus分布数据

4.2 实验设置

实验采用Pytorch 0.4.0 框架，并用$NVIDIA$的1080GPU进行加速。具体的模型参数配置为：使用word2vec预训练词向量进行初始化，word dim 为200。样本的回合数设置为10，$MultiHead$的输入是一个形状为$[batchsize, seqlength, hidden]$ 的张量，其中第一个维度表示$batchsize$ ，训练中不同数据集的$batchsize$设置为不同，Ubuntu设置为200。Douban设置为150；第二维表示$batchsize$ 个句子中最大句子长度，设置为50，；第三维表示隐藏层维数，实验中，$hidden$ 设置为200。$MultiHead$的输出是一个形状为$[batchsize, seqlength, hidden]$的张量，使用$Adam$（Adaptive Moment Estimation）算法优化模型参数，学习率lr设置为0.004，$dropout$ 设置为0.5，$MultiHead$ 的多头设置为4，损失函数为交叉熵损失函数，$Conv2d$ 的卷积核设置为（3，3）。本文着重验证所提方法的有效性，并没有刻意的关注模型的极限性能。因此，没有刻意对模型进行调参。

4.3 实验结果

本文主要使用数据集作者指出的评价指标作为模型性能的评价指标，其中Ubuntu Corpus V1(Lowe et al., 2015)使用$R_{10}@K$(Lowe et al., 2015)作为评价指标，Douban Conversation Corpus(Wu et al., 2017)使用$R_{10}@K$，MAP，MRR和$P@1$(Wu et al., 2017)作为评价指标。

本文选取的对比模型有：

基于句子编码的方法：BiLSTM(Lowe et al., 2015)。首先，将对话上文和回应编码；然后，计算对话上文和回应之间的语义相似度。

基于序列匹配的方法：MV-LSTM(Wang and Jiang, 2016)和Match-LSTM(Wang and Jiang, 2017)。将对话上文拼接成一个长文，使用注意力机制计算对话上文和回应之间单词级别的信息。

复杂的基于层次的方法：SMN(Wu et al., 2017)，将备选回应分别与对话上文中的每一次对话匹配，之后将匹配的信息进行聚合。DUA(Zhang et al., 2018)，细化处理对话，同时使用自注意力机制来寻找每次对话中的重要信息。MRFN(Tao et al., 2019)，使用多表示网络将文本特征融合。

表2给出了本文模型和各个模型的实验结果。

从表2给出的结果可以看到：

本文的方法取得了相当可观的性能。普通的层次编码方法着重强调每一轮对话与备选回应之间的关系，没有重视对话之间的联系，而实际上，人们对话之间的联系是很密切的。考虑到对话上文整体语义信息的重要性，本文使用交叉注意力机制从上到下挖掘对话上文的语义信

息，并对其进行整合。之后，从对话上文整体、每一回合对话单词级别和每一回合对话整体语义三个层面将对话上文与备选回应进行交互。相比普通的层次编码模型，本文的模型在复杂度上略高，但取得了高于普通层次编码模型的效果；同时与采用多级别表示文本特征的MFRN模型相比，本文的模型在复杂度上较低，却取得了与之相当的性能。

	Ubuntu Corpus			**Douban Conversation Corpus**					
模型	R_{10}@1	R_{10}@2	R_{10}@5	MAP	MRR	P@1	R_{10}@1	R_{10}@2	R_{10}@5
BiLSTM	0.630	0.780	0.944	0.479	0.514	0.313	0.184	0.330	0.716
MV-LSTM	0.653	0.804	0.946	0.498	0.538	0.348	0.202	0.351	0.710
Match-LSTM	0.653	0.799	0.944	0.500	0.537	0.345	0.202	0.348	0.720
SMN	0.726	0.847	0.961	0.529	0.569	0.397	0.233	0.396	0.724
DUA	0.752	0.868	0.962	0.551	0.599	0.421	0.243	0.421	0.780
MFRN	0.786	0.886	0.976	0.571	0.617	0.448	0.276	0.435	0.783
IODIRS	0.782	**0.886**	0.973	0.561	**0.617**	0.427	0.258	**0.436**	**0.791**

表 2　各个模型的结果

4.4　实验分析

为了验证不同模块的作用，本文设置了以下对比实验。

B：使用GRU编码，将每一回合对话与备选回应匹配，使用CNN聚合匹配后的信息，不考虑对话上文整体语义信息。

B+整体：使用GRU编码，使用交叉注意力机制，从上至下整合对话上文语义信息。之后从三个层面将对话上文与备选回应进行交互。

B+多头：使用MulitHead编码，将每一回合对话与备选回应匹配，使用CNN聚合匹配后的信息，不考虑对话上文整体语义信息。

IODIRS：使用MulitHead编码，使用交叉注意力机制，从上至下整合对话上文语义信息。之后从三个层面将对话上文与备选回应进行交互。

实验结果见表3。

	Ubuntu Corpus			**Douban Conversation Corpus**					
模型	R_{10}@1	R_{10}@2	R_{10}@5	MAP	MRR	P@1	R_{10}@1	R_{10}@2	R_{10}@5
B	0.755	0.865	0.961	0.558	0.597	0.425	0.255	0.424	0.753
B+多头	0.761	0.866	0.963	0.567	0.609	0.439	0.261	0.435	0.782
B+整体	0.773	0.877	0.965	**0.569**	0.608	**0.443**	**0.262**	0.434	0.780
IODIRS	**0.782**	**0.886**	**0.973**	0.561	**0.617**	0.427	0.258	**0.436**	**0.791**

表 3　详细实验对比结果

4.4.1　多头自注意力机制效用分析

比较B和B+多头的结果，可以得出使用多头自注意力机制编码的模型相比没有使用多头自注意力机制的模型，模型性能在各个评价指标上均有近1个点的提升，说明多头自注意力机制在挖掘文本潜在语义信息方面效果明显，因为多头自注意力机制可以从多角度、多层次深度挖掘文本的语义信息。

4.4.2　对话上文整体语义分析

比较B和B+整体的结果，可以得出模型在没有整合对话上文整体语义信息的情况下，性能有着近1个点的下降。说明整合到的对话上文整体语义信息在匹配回应上起到了关键作用。在实际生活中，对话之间存在较多的省略和指代现象，只有交互相关回合的对话才能有效捕捉到这些信息。而普通的层次匹配方法只重视每一回合对话与备选回应的匹配，忽视了这些信息。

4.4.3 整合后模型效用分析

上述分析了多头自注意力机制作为编码器以及整合对话上文整体语义信息对模型性能的影响。为了探究二者结合的性能，我们搭建了IODIRS模型。实验显示，相比B+多头，IODIRS模型在各个指标上均取得了提升。相比B+整体，IODIRS模型在准确率上略有下降，但是模型的整体性能却有所提升，这是因为多头自注意力机制可以挖掘到文本中不同层次，不同角度的信息，过于丰富的信息可能导致模型准确率的降低，却能提升模型整体的性能。

4.5 样例分析

为了具体说明各个模块的作用，本文验证了四个模型在一个样例上的输出，各个模型的预测结果见表4:

	样例1	B	B+多头	B+整体	IODIRS
speaker A	Hi I am looking to see what packages are installed on my system, I don't see a path, is the list being held somewhere else?				
speaker B	Try dpkg - get-selections				
speaker A	What is that like? A database for pack-ages instead of a flat file structure?				
answer	**dpkg is the debian package manager - Get - selections simply shows you what packages are handed by it**				
预测结果		**0.485**	**0.512**	**0.735**	**0.855**

表 4　对比模型在样例1上的结果

从表4中各个模型的预测结果，我们可以清晰的看出，在使用多头自注意力机制进行编码后，模型预测的准确率有着一定的提升。但因为忽视了对话之间的语义信息，未能有效的捕捉对话之间的指代信息，如“that”代指什么。在整合了对话上文整体语义信息之后，模型的准确率有着明显的提升。而本文提出的模型在使用多头自注意力机制进行编码的同时将全文信息进行整合，给出了最精确的预测结果。

5 结论

本文借助多头注意力机制多视角挖掘潜在语义，借助交叉注意力从上到下挖掘对话上文的整体语义信息，并从对话上文整体语义，每一回合对话单词级别和每一回合对话整体语义三个层面将对话上文与备选回应进行交互。据此构建了一个层次匹配对话回应选择模型。实验证明，本文的模型能够提升对话回应选择的性能。

本文将对话全文信息进行整合，考虑到全文的信息比较多，容易造成信息的冗余。未来我们将提升模型对文本信息关键信息的挖掘能力，准确寻找到相关信息，高效整合对话上文的信息。提高模型效率的同时，进一步提升回应选择的性能。

参考文献

VV Chudanov, AE Aksenova, VA Pervichko, VF Strizhov, PN Vabishchevich, and AG Churbanov. 1999. Current status and validation of conv2d and 3d code. Technical report.

Rahul Dey and Fathi M. Salem. 2017. Gate-variants of gated recurrent unit (GRU) neural networks. In *IEEE 60th International Midwest Symposium on Circuits and Systems, MWSCAS 2017, Boston, MA, USA, August 6-9, 2017*, pages 1597–1600. IEEE.

Yanchao Hao, Yuanzhe Zhang, Kang Liu, Shizhu He, Zhanyi Liu, Hua Wu, and Jun Zhao. 2017. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting*

of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers, pages 221–231. Association for Computational Linguistics.

Zongcheng Ji, Zhengdong Lu, and Hang Li. 2014. An information retrieval approach to short text conversation. *CoRR*, abs/1408.6988.

Rudolf Kadlec, Martin Schmid, and Jan Kleindienst. 2015. Improved deep learning baselines for ubuntu corpus dialogs. *CoRR*, abs/1510.03753.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Feng-Lin Li, Minghui Qiu, Haiqing Chen, Xiongwei Wang, Xing Gao, Jun Huang, Juwei Ren, Zhongzhou Zhao, Weipeng Zhao, Lei Wang, Guwei Jin, and Wei Chu. 2017. *AliMe Assist* : An intelligent assistant for creating an innovative e-commerce experience. In Ee-Peng Lim, Marianne Winslett, Mark Sanderson, Ada Wai-Chee Fu, Jimeng Sun, J. Shane Culpepper, Eric Lo, Joyce C. Ho, Debora Donato, Rakesh Agrawal, Yu Zheng, Carlos Castillo, Aixin Sun, Vincent S. Tseng, and Chenliang Li, editors, *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017*, pages 2495–2498. ACM.

Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the SIGDIAL 2015 Conference, The 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2-4 September 2015, Prague, Czech Republic*, pages 285–294. The Association for Computer Linguistics.

Christopher D Manning, Christopher D Manning, and Hinrich Schütze. 1999. *Foundations of statistical natural language processing*. MIT press.

Alan Ritter, Colin Cherry, and William B Dolan. 2011. Data-driven response generation in social media. In *Proceedings of the conference on empirical methods in natural language processing*, pages 583–593. Association for Computational Linguistics.

James B Rowe, Ivan Toni, Oliver Josephs, Richard SJ Frackowiak, and Richard E Passingham. 2000. The prefrontal cortex: response selection or maintenance within working memory? *Science*, 288(5471):1656–1660.

Ayse Pinar Saygin and Ilyas Cicekli. 2002. Pragmatics in human-computer conversations. *Journal of Pragmatics*, 34(3):227–258.

Ming Tan, Bing Xiang, and Bowen Zhou. 2015. Lstm-based deep learning models for non-factoid answer selection. *CoRR*, abs/1511.04108.

Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. Multi-representation fusion network for multi-turn response selection in retrieval-based chatbots. In J. Shane Culpepper, Alistair Moffat, Paul N. Bennett, and Kristina Lerman, editors, *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM 2019, Melbourne, VIC, Australia, February 11-15, 2019*, pages 267–275. ACM.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

Shuohang Wang and Jing Jiang. 2016. Learning natural language inference with LSTM. In Kevin Knight, Ani Nenkova, and Owen Rambow, editors, *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 1442–1451. The Association for Computational Linguistics.

Shuohang Wang and Jing Jiang. 2017. Machine comprehension using match-lstm and answer pointer. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Yu Wu, Wei Wu, Zhoujun Li, and Ming Zhou. 2016. Topic augmented neural network for short text conversation. *CoRR*, abs/1605.00090.

Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2017. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 496–505. Association for Computational Linguistics.

Rui Yan, Yiping Song, and Hua Wu. 2016. Learning to respond with deep neural networks for retrieval-based human-computer conversation system. In Raffaele Perego, Fabrizio Sebastiani, Javed A. Aslam, Ian Ruthven, and Justin Zobel, editors, *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, SIGIR 2016, Pisa, Italy, July 17-21, 2016*, pages 55–64. ACM.

Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018. Modeling multi-turn conversation with deep utterance aggregation. In Emily M. Bender, Leon Derczynski, and Pierre Isabelle, editors, *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 3740–3752. Association for Computational Linguistics.

Xiangyang Zhou, Daxiang Dong, Hua Wu, Shiqi Zhao, Dianhai Yu, Hao Tian, Xuan Liu, and Rui Yan. 2016. Multi-view response selection for human-computer conversation. In Jian Su, Xavier Carreras, and Kevin Duh, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 372–381. The Association for Computational Linguistics.

Li Zhou, Jianfeng Gao, Di Li, and Heung-Yeung Shum. 2020. The design and implementation of xiaoice, an empathetic social chatbot. *Comput. Linguistics*, 46(1):53–93.

一种结合话语伪标签注意力的人机对话意图分类方法

丁健德, 黄沛杰*, 许嘉宝, 彭佑铭
华南农业大学, 数学与信息学院, 广东广州, 510642
bighead@stu.scau.edu.cn, pjhuang@scau.edu.cn, xujiabao@stu.scau.edu.cn,
ympengcoder@stu.scau.edu.cn

摘要

在人机对话中，系统需要通过意图分类判断用户意图，再触发相应的业务类型。由于多轮人机对话具有口语化、长文本和特征稀疏等特点，现有的文本分类方法在人机对话意图分类上还存在较大困难。本文在层次注意力网络(hierarchical attention networks, HAN)基础上，提出了一种结合话语伪标签注意力的层次注意力网络模型PLA-HAN (HAN with utterance pseudo label attention)。PLA-HAN通过优选伪标签集、构建单句话语意图识别模型以及设计话语伪标签注意力机制，识别单句话语意图伪标签，并计算话语伪标签注意力。进而将单句话语伪标签注意力嵌入到HAN的层级结构中，与HAN中的句子级别注意力相融合。融合了单句话语意图信息的句子级注意力使模型整体性能得到进一步的提升。我们在中国中文信息学会主办的"客服领域用户意图分类评测比赛"的评测语料上进行实验，实验结果证明PLA-HAN模型取得了优于HAN等对比方法的意图分类性能。

关键词： 意图分类 ； 伪标签注意力 ； 层次注意力网络 ； 人机对话

A Human-machine Dialogue Intent Classification Method using Utterance Pseudo Label Attention

Jiande Ding, Peijie Huang*, Jiabao Xu, Youming Peng
College of Mathematics and Informatics, South China Agricultural University, China
bighead@stu.scau.edu.cn, pjhuang@scau.edu.cn, xujiabao@stu.scau.edu.cn,
ympengcoder@stu.scau.edu.cn

Abstract

In human-machine dialogue system, it needs to judge the user's intent through the intent classification, and then triggers the corresponding business type. Due to the characteristics of colloquialization, longer texts and sparse features of multi-turn dialogues, the existing classification methods still have great difficulties in the classification of human-machine dialogue intent. Based on hierarchical attention networks (HAN), we propose PLA-HAN model that combines the utterance pseudo-label attention. Through selecting utterance intent set, constructing utterance intent detection model and designing an utterance pseudo-label attention mechanism, PLA-HAN recognizes the pseudo-label of utterance intent and then computes utterance pseudo-label attention. Furthermore, the utterance pseudo-label attention is embedded into

*通讯作者

the hierarchical structure of HAN and is integrated with its sentence-level attention. Sentence-level attention that incorporates utterance intent information further improves the overall performance of the model. We conducted experiments on the shared task dataset of "Customer Intent Classification Evaluation Competition for Customer Service Domain" sponsored by the Chinese Information Processing Society of China. Experiment results show that the proposed model achieved better performance than HAN on dialogue intent classification.

Keywords: Intent classification , Pseudo label attention , Hierarchical attention network ,

1 引言

近年来，人机对话由于其潜在的发展潜力和诱人的商业价值而收到越来越多的关注 (俞凯等, 2015; Chen et al., 2017)。意图分类是人机对话中的基础任务。在人机对话领域中，通常的做法是，先将用户语音通过语音识别技术转换成文本，然后再通过意图分类算法将其识别到相应类型下的具体意图。这样可以识别出用户输入到对话系统执行某个动作之间的一个映射关系，达到识别和理解用户要表达的意图的目的。在人机对话系统中，系统通过意图分类判断用户意图，再触发相应的业务类型。在面向智能客服的意图分类任务中，业务类型通常包括查询类、咨询类、办理类和投诉类等(在本文的意图分类数据上，查询类和咨询类合并成一个类型),每个业务类型下还有多种不同的用户意图。表1给出了一个例子，表示客服和用户的语音对话记录，以及对应的意图类别。

序号	对话段	
1	客服：	您好请讲
2	用户：	我问一下就是那个我我在那个中国移动这个官方网页上看到的是六十元可以充值呃十二g那个流量哦坐车电话
3	客服：	现在准确的但是前提那个得是一百三十八以上的套餐才能办理而且我这没有套餐的能办理什么嗯这个只能办理正常包月流量一号
4	用户：	包月咋办的
5	客服：	十元一百兆二十元三百兆三十元五百兆四十元七百兆五十元一个g的这种
6	用户：	哦那就不能换
7	客服：	对那种的办理不了那种前提您套餐资费也比较高的您得先改成一百三十八以上才可以办理的
8	用户：	我这卡要变成那个国通卡最低消费多钱的
...		.....
21	客服：	请问您还有其他问题需要咨询吗
22	用户：	六
23	客服：	哎好了祝您生活愉快再见好再见哎
业务类型：咨询(含查询)		
用户意图：营销活动信息		
类别合并：咨询(含查询)-营销活动信息		

Table 1: 一个意图分类例子

人机对话意图分类属于文本分类任务。已经有很多文本分类的研究致力于提高分类器性能，早期的典型代表是SVM和最大熵等 (Haffner et al., 2003; Phan et al., 2008)。随后，深度学习在自然语言处理(natural language processing, NLP)中受到关注，主流的应用包括深度信念网络(deep belief networks，DBN) (Sarikaya et al., 2011)、CNN (Xu and Sarikaya, 2013; Kim, 2014)和RNN (Xu and Sarikaya, 2014)等，尤其是RNN中最常用的LSTM (Cheng et al., 2016; Ravuri and Stolcke, 2016; Vu et al., 2016; 柯子等, 2018)。近年来，注意力机制被引入到了NLP中，实验证明其善于在文本分类任务中抽取文本的含义，例如话语意图检测 (Liu and

Lane, 2016)、话语领域分类 (Kim et al., 2018)、问答情感分类 (安明慧等, 2019)和针对语篇长文本的文档分类 (Yang et al., 2016)等。尽管基于深度学习的分类模型，尤其是层次注意力网络(hierarchical attention network, HAN) (Yang et al., 2016)，将长文本的分类性能提高到一个新的水平，但对于同时具有口语化、长文本和特征稀疏的多轮人机对话段文本，其意图分类仍然存在较大挑战。如图1的真实客服对话段例子可以看到，人类对话文本以寒暄和多轮询问式对话为主，文本通常只包含极少实际有意义的词语,导致内容具有特征稀疏性问题，增加了抽取有效特征的难度。此外，由于用户不同的口音和语义表达偏好，以及容易出现的不规则特征词和未登录词，也加剧了语音识别的错误比例，进一步给对话段的意图分类带来困难。尤其是长的对话段(如本文实验里的数据集，对话段的平均话语数达到20句左右)挑战更大。

针对上述挑战，经过分析和借鉴人类理解对话段的经验，我们发现每个单句话语所表达的意图也是一种十分重要的信息，这些信息对于整体对话段的意图理解有着举足轻重的作用。对于人机对话意图分类任务而言，标注数据仅为整个对话段的意图标签，参加到对话段意图识别的单句话语的意图标签只能靠额外构建的单句话语意图识别模型预测得到，有别于真实标注的标签，本文称之为伪标签(pseudo label)。本文提出了一种结合单句话语伪标签注意力的层次注意力网络模型PLA-HAN (HAN with utterance pseudo label attention)。PLA-HAN通过构建单句话语意图识别模型和设计话语伪标签注意力机制，识别单句话语意图伪标签，并计算话语伪标签注意力。进而将单句话语伪标签注意力嵌入到HAN的层级结构中，与HAN中的句子级别注意力相融合。此外，我们还对伪标签集进行了优选，选用了和人机对话任务关联度高的单句话语意图伪标签。融合了单句话语意图信息的句子级注意力使模型整体性能得到进一步的提升。我们在中国中文信息学会主办的“客服领域用户意图分类评测比赛”的评测语料上进行实验，实验结果证明PLA-HAN模型取得了优于HAN的意图分类性能。

本文的其余部分组织如下：在第2节，我们介绍了相关的基础工作，包括层次注意力网络HAN以及单句话语意图识别模型。第3节介绍本文提出的方法，包括我们设计的伪标签注意力机制，以及结合伪标签注意力的人机对话意图分类模型。第4节是实验和分析。第5节简要总结了本文的工作。

2 基础工作

2.1 层次注意力机制

更符合长文本层级结构的是HAN(Yang et al., 2016)所采用的层次注意力机制，它带有两个层级的注意力机制，分别是词级别和句子级别，能够更好地表示长文本中重要信息的位置。

注意力机制。近年来，在文本分类问题上，基于注意力机制的模型在效果和效率上都展现出了一定的优越性。我们考虑非层级结构的软注意力机制以及层级结构的分层注意力机制。

注意力机制最早在机器翻译中应用，现在已经成为神经网络相关中一个十分具有影响力的概念，注意力机制相当于模仿人类将大量视觉信息压缩成描述性语言的非凡能力(Xu et al., 2015)，它将编码器的输出映射为注意力权重，将权重与编码器输出进行加权，如式(1)-(3)所示。

$$u_i = score(h_i) \tag{1}$$

$$\alpha_i = \frac{exp(u_i)}{\sum_i^L (u_i)} \tag{2}$$

$$h_* = \sum_i^L \alpha_i h_i \tag{3}$$

其中，本文采用前馈神经网络为对齐函数(score)、在语音识别(Graves et al., 2013)和口语语言理解(Xu and Sarikaya, 2014)领域有着非常成功应用的双向LSTM作为编码器。通过对齐函数将编码器的输出对齐，得到注意力分数，进行归一化后得到注意力权重α，最终将编码器输出与注意力权重进行加权。

词级别。对于一个句子的词w^*，在经过embedding之后，获得词向量w^*_{emb}，编码器对词向量进行编码，获得词级别的表达h^*_{word}，通过单层MLP对h^*_{word}进行对齐计算，获

得u^*_{word}，使用softmax函数得到字级别的注意力权重α^*_{word}，在这之后，通过字级别注意力权重α^*_{word}与h^*_{word}加权求和得到句子向量s^*。如式(4)-(8)所示。

$$w^*_{emb} = embedding(w^*) \tag{4}$$

$$h^*_{word} = Encoder(w^*_{emb}) \tag{5}$$

$$u^*_{word} = tanh(W_w h^*_{word} + b_w) \tag{6}$$

$$\alpha^*_{word} = \frac{exp(u^*_{word} u_w)}{\sum_t exp(u^*_{word} u_w)} \tag{7}$$

$$s^* = \sum_t (\alpha^*_{word} word^*_{emb}) \tag{8}$$

句子级别。与词级别的做法相似，编码器对句子向量s^*进行编码获得句子向量的表达$h^*_{sentence}$，同样通过单层MLP对句子向量进行对齐计算，获得$u^*_{sentence}$，使用softmax函数得到句子级别的注意力权重$\alpha^*_{sentence}$，最终通过句子级别注意力权重$\alpha^*_{sentence}$与$h^*_{sentence}$加权求和作为文档向量v。如式(9)-(12)所示。

$$h^*_{sentence} = Encoder(s^*) \tag{9}$$

$$u^*_{sentence} = tanh(W_s h^*_{sentence} + b_s) \tag{10}$$

$$\alpha^*_{sentence} = \frac{exp(u^*_{sentence} u_s)}{\sum_t exp(u^*_{sentence} u_s)} \tag{11}$$

$$s^* = \sum_t (\alpha^*_{sentence} sentence^*_{emb}) \tag{12}$$

2.2 单句话语意图识别模型

对于单句话语的意图识别，我们采用了基于BERT的双向LSTM模型，BiLSTM结构在语音识别(Graves et al., 2013)和口语语言理解(Xu and Sarikaya, 2014)领域有着非常成功的应用。在BERT层，我们采用预先训练好的中文BERT模型对固定长度为L的话语序列$w^* = \{w_1, w_2, ..., w_L\}$进行编码，每个字$w_i$会被编码为字向量$e_i$；这样$w^*$就被编码成了$emb_u = \{e_1, e_2, ..., e_L\}$。如式(13)所示。

$$emb_u = BERT(w^*) \tag{13}$$

在BiLSTM网络层，我们将$emb_u = \{e_1, e_2, ..., e_L\}$输入到双向的长短时记忆网络，然后分别从网络中得到正向的输出$h^{fw} = \{h^{fw}_1, h^{fw}_2, ...h^{fw}_L\}$和反向的输出$h^{bw} = \{h^{bw}_1, h^{bw}_2, ...h^{bw}_L\}$。如式(14)所示。

$$h^{fw}, hbw = BiLSTM(emb_u) \tag{14}$$

我们将正向和反向的结果拼接起来得到双向LSTM的输出h。如式(15)所示。

$$h = [h^{fw}, f^{bw}] \tag{15}$$

对于输入序列中的每个元素，每个LSTM结构计算以下函数，其中h_t是t时刻的隐藏层，c_t是t时刻的记忆单元，x_t是t时刻的输入，h_{t-1}是t-1时刻的隐藏层，i_t、f_t、$\tilde{c}_t$和o_t分别

是输入门、遗忘门、记忆单元门和输出门，σ是sigmoid函数，$\odot$是哈达玛积。如式(16)-(21)所示。

$$i_t = \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}) \tag{16}$$

$$f_t = \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}) \tag{17}$$

$$\tilde{c_t} = tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \tag{18}$$

$$o_t = \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \tag{19}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c_t} \tag{20}$$

$$h_t = o_t tanh(c_t) \tag{21}$$

接着将h累加得到话语序列的句子向量s，最后接上全连接-softmax层得到各个意图类别的概率输出。如式(22)所示。

$$y = softmax(Wh + b) \tag{22}$$

3 本文的方法

在本节中，我们将介绍如何将意图伪标签模型学到的伪标签注意力权重和层级注意力权重相融合，从而构成层级结构中嵌入单句话语意图伪标签注意力的机制。

3.1 伪标签注意力

在面向长文本的任务型意图分类场景中，能够提取到长文中的关键单句话语将会对整体的分类效果带来提升。除了在模型内部采用注意力机制来实现，也可以通过外部给模型带来额外的信息以更好地提取到关键单句话语。在此，我们通过单句话语的意图伪标签来实现将外部信息注入模型结构中。长文本的每个单句话语意图伪标签的加入，可以视为构建了一个内部的"对话意图-伪意图标签"的分布，反映总体对话所表达的意图倾向。我们通过一个注意力模型学习到了每个伪标签所对应的注意力，这种注意力作为一种额外的信息，反映了其对应的单句话语对于总体对话意图的重要程度，利用该信息可以让模型更好地选择出重要的单句话语。

通过相关联的单句话语意图任务，我们训练了一个针对单句话语的意图识别模型(详见2.2节)，记为$model_{intent}$，通过此模型，给主任务中的每个子句标注上意图伪标签$intent^{pseudo}$，使用one-hot对意图伪标签进行独热编码，获得意图伪标签的文本表示$intent_{emb}^{pseudo*}$，经过编码器得到$intent_h^{pseudo*}$，在此处采用软注意力来获取注意力权重β_{pseudo}(详见2.1节的注意力机制部分)，记为伪标签意图注意力，通过伪标签意图注意力可以反映出"对话意图-伪意图标签"的分布，从而表达单句话语意图的重要程度。如式(23)-(25)所示。

$$intent_{emb}^{pseudo*} = OneHot(intent^{pseudo}) \tag{23}$$

$$intent_h^{pseudo*} = Encoder(intent_{emb}^{pesudo*}) \tag{24}$$

$$\beta_{pseudo} = Attention(intent_h^{pseudo*}) \tag{25}$$

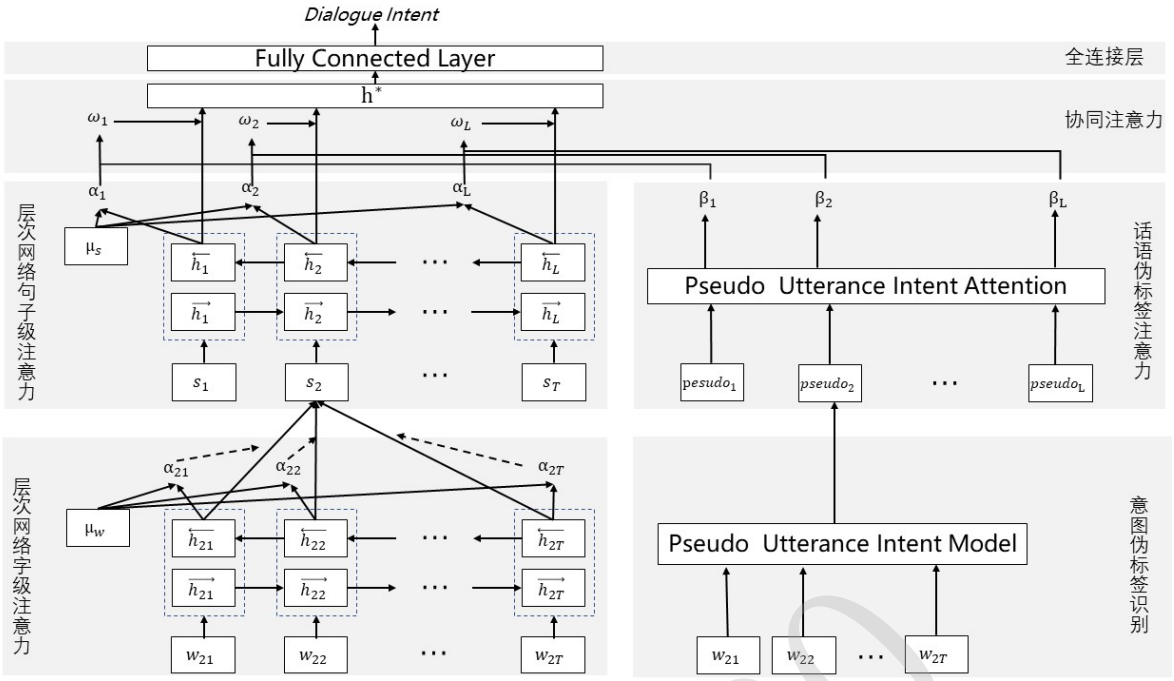


Figure 1: PLA-HAN模型的架构

3.2 结合伪标签注意力的人机对话意图分类

本文的模型是在HAN模型基础上融入伪标签注意力构成的话语伪标签注意力层次注意力网络PLA-HAN，模型采用BERT编码 (Devlin et al., 2019)，由字级层、句子级层、协同注意力层和全连接层组成，如图1所示。

BERT编码。我们采用中文BERT预训练模型对字进行编码。固定长度为L的序列输入，通过BERT编码后将会被编码为(L, 768)的向量。

PLA-HAN字级层。包括了HAN模型中的字级注意力网络(图1的左下部分)和单句话语的意图伪标签识别模型(图1的右下部分)。

(1) HAN的字级注意力：在HAN中，对于经过BERT编码的篇章结构作为输入计算得到词级别的注意力α_{word}(详见2.1节)。

(2) 意图伪标签识别：通过单句话语意图识别模型，给对话段中的每个话语打上伪意图标签，此时得到对话段的“对话意图-伪标签意图”分布。

PLA-HAN句子级层。包括了HAN模型中的句子级注意力网络(图1的左中部分)和单句话语的意图伪标签注意力机制(图1的右中部分)。

(1) HAN的句子级注意力：在HAN中，在句子向量的基础上进一步计算句子级别的注意力$\alpha_{sentence}$(详见2.1节)。

(2) 话语伪标签注意力：同时，我们采用3.1节中介绍的方法，通过额外构建的注意力模型，将“对话意图-伪标签意图”分布转化为伪标签注意力β_{pseudo}。

协同注意力层。我们采用加法将分层注意力机制的句子级别注意力和伪标签注意力相融合，记为ω_i。如式(26)所示。

$$\omega_i = \alpha_{sentence} + \beta_{pseudo} \tag{26}$$

最后我们利用注意力权重对分层注意力结构中的句子向量s^*进行加权缩放，并将其累加得到的h^*作为注意力层的输出。如式(27)所示。

$$h^* = \sum_t (\omega_i * s_i) \tag{27}$$

全连接层。 我们利用全连接层将模型的输出映射为相应意图类别数量，接着使用softmax输出各个意图类别的概率，最终采用概率最高的意图类别作为输出。公式如式(28)-(29)所示。

$$y^* = softmax(W * h^* + b) \tag{28}$$

$$predict = argmax(y^*) \tag{29}$$

3.3 伪标签的挑选

对于单句话语意图模型所带来的意图伪标签，并不是所有的伪标签都能对人机对话段的意图分类产生正面效果。为此，本文的方案中，通过全部伪标签的初步试验，计算出每个伪标签在对话段中的平均覆盖率:

$$pesudo_{cover} = \text{意图伪标签在对话段中的出现次数/对话段的对话轮次} \tag{30}$$

然后，筛选出覆盖率高的伪标签:

$$pseudo_{select} = topk(pesudo_{cover}) \tag{31}$$

在覆盖率分析的基础上，我们进一步从语义角度挑选出高关联度的伪标签嵌入到模型中，去掉了部分关联度不高的标签。在本文的实验中，用以训练单句话语意图识别模型的数据集共有48种单句话语意图类别。经过伪标签挑选，我们的最终方案中，选取了25种与人机对话意图分类任务高关联度的单句话语意图，同时将其它的单句话语意图标签标记为“其它”。不同伪标签集的实验效果见4.5小节。

4 实验

4.1 数据集

本文的人机对话实验数据来自于中国中文信息学会主办的“客服领域用户意图分类评测比赛”[1]，属于客服领域对话文本，视为多轮话的长文本。同时这项比赛还有一个子任务是单句用户话语的自然语言理解(包括意图识别和槽填充)。本文将其中的话语意图识别任务视为本文人机对话任务的辅助任务，用于产生人机对话段中每一句单句话语的意图伪标签。

人机对话数据集人机对话数据集为2万条真实客服对话段标注数据，此数据集中共有35种人机对话意图类别，表2给出了业务类型与用户意图的种类。我们按照8：2的比例分别划分训练集和测试集。训练时，再从训练集中划分出20%作为验证集。

单句话语意图识别数据集 单句话语的数据集为2万条真实的单句话语数据，此数据集中共有48种单句意图类别，如表3所示。经过伪标签优选，选取了与对话段意图分类任务相关度高的25种单句意图类别，其它类别的伪标签标记为“其它”。按照8：2切分训练集和验证集，不设置测试集。

业务类型	对话意图
咨询(含查询)	业务订购信息查询、业务规定、业务订购信息查询、业务资费、产品/业务功能、使用方式、办理方式、号码状态、宽带覆盖范围、工单处理结果、工单处理结果、服务渠道信息、用户资料、电商货品信息、营销活动信息、账户信息
办理	下载/设置、停复机、取消、变更、开通、打印/邮寄、移机/装机/拆机、缴费、补换卡、重置/修改/补发、销户/重开
投诉(含抱怨)	不知情定制问题、业务使用问题、业务办理问题、业务规定不满、信息安全问题、服务问题、网络问题、营销问题、费用问题

Table 2: 业务类型种类与35种对话意图类别

[1]http://www.cips-cl.org/static/CCL2018/call-evaluation.html

单句话语意图标签
查询、查询套餐余量、查询充值缴费记录、查询流量、查询本机号码、查询本机业务、查询余额、查询短信、查询积分、查询语音、查询月初扣费、查询账单、查询宽带、查询手机、咨询宽带、具实帮助、具实返回、具实转人工、具实退出、具实重听、具实业务列表、具实转ivr、办理套餐、办理手机充值、取消流量、预约宽带、修改宽带、开通流量、重置、重置手机、重置宽带、GPRS、拒识、修改、确认、手机、修改手机、短信、宽带、流量、empty、集外说法批评,抱怨,脏话、集外说法机器人、集外说法集外业务、咨询、集外说法结束、集外说法短拒识、集外说法友好问候、集外说法感谢

Table 3: 48种单句话语意图类别

4.2 实验设置

人机对话意图分类实验 batch_size设置为16，epoch设置为20，Dropout设置为0.1，采用Adam优化器，学习率为1e-5，非层级模型设置最大句子长度为512，层级模型设置最大句子数量为25，最大句长为25。

话语伪标签预测实验 batch_size设置为32，epoch设置为100，Dropout设置为0.1，采用Adam优化器，学习率为1e-3，非层级模型设置最大句子长度为30。

4.3 对比方法

本文提出的PLA-HAN模型将与以下代表性的基线方案进行比较，为了公平比较，全部模型都采用了BERT编码：

- BERT FineTune：该方法将BERT fine-tuning (Devlin et al., 2019)应用到分类任务，在BERT分类层增加了一个新的输出层。
- BERT BiLSTM：该方法是文本分类的经典基线 (Vu et al., 2016)，适合于序列问题。
- BERT SoftAtt：该方法采用了Liu等(Liu and Lane, 2016)在ATIS数据集的话语意图识别的BILSTM模型中采用的软注意力。
- BERT HAN：Yang等(Yang et al., 2016)提出的更加适合篇章结构文本的层次注意力模型结构。

4.4 整体性能

我们提出的PLA-HAN模型与几种对比方法进行比较，包括BERT FineTune、BiLSTM、带软注意力的BiLSTM以及层次注意力模型HAN。实验结果如表4所示。

模型类别	模型	意图分类正确率(%)			
		总体	咨询(含查询)	办理	投诉
长文本结构	BERT FineTune	53.11	49.50	66.72	48.14
	BERT BiLSTM	55.75	52.39	69.83	49.35
	BERT SoftAtt	55.87	51.98	71.03	49.91
层次结构	BERT HAN	56.31	52.55	71.08	50.40
	BERT PLA-HAN	**56.94**	**53.33**	**71.26**	**51.35**

Table 4: 不同模型性能对比

从表4的结果可以看到：

(1) 在以整篇长文本直接作为输入的模型中，通过加入编码器BiLSTM和注意力机制的应用，可以在BERT的基础上进一步提升对话段意图分类的性能。

(2) 相比于整篇长文本结构的输入，层次结构模型取得了更好的分类性能。这一方面得益于不受BERT输入长度的限制，另一方面也由于分层次的注意力能更好地建模字到句子再到对话段的语义结构。

(3) 融合了单句话语伪标签注意力的PLA-HAN取得了最好的性能，优于HAN模型。

4.5 进一步分析

通过比较以上结果可以看出PLA-HAN模型取得了良好的性能，我们也想进一步探究模型能有所提升的原因。我们首先分析了不同的伪标签集对模型性能的影响。然后，我们给出了一个不同长度对话段情况下的PLA-HAN模型与基础的HAN模型的性能对比的定量分析。

不同伪标签集对模型性能的影响。为了研究不同伪标签集在PLA-HAN模型中效果，我们对采用三种不同的伪标签集的BERT PLA-HAN的实验结果进行观察。PLA-HAN(All)代表不做选择采用了全部的48种伪标签，PLA-HAN(Fit)代表只选用了和人机对话意图相关的32种伪标签，PLA-HAN(Select)代表进一步优选的25种伪标签。结果如表5所示。

模型	意图分类正确率(%)			
	总体	咨询(含查询)	办理	投诉
BERT PLA-HAN(All)	56.06	52.48	70.41	49.95
BERT PLA-HAN(Fit)	56.59	53.31	70.94	50.72
BERT PLA-HAN(Select)	**56.94**	**53.33**	**71.26**	**51.33**

Table 5: 不同伪标签集的PLA-HAN模型性能对比

结果表明，我们的伪标签集选择策略是必要的，不同的伪标签集对模型整体性能存在明显的影响。详细的分析如下：

- PLA-HAN(All)：在此方案中，我们采用了所有48种伪标签。其中部分伪标签实际上和人机对话意图分类任务中的35种意图的相关性并不强。从实验结果可以看到，分类性能都不够好，甚至都略微低于HAN模型。

- PLA-HAN(Fit)：在此方案中，我们选取与人机对话任务的意图有较高覆盖率的伪标签嵌入到模型中，将基本不相关的标签标记为“其它”。实验中通过验证集选取了32个伪标签。从实验结果可以看到，分类性能相比于PLA-HAN(All)有了明显提高，相同时也优于HAN模型。

- PLA-HAN(Select)：在此方案中，我们在PLA-HAN(Fit)的基础上，进一步从语义角度挑选出高关联度的伪标签嵌入到模型中，去掉了部分关联度不高的标签，一共选取了25种伪标签。从实验结果看，PLA-HAN(Select)取得了最好的性能。

不同长度对话段的模型性能对比。我们进一步对比不同对话段长度情况下的PLA-HAN模型与基础的HAN模型的性能。我们按照对话段的长度分为长(600字以上)、中(301-600字)和短(300字以下)三类进行观察，结果如图2所示。

从图2可以看到：

(1) 长的对话段的意图分类存在较大的挑战，正确率明显低于短的对话段。

(2) 我们的PLA-HAN，在伪标签注意力的帮助下，在不同长度的对话段性能均优于HAN。尤其是长的对话段(超过600字)，意图分类正确率提升较为显著，达到1.56%。

5 结束语

针对现有文本分类方法在人机对话意图分类上存在的挑战，本文提出了一种结合话语伪标签注意力的层次注意力网络模型PLA-HAN。PLA-HAN通过优选伪标签集，设计和计算单句话语意图伪标签注意力，并将其嵌入到HAN的层级结构中，与HAN中的句子级别注意力相融合，提升了人机对话意图分类性能。我们在中国中文信息学会主办的“客服领域用户意图分类评测比赛”的评测语料上进行实验，实验结果证明PLA-HAN模型取得了优于HAN等研究进展文本分类方法的意图分类正确率。

致谢

本文受到国家自然科学基金(项目编号:71472068)的资助。

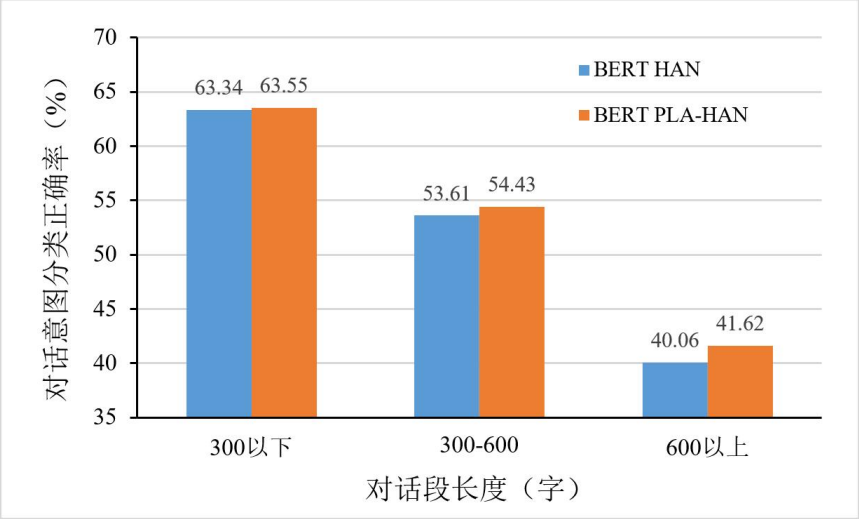


Figure 2: 不同长度对话段的模型性能对比

参考文献

安明慧, 沈忱林, 李寿山, 等. 2019. 基于联合学习的问答情感分类方法. 中文信息学报, 33(10):119-126.

柯子, 黄沛杰, 曾真. 2018. 基于优化“未定义”类话语检测的话语领域分类. 中文信息学报, 32(4):105-113.

俞凯, 陈露, 陈博, 等. 2015. 任务型人机对话系统中的认知技术——概念、进展及其未来. 计算机学报, 38 (12):2333-2348.

Chen, Hongshen and Liu, Xiaorui and Yin, Dawei and Tang, Jiliang. 2017. *A survey on dialogue systems: Recent advances and new frontiers*, volume 19. ACM New York, NY, USA.

J. P. Cheng, L. Dong, and M. Lapata. 2016. Long short-term memory-networks for machine reading. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pp. 551–561.

J. Devlin, M. Chang, K. Lee, et al. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019)*, pp. 4171–4186.

A. Graves, N.Jaitly and A. Mohamed. 2013. Hybrid speech recognition with deep bidirectional LSTM. *Proceedings of the 2013 IEEE workshop on automatic speech recognition and understanding (ASRU 2013)*, 273–278.

P. Haffner, G. Tur, and J. H. Wright. 2003. Optimizing SVMs for complex call classification. *Proceedings of the 28th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003)*, pp. 632-635.

B. Liu and T. Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *Proceedings of the 17th Annual Conference of the Inter-national Speech Communication Association (INTERSPEECH 2016)*, pp. 685-689.

Y. Kim. 2014. Convolutional Neural Networks for Sentence Classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pp. 1746–1751.

Y. Kim, D. Kim, A. Kumar. 2018. Efficient large-scale neural domain classification with personalized attention. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, pp. 2214–2224.

X .H. Phan, L. M. Nguyen, S. Horiguchi. 2008. Learning to classify short and sparse text & web with hidden topics from largescale data collections. *Proceedings of the 17th International Conference on World Wide Web (WWW 2008)*, pp. 91-100.

S. Ravuri and A. Stolcke. 2016. A comparative study of recurrent neural network models for lexical domain classification. *Proceedings of the 41th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2016)*, pp. 6075-6079.

R. Sarikaya, G. E. Hinton, and B. Ramabhadran. 2011. Deep belief nets for natural language call-routing. *Proceedings of the 36th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2011)*, pp. 5680–5683.

N. T. Vu, P. Gupta, H. Adel, et al. 2016. Bi-directional recurrent neural network with ranking loss for spoken language understanding. *Proceedings of the 41th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2016)*, pp. 6060–6064.

P. Y. Xu and R. Sarikaya. 2013. Convolutional neural network based triangular CRF for joint intent detection and slot filling. *Proceedings of the 2013 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU 2013)*, pp. 78–83.

P. Y. Xu and R. Sarikaya. 2014. Contextual domain classification in spoken language understanding systems using recurrent neural network. *Proceedings of the 39th International Conference on Acoustics, Speech and Signal Processing (ICASSP 2014)*, pp. 136–140.

K. Xu, J. Ba and R. Kiros,et al. 2015. Show, attend and tell: Neural image caption generation with visual attention. *Proceedings of the 31st International Conference on Machine Learning (ICML 2015)*, pp. 2048–2057.

Z. C. Yang, D. Y. Yang, C. Dyer, et al. 2016. Hierarchical attention networks for document classification. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2016)*, pp. 1480-1489.

基于BERTCA的新闻实体与正文语义相关度计算模型

向军毅[1,2], 胡慧君[1,2],毛瑞彬[3] ,刘茂福[1,2]✉
1. 武汉科技大学计算机科学与技术学院, 武汉, 430065
2. 智能信息处理与实时工业系统湖北省重点实验室, 武汉, 430065
3. 武汉大学信息资源研究中心, 武汉, 430072
liumaofu@wust.edu.cn

摘要

目前的搜索引擎仍然存在“重形式，轻语义”的问题，无法做到对搜索关键词和文本的深层次语义理解，因此语义检索成为当代搜索引擎中亟需解决的问题。为了提高搜索引擎的语义理解能力，提出一种语义相关度的计算方法。首先标注金融类新闻标题实体与新闻正文语义相关度语料1万条，然后建立新闻实体与正文语义相关度计算的BERTCA(Bidirectional Encoder Representation from Transformers Co-Attention)模型，通过使用BERT预训练模型，综合考虑细粒度的实体和粗粒度的正文的语义信息，然后经过协同注意力，实现实体与正文的语义匹配，不仅能计算出金融新闻实体与新闻正文之间的相关度，还能根据相关度阈值来判定相关度类别，实验表明该模型在1万条标注语料上准确率超过95%，优于目前主流模型，最后通过具体搜索示例展现该模型的优秀性能。

关键词： 语义相关度计算 ； BERT模型 ； 协同注意力机制 ； 语言模型

Semantic Relevance Computing Model of News Entity and Text based on BERTCA

Junyi Xiang[1,2], Huijun Hu[1,2],Ruibin Mao[3] ,Maofu Liu[1,2]✉
1. School of Computer Science and Technology,
Wuhan University of Science and Technology, Wuhan, 430065
2. Hubei Province Key Laboratory of Intelligent Information Processing and
Real-time Industrial System, Wuhan, 430065
3. Center for Studies of Information Resources, Wuhan University, Wuhan, 430072
liumaofu@wust.edu.cn

Abstract

At present, the search engines cannot understand the semantic deep meaning of keywords and texts, and still face the problem of more attentions to form than semantics. In order to improve the ability of semantic retrieval in contemporary search engines, this paper proposes the semantic relevancy computing method. The corpus with the entity and text semantic relatedness in 10,000 financial news headlines has been constructed firstly by manual annotation, and then the BERTCA (Bidirectional Encoder Representation from Transformers co-attention semantic relevancy computing) model has been established using this corpus. This model has taken both fine-grained entity semantic information and coarse-grained text by BERT pre-trained language mode into consideration. Through the co-attention mechanism, this model can obtain the semantic matching between the entity and text, and it can not only calculate the degree of correlation between entity and text, but also determine the degree of correlation according to the semantic relevancy. The experimental results show that the accuracy of the proposed model has achieved more than 95% on the constructed corpus and is better than the state-of-the-art models.

Keywords: semantic relevance computing , BERT , co-attention mechanism , language model

1 引言

当代互联网飞速发展，人们可以方便地通过搜索引擎获取符合需求的信息。其中，金融新闻的检索占比尤为重要。对关心市场变化的投资者而言，金融类新闻、公告和资讯等数据有着极其重要的参考价值。然而金融数据来源广泛、种类繁杂，直接定位关键信息非常困难。因此，通过实体对海量的金融类数据进行精确有效且快速地定位 (MacAvaney et al., 2019; Ai et al., 2018; Wang et al., 2018; Xiong et al., 2018)，仍然面临着重大挑战。虽然用户一直迫切需要精准检索，但庞杂的无关结果仍是困扰用户的一个顽疾。

目前，大型搜索引擎公司为了提高搜索引擎的效率，主要的解决方案是对每个入库的网页进行分析，得到每个网页的关键词或者标题等部分重要信息，然后再将用户查找的关键词切分，与每个网页的关键词进行匹配，再过滤垃圾结果，最后根据网站的整体评价、网页质量、内容质量、资源质量等指标进行重排。这样就会因为牺牲了全文语义信息，导致搜索引擎的准确度下降，例1是使用搜索引擎搜索“万科”金融新闻的实例：

例1：
标题： 宝能系增持华润旗下东阿阿胶，万科股权事件重演
正文： 万科股权之争稍稍有些停息，“宝能系”又盯上了华润旗下的另一家上市公司东阿阿胶。据东阿阿胶2016年半年财报,宝能旗下的前海人寿二季度再度增持东阿阿胶,目前,持股比例已增至4.17%,逼近举牌线,成为了该公司的第三大股东。同时,华润也在二季度增持了东阿阿胶4.6%的股份,借此巩固其控股地位。分析人士指出,宝能系意图不明,华润显然感到了压力,以防“宝能系”再度搅局,万科事件重演...

在检索阶段，将实体“万科”与新闻标题进行匹配，得到一系列标题中包含“万科”的新闻。但是通过观察返回的结果不一定与该实体相关，新闻正文的核心内容和部分实体语义相关度较小。该新闻标题实体包含“宝能”、“华润”、“东阿阿胶”与“万科”，然而新闻正文主要报道了新闻标题的前半部分，即“宝能系增持华润旗下东阿阿胶”，新闻标题的后半部分“万科股权事件重演”基本上没谈及，新闻正文的主要内容与“万科”本身关系不大，而搜索引擎无法判断这一点，在检索“万科”的时候也会将该条新闻返回，。从金融债券大数据的统计情况来看，这样的新闻数量大约占搜索引擎返回结果的1/5。从用户的角度出发，这样的结果无法准确找到体现所关注公司价值和风险的信息，最终降低用户体验。因此，解决实体与正文的“语义鸿沟”仍然是一项艰巨的任务和挑战。

为了让搜索结果中的高匹配结果排名靠前，采用排序学习(Learning to Rank,LTR)方法对搜索结果进行重排是非常好的选择。目前，使用机器学习方法来进行排序学习取得了非常好的效果 (Burges, 2010)，排序模型依靠多种特征，比如关键词匹配特征、BM25特征、词频、逆文档频率以及PageRank特征等等，对人工相关性标注数据进行学习，从而实现结果排序。使用深度学习方法进行排序也进行了探索 (Köppel et al., 2019)。然而，目前的开源的排序学习数据集都是通过对特征项进行拟合，而并非使用自然语言，例如：在LETOR4.0 (Qin et al., 2010)数据集中，每一条查询文档对包括46个特征。丰富特征确实可以得到相对优秀的结果，但是仍然没法避免关键语义信息丢失的问题。因此，本文通过众包的形式，构建了自然语言形式的新闻实体与正文的语义相关度数据集1万条。之所以使用新闻实体，这是由于金融领域搜索引擎的日志中占比80%以上都是对某个公司进行检索，并且搜索引擎的检索对象也是标题。通过该数据集，模型可以学习到实体与正文的语义相关度，然后通过计算实体与正文的相关度，对搜索结果进行排序，这样既缩短了检索时间，又能达到语义检索的效果。

语义相关度的应用非常广泛，对于该方面的研究也非常火热，目前，语义相关度计算的方法主要分为机器学习方法和深度学习方法两类。机器学习方法主要有TF-IDF、BM25、simHash、VSM等，在粗粒度的文本上达到了可观的效果，但是在对文本的深层次语义理解上存在缺陷。为此，研究人员运用深度学习方法来进一步提高语义理解能力，

主要有包括以DSSM为代表的深层深度学习网络，这些方法都取得了很好的效果，但是仍然存在长距离语义缺失、语义偏移等问题，并且将语义相关度与排序学习综合起来的模型较少。

为了解决上述问题，本文提出基于BERTCA的语义相关度的计算模型BERTCA，通过模型计算标题实体与正文的语义相关度，然后通过语义相关度对搜索结果进行重排。该方法将语义相关度计算方法与排序学习方法结合，充分利用了文本的语义信息，实现了搜索引擎的语义检索功能。标题与正文的语义相关度可以预先计算存入数据库中，因此搜索引擎的速度并不会降低。BERT的多层多头自注意力的堆叠结构使得模型能够在每个层次注意不同的信息，语义偏移问题得到了显著地缓解，而语义交互层可以让实体与正文的语义信息得到充分交互，即使是长距离的文本，也会得到足够的重视，将两者结合起来，就会得到了更好的语义相关度的计算结果。

2 相关工作

近年来，随着深度学习的突破，神经网络模型不仅在图像处理、语音识别领域展现出了较好的性能，也在自然语言处理领域展现出了很大的优势。针对语义相关度计算方法，国内外学者提出了很多研究方法和模型，大体可以分为以下3类 (庞亮et al., 2017)：

(1) 单语义文档表达的计算模型：简单地使用全连接层、卷积神经网络或循环神经网络将文本表达成一个稠密向量，然后直接计算两个向量间的相关度。 Huang et al. (2013)探索了一种潜在语义模型DSSM，该模型是一个具有深层结构的潜在语义模型，它会将查询和文档投射到一个常见的低维空间中，这样子很容易将给定的文档和查询的相关性计算为它们之间的距离； Kim et al. (2019)提出密集连接协同注意力循环神经网络模型(DRCN)，将循环神经网络中的隐含层与协同注意力层进行拼接，这样原始信息就会在最底层到最顶层之间一直保留，而在循环神经网络每一个块中，使用协同注意力的方式得到两个句子之间的交互信息，因为参数量的迅速增加，影响模型训练，因此用自编码器进行压缩表示。

(2) 多语义文档表达的计算模型：此方法认为单一粒度的语义信息无法精细地描绘出文本所有内容，因此建立了多语义表示，让两段文本进行交互，挖掘文本交互后的模式特征，综合得到文本间的相关度。 Wan et al. (2016)提出了MV-LSTM来解决模型无法捕获上下文相关的局部信息的问题，利用长短时记忆神经网络(Long Short-Term Memory，LSTM)获取上下文语义向量，通过k-Max池化和多层感知器变换，最后聚合这些不同位置的句子表示之间的交互，最终产生语义相关度得分；(3) 直接计算模型：为了进行更深层次的语义信息交互，应该考虑不同层次的交互信息，更精细的处理句子中的联系，再用深度神经网络挖掘交互后的模式特征，综合计算文本之间的相关度。 Lu and Li (2013)提出了DeepMatch网络，通过使用文档主题生成模型(Latent Dirichlet Allocation，LDA)模型获取两个文本的共现情况，从而得出两个文本的匹配分数； Xu et al. (2019)提出了一种文本匹配的多层次匹配网络(MMN)，该网络利用多个词的表示来获得多个词的水平匹配结果，从而进行最终的文本水平评分。

深度学习模型使词向量的具有线性的语义信息，但是仍然没有解决语义偏移问题，研究者在这个方面进行了大量的探索，其中预训练语言模型表现的非常好。 Devlin et al. (2018)提出了一种新的语言表示模型BERT，它是由Transformers (Vaswani et al., 2017)的双向编码器表示，与其他的语言表示模型不同，BERT会利用未标记文本来预训练深层双向表示，这种双向表示不仅会对上文进行编码，还会对下文进行编码。另外还有许多BERT模型的变体：ALBERT (Lan et al., 2019)、XLNet (Yang et al., 2019) RoBERTa (Liu et al., 2019)。其中XLNet是一种泛化的自回归预训练模型，通过最大化所有可能的因式分解顺序的对数似然，学习双向语境信息，用自回归本身的特点克服BERT的缺点，并融合了最优自回归模型Transformer-XL (Dai et al., 2019)的思路，在各项任务上超过BERT创下的记录。

排序学习 (Liu, 2011; Yin et al., 2016; Wang et al., 2017)是一个监督学习，通过对每一个查询文档对进行特征抽取，训练排序模型，使得输出与标签相符。常见的排序学习方法一般分为三类：单文档型(Pointwise)，文档对型(pairwise)，文档列表型(Listwise)。

单文档方法只对单个文档进行处理，将文档转换为特征向量，根据训练数据得到的模型对其进行打分，再将所有文档按照得分结果进行排序。主要包括以下算法：Pranking, OC SVM，McRank等。

基金项目：深圳证券信息有限公司联合研究计划(No.2018002)；全军共用信息系统装备预先研究项目（31502030502）；

文档对方法将相关性得分转换为文档对关系，根据标注信息，A的得分为2，B的得分为4，C的得分为1，可以得到B>A,B>C,A>C的比较关系，这样就把排序问题转化成了二分类问题，模型通过对任意两个文档之间的关系进行分类，得到全集的排序关系。主要包括以下算法：LambdaMART (Wu et al., 2010)、RankNet、Ranking SVM等。

文档列表方法的输入为一个文档序列，通过构造合适的度量函数来优化排序，得到排序模型。主要包括以下算法：AdaRank、SVM MAP21、Soft Rank。

3 ETSR语料构建

针对ETSR语料集的构建，爬取多个金融网站的近10年新闻，制定了实体与正文语义相关度标注规范，标注了20,000条实体与正文语义相关度语料，语料集构建整体流程如图1所示。

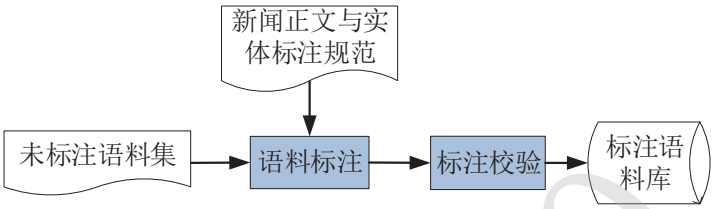


Figure 1: 语料标注流程图

3.1 语料获取

本文使用开源的Scrapy爬虫框架，对21世纪经济报道[1]、财新网[2]、每经网[3]、生意社[4]、人民网[5]、新浪上市公司[6]和腾讯上市公司[7]共7个网站爬取了45,000多条金融类新闻。由于新闻门户网站存在大量抄袭或者搬运的状况，剔除重复新闻后余下的41,000条新闻。

为了后面的语料标注的方便，首先对新闻标题进行了命名实体识别，剔除了不存在公司名、人名、组织名的新闻，保留了20,000条新闻作为未标注语料集。

3.2 语料标注

对于语料标注工作，采用众包的方式进行。同一个新闻文本，至少5位以上标注者同时标注，采用少数服从多数原则，由人工校验后，得到最终的标注语料库

人工标注内容包括实体标注和相关度标注两项。为了制订实体标注规范和相关度标注规范，共经过了8轮的交叉验证，共5,000条语料，剩余的15,000篇新闻将通过众包形式进行标注。

实体标注规范是对新闻标题中的实体进行标注，标题中可能包含多个实体或者嵌套实体，还有一些不确定的实体，其中标题实体为命名实体工具识别出来的实体，关键词实体为那些可能是实体的关键词，例如：“微信”不是一个公司名，只是腾讯公司推出的一个为智能终端提供即时通讯服务的免费应用程序。

(1) 全模式：兼顾最长原则和最短原则，标出所有可能的实体。

例1：
标题：“恒大健康轮值董事长张三”
标注：依次根据公司、职称、人名标注“恒大”、“恒大健康”、“恒大健康轮值董事长”、“张三”等四个实体

(2) 标题实体优先于关键词实体的原则：当且仅当标题中的实体无法覆盖正文内容或者不足以作为新闻正文关键词时，加标关键词实体。

[1]21世纪经济报道：http://www.21jingji.com/channel/finance
[2]财新网：http://finance.caixin.com/
[3]每经网：http://finance.nbd.com.cn/
[4]生意社：http://www.100ppi.com/kx/
[5]人民网：http://finance.people.com.cn/
[6]新浪上市公司：http://vip.stock.finance.sina.com.cn/
[7]腾讯上市公司：https://finance.qq.com/stock/

例2：
标题：“滴滴整改重组，顺风车何去何从”；
标注：优先标注“滴滴”，但是该实体无法覆盖正文内容，随后再加标“滴滴顺风车”；

语义相关度标注则是标注出标题实体与新闻正文是强相关、弱相关、不相关。语义相似度标注的原则是依据正文与实体的相关程度来界定。(1)对于报道当前标注新闻的媒体机构，如果正文没有特意介绍，一般为不相关。(2)新闻正文讲述发生在子公司身上的事情时，母公司(短名字)一般为弱相关，例如“网易云音乐”和“网易”。(3)有出现在正文的标题实体一般为不相关。

例3：
标题：TCL进军互联网电视，意欲赶超小米、乐视。
标注：新闻正文中几乎没有提及小米、乐视，二者皆为不相关。

由于人工标注对命名实体识别的公司人名实体有部分调整，因此语料标注完成后，还需要剔除标题中既没有关键词实体也没有标题实体的新闻，最终挑选了10,000条高质量的标注语料进行实体与新闻正文的语义相关度计算。

3.3 语料分析

所有语料标注结果存为JSON格式，并以UTF-8格式编码，标题最长为57个字符，其中0~35区间占比为94.8%，正文最长有10000个字符，其中0~2000个字符占比为87.2%，可见正文文本都是偏长的。但是经过阅读接近100条数据，发现正文的前200个字符能涵盖新闻正文大部分内容，因此在后续的处理中，可以合理的利用这个特点。

id: 5c39b52be65e992d1c4d74f3
title: 鲁亿通30亿收购比特币芯片商
title_entity: {”鲁亿通”: ”强相关” }
content: 比特币价格今年再现疯狂，22天涨幅超过60%;更加土豪的是,上市公司鲁亿通拟以30亿元收购一家主营业务为比特币“矿机”制造芯片的公司。重组...

Figure 2: 标注语料示例

然后，我们对10,000条语料中不同相关度的实体进行统计，统计结果如表1所示。

Table 1: 实体相关度统计结果

强相关	弱相关	不相关	总计
12,589	1,801	1,142	15,532

从表1统计结果可以看出，强相关数据偏多，弱相关和不相关数据相对偏少。这也证实了引言中的大数据统计结论，弱相关与不相关的数据占比大约1/5，从标注数据统计分析的角度验证了前述的不相关或者弱相关新闻大约占搜索引擎返回结果的1/5的假设。

4 语义相关度计算模型BERTCA

语义相关度计算模型BERTCA的整体结构如图3所示，包含动态语义编码层、上下文编码层、语义交互层和解码层四个部分，上下文编码层是LSTM (Xingjian et al., 2015)，解码层是Fusion LSTM (Liu et al., 2016)。模型的输入是实体和新闻正文两段文本，通过BERT模型得到文本的动态语义编码后，然后通过LSTM对文本进行上下文编码，然后通过协同注意力的语义交互，最后通过解码层得到两者之间的语义相似度得分。

定义输入实体为$\boldsymbol{X} = \{x_1, x_2, \ldots, x_N\}$ 和新闻正文$\boldsymbol{Y} = \{y_1, y_2, \ldots, y_M\}$ ，其中x_i和y_j分别是实体$\boldsymbol{X}$和新闻正文$\boldsymbol{Y}$中的第i和第j个字，N和M为实体X和新闻正文Y的长度。

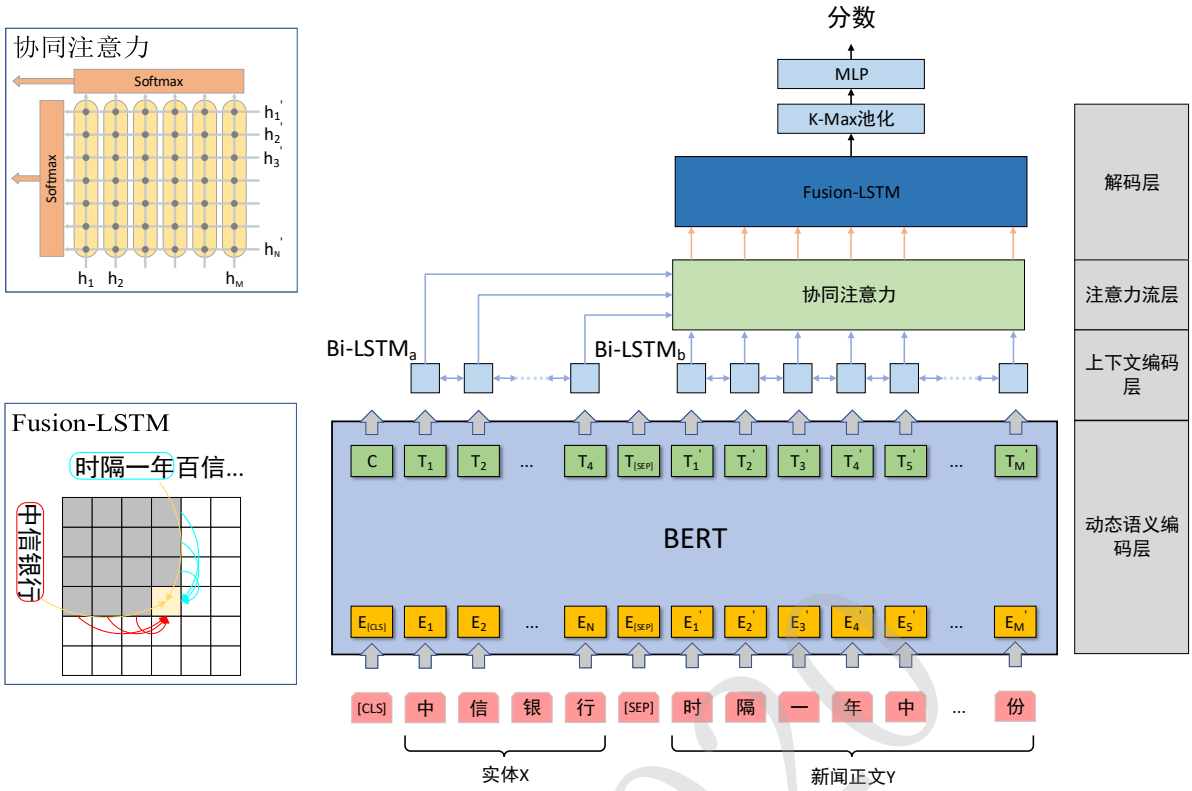


Figure 3: BERTCA语义相关度计算模型

4.1 动态语义编码层

本文通过使用预训练语言模型 (Cui et al., 2019)来构建文本动态语义编码。由于谷歌官方发布的预训练语言模型中，中文是以字为粒度进行切分，没有考虑中文分词的问题，因此该模型使用了全词遮罩(Whole Word Masking)技术来进行训练。

首先，将每个词转换成BERT模型输入的格式，包括索引、遮罩、位置和文本类型编码，这些编码会在BERT内部的嵌入层转换为向量形式，其中索引编码会转换成字向量$E_i = Emb(x_i)$ 和$E'_j = Emb(y_j)$。

其次，这些向量通过多层的编码结构，如公式(1)

$$Layer_{output} = Layer(x + \text{SubLayer}(x)) \tag{1}$$

其中x可以代表上述任何一种编码的向量形式。这样，让模型就拥有了多层结构相同但权重不同的自注意力，每一个注意力头都能关注到不同的特征，模型整体就会关注到更多的特征，如公式(2)(3):

$$MultiHead(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = Con(h_1, \ldots, h_h)\boldsymbol{W}^o \tag{2}$$

$$head_i = Attention\left(\boldsymbol{Q}\boldsymbol{W_i}^Q, \boldsymbol{K}\boldsymbol{W_i}^K, \boldsymbol{V}\boldsymbol{W_i}^V\right) \tag{3}$$

为了解决深层次神经网络中出现的退化问题，每一个Encoder都加入了残差网络和层归一化，如公式(4)(5)。

$$LN(x_i) = \alpha \times \frac{x_i - \mu_L}{\sqrt{\sigma_L^2 + \varepsilon}} + \beta \tag{4}$$

$$FFN = \max(0, x\boldsymbol{W_1} + b_1)\boldsymbol{W_2} + b_2 \tag{5}$$

代码中的注意力机制采用了缩放点积，$\boldsymbol{Q}$表示查询，$\boldsymbol{K}$为键字，$\boldsymbol{V}$为键值，都是输入的字向量。其核心思想是计算一句话中每个字与其他字之间的关联程度，并利用这种关联程度来调整字在句子中的权重，这个字向量不仅蕴含了自己本身的意思，还蕴含了与它相关联的字的关系，因此它能根据上下文对字的表征进行调整，如公式(6)。

$$Attention\left(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}\right)=softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{T}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{6}$$

整个动态语义编码层的输出是每个字的语义向量。其中[CLS]表示的是整个句子的语义向量，T_i表示实体$\boldsymbol{X}$第i个字的动态语义表示，T'_j表示的是新闻正文$\boldsymbol{Y}$中第j个字的动态语义表示。

最后，将E_i与T_i、E'_j和T'_j分别进行拼接得到实体向量$d_i=[E_i,T_i]\,(1\le i\le N)$、新闻正文向量$d'_j=\left[E'_j,T'_j\right]\,(1\le j\le M)$。

4.2 上下文编码层

上下文编码层采用了双向长短时记忆网络(Bi-LSTM)来对输入文本序列进行编码，因为实体与新闻正文是不同的文本，因为两个网络参数并不共享，t位置的隐含状态输出为：

$$h_t=\left[\overrightarrow{LSTM_a}(d_t,h_{t-1})^T,\overleftarrow{LSTM_a}\right](d_t,h_{t-1})^T] \tag{7}$$

$$h'_t=\left[\overrightarrow{LSTM_b}(d'_t,h'_{t-1})^T,\overleftarrow{LSTM_b}\right](d'_t,h'_{t-1})^T] \tag{8}$$

其中，$(\cdot)^T$表示转置操作，$\overrightarrow{LSTM}(\mathrm{d_t},\mathrm{h_{t-1}})^{\mathrm{T}}$和$\overleftarrow{LSTM}](d_t,h_{t-1})^T$分别表示长短时记忆网络在$t$时刻的正向输出和反向输出，然后将网络的正向输出与反向输出拼接起来，$\mathrm{h_i}\,(1\le i\le N)$为实体语义向量，$\mathrm{h'_i}\,(1\le i\le M)$为新闻正文的语义向量。

4.3 语义交互层

语义交互层是实现实体与正文的相互理解的重要机制，本文主要采用Co-attention，它是注意力机制的一种变体，不仅要给阅读的新闻正文生成一个注意力权重，还要给实体也生成一个注意力权重，然后利用两者的注意力权重，得到经过修正后的语义向量。

首先，计算两段文本的相关性矩阵L，然后根据新闻正文的每一个字计算实体中每一个字的注意力分数A，同理，可以根据实体的每一个字计算新闻正文中每一个字的注意力分数A'：

$$\boldsymbol{L}=\left(\boldsymbol{H'}\right)^{T}\boldsymbol{H}\left(\boldsymbol{L}\in R^{MN}\right) \tag{9}$$

$$\boldsymbol{A}=\mathrm{softmax}\left(\boldsymbol{L}\right) \tag{10}$$

$$\boldsymbol{A'}=\mathrm{softmax}\left(\boldsymbol{L}^{T}\right) \tag{11}$$

然后，通过$\boldsymbol{A'}$来对实体的语义向量进行加权，得到经过注意力修正后的实体向量$\boldsymbol{F}$，再利用实体语义向量$\boldsymbol{H}$和修正后的实体语义向量$\boldsymbol{F}$，经过新闻正文注意力分数$\boldsymbol{A}$得到修正后的实体语义向量$\boldsymbol{G}$，同理，可以得到修正后的新闻正文语义向量$\boldsymbol{G'}$：

$$\boldsymbol{F}=\boldsymbol{H}\,\boldsymbol{A'}\left(\boldsymbol{F}\in R^{l*M}\right) \tag{12}$$

$$\boldsymbol{G}=\left[\boldsymbol{H'},\boldsymbol{F}\right]\,\boldsymbol{A}\left(\boldsymbol{G}\in R^{2l*N}\right) \tag{13}$$

$$\boldsymbol{F'}=\boldsymbol{H'}\,\boldsymbol{A}\left(\boldsymbol{F}\in R^{l*N}\right) \tag{14}$$

$$\boldsymbol{G'}=\left[\boldsymbol{H},\boldsymbol{F'}\right]\,\boldsymbol{A'}\left(\boldsymbol{G}\in R^{2l*M}\right) \tag{15}$$

其中，$[\cdot,\cdot]$表示两个矩阵在第1维进行拼接。$\boldsymbol{G}$和$\boldsymbol{G}'$，作为下一层的输入，分别是实体与新闻正文的协同语义表示，是经过相互理解后的文本语义表示。

4.4 解码层

在解码层，使用了由 Liu et al. (2016)提出来的DF-LSTMs(Deep Fusion LSTMs)网络，它包含了两个独立的LSTMs (Wu et al., 2017)来挖掘更高层次的文本语义信息，对实体与新闻正文进行多次递归融合。

定义$g_{1:i}$表示实体的子序列$\{g_1, g_2, \ldots, g_i\}(1 \leq i \leq N)$，$g'_{1:j}$表示新闻正文的子序列$\{g'_1, g'_2, \ldots, g'_j\}(1 \leq j \leq M)$，$I_{i,j}$表示子序列$g_{1:i}$和$g'_{1:j}$之间的相互作用，

$$h_{i,j} = \mathrm{h}^{g}_{i,j} \oplus h^{g'}_{i,j} \tag{16}$$

其中，$I^{g}_{i,j}$表示经过新闻正文LSTM的隐藏层编码的实体输出编码，$I^{g'}_{i,j}$表示经过实体LSTM的隐藏层编码的新闻正文输出编码。整个网络推演的过程如下：

$$h^{g}_{i,j}, c^{g}_{i,j} = LSTM\left(h_{i,j}, c^{g}_{i-1,j}, g_i\right) \tag{17}$$

$$h^{g'}_{i,j} c^{g'}_{i,j} = LSTM\left(H_{i,j}, c^{g'}_{i-1,j}, g'_i\right) \tag{18}$$

其中，$H_{i,j}$是包含过去信息的隐藏层，此方法可以看成Grid LSTMs (Kalchbrenner et al., 2015)的一种变种。

为了计算两段文本交互匹配信息，选用了Cosine(余弦相似度)、Bilinear(双线性变换)和Tensor Layer(张量变换)三种相似性度量方法来进行综合评定分数，给定两个向量u和v，三个相似度评分$s\,(u,v)$分别为:

$$Cosine : s\,(u,v) = \frac{u^T v}{u \cdot v} \tag{19}$$

$$Bilinear : s\,(u,v) = u^T \boldsymbol{M} v + b \tag{20}$$

$$TensorLayer : s\,(u,v) = f\left(u^T \boldsymbol{M}^{[1:c]} v + \boldsymbol{W}_{uv} \begin{bmatrix} u \\ v \end{bmatrix} + b\right) \tag{21}$$

其中，$\boldsymbol{M}$是两短文本交互的权重矩阵，$\boldsymbol{M}^i, i \in [1, \ldots, c]$是张量参数的一个切片，$\boldsymbol{W}_{uv}$和$b$是线性部分参数，$f\,(z) = \max\,(0, \mathrm{z})$。Cosine和Bilinear函数输出形式为矩阵，而Tensor Layer函数输出的形式为张量。Cosine相似度计算是一种常用做法，而Bilinear能够考虑不同维度之间的关联信息，因此相比Cosine方法能够捕获更复杂的交互信息。Tensor Layer在建模两个向量之间相互关系表现出了较大的优越性，且能够退化为Bilinear和点积相似度度量方法。

需要通过K-max池化层 (Shu et al., 2018)来整合这三种语义交互信息来得到最终的匹配得分。对于Cosine和Bilinear，经过K-Max池化层后可以得到K个数值，并以降序排列，组成一个新的向量q 。

对于Tensor Layer，每个张量切片返回K个数值形成一个向量，所有张量切片返回向量拼接在一起生成向量q。把向量q输入多层感知机(MLP)以获取更深层次交互向量表示r，然后用一个线性变换输出匹配得分值s，

$$s = W_s f\,(W_r q + b_r) + b_s \tag{22}$$

其中，W_r、W_s分别是权重矩阵，b_r和b_s为偏置向量，$f\,(\cdot)$是tanh函数。

5 实验结果分析

5.1 实验设置

由于网上开源的LTR数据集并没有给出相应的查询文本，而是给出若干个特征项，很难用作做对比数据集，因此本文选用文本匹配数据集STS-B、QQP、MRPC作为对比数据集。同时，本文提出的BERTCA模型将与DRCN (Kim et al., 2019)、ESIM (Chen et al., 2016)、BERT (Devlin et al., 2018) 、XLNetCA (Yang et al., 2019)、MV-LSTM (Wan et al., 2016)、ALBERTCA (Lan et al., 2019)、RoBERTaCA (Liu et al., 2019)进行对比实验，XLNetCA的意思就是在XLNet模型的基础上增加语义相关度计算模型，其他模型以此类推。

在实验参数方面，BERT中的隐藏层维度为1024，隐藏层激活函数为GELU，注意力层数为24层，注意力层dropout率为0.1，注意力头数为16个，词表大小为21128，最大位置编码为512。所有LSTM隐含层维度设为320，全连接网络MLP的隐含层维度设为128、32，dropout率为0.15，Tensor Layer中张量的c设为8，K-Max层的卷积核的宽度为2*2。

训练参数batchsize设为64，AdaW (Loshchilov and Hutter, 2017)优化算法的初始学习率设为1e-5，迭代训练次数epoch=300。语句序列的长度分别为ETSR512，其中实体为10，新闻正文为499，STS-B取128，MRPC取128，QQP取128。

所有任务都转化为排序任务，模型要使正样本排名比负样本高，因此选用精准度P@1和平均倒数排名MRR(Mean Reciprocal Rank)作为评估指标：

$$P@1 = \frac{1}{N}\sum_{i=1}^{N}\delta\left(r\left(S_Y^{+(i)}\right)=1\right) \tag{23}$$

$$MRR = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{r\left(S_Y^{+(i)}\right)} \tag{24}$$

其中，N是排序列表的长度，$S_Y^{+(i)}$指第i个排序列表中的正样本语句，$r(\cdot)$表示排序列表中语句的排名，δ是一个指示函数，即$\delta(true)=1$,$\delta(false)=0$。

5.2 结果分析

Table 2: 在ETSR、STS-B、QQP及MRPC数据集上的评估结果

模型	ETSR		STS-B		QQP		MRPC	
	P@1	MRR	P@1	MRR	P@1	MRR	P@1	MRR
DRCN	0.812	0.842	0.723	0.772	0.739	0.783	0.739	0.798
ESIM	0.845	0.873	0.693	0.750	0.715	0.763	0.754	0.801
MV-LSTM	0.864	0.896	0.741	0.785	0.703	0.754	0.772	0.830
BERT	0.864	0.882	0.870	0.927	0.864	0.935	0.843	0.901
XLNetCA	0.863	0.899	0.874	0.930	0.882	0.931	0.882	0.925
ALBERTCA	0.899	0.925	0.905	0.936	0.867	0.904	0.874	0.907
RoBERTaCA	0.902	0.933	0.902	0.931	0.876	0.913	0.905	0.933
BERTCA-Cosine	0.943	0.974	0.931	0.968	0.922	0.948	0.932	0.966
BERTCA-Bilinear	0.942	0.972	0.903	0.963	**0.938**	**0.962**	0.898	0.962
BERTCA-TL	**0.954**	**0.988**	**0.919**	**0.969**	0.882	0.951	**0.933**	**0.973**

本模型的参数量为370M，在参数量为330M的BERT基础上只增加了40M的参数量。经过计算，在8个Tesla P100 16G上的运行一个批次大约需要16分钟，而在XLNetCA、RoBERTaCA等模型的上大约需要20分钟，时间相差较大。表2显示了本文所提出的模型在ETSR、STS-B、QQP和MRPC数据集上与其他模型的对比结果，BERTCA-Cosine、BERTCA-Bilinear、BERTCA-TL分别代表计算语义相似度时所采用的三种不同计算方法。其中DRCN为单语义文档表达的计算模型，ESIM、MV-LSTM和BERTCA为多语义文档表达的计算模型，其他的都是直接计算模型。

从该表中可以发现BERTCA模型在三种相关度计算方式下结果略有差异，TL的效果普遍偏好，猜测该方法能够捕获两个文本不同位置特征向量之间的差异，相比于Cosine和Bilinear方

Table 3: 去除解码层、语义交互层层、上下文编码层在ETSR上的消融实验

模型	P@1	MRR
BERTCA-TL	**0.954**	**0.988**
-Decoder	0.923	0.967
-Coattention	0.889	0.925
-Context	0.912	0.953

法能够学到更有意义的交互信息。我们还发现Bilinear相似度计算方法在QQP数据集上的效果比较好，经观察发现QQP的两段文本的长度集中在5~30之间，并且文中的句式较为简单，词表前2000个词可以覆盖95%的句式，句式的变化较少导致最终形成的语义向量变化平顺，因此语义信息可能记录很完整，经过Bilinear计算后，得到了较好的结果。通过对比BERTCA模型与其他模型的效果，BERTCA-TL除了在MRPC上的效果略逊色于BERTCA-Bilinear以外，其他数据集上，达到了最优效果，说明本文提出的基于BERTCA模型在ETSR、STS-B、QQP和MRPC这3个数据集上具有明显的提升。

除此之外，为了更好地衡量不同网络参数对实验结果的影响，本文也进行了对比实验，通过选用不同参数量的预训练模型[8]，对模型的效果进行了验证。通过观察表4可以发现，虽然ALBERT或者XLNet对BERT模型进行了一定程度的改进，但是最终呈现出来的效果却不是最好的。得益于ALBERT的参数共享和词向量分解机制，ALBERT的参数量大大减少，运算速度大大提高，然而却破坏有效的语义信息，效果有所衰减，XLNet的训练模式则是词生成，语义理解的能力还有待提高。

Table 4: 不同参数量下的模型效果

模型	P@1	MRR
BERT-chinese-wwm-ext	**0.864**	**0.882**
BERT-chinese-wwm	0.834	0.850
BERT-chinese-base	0.776	0.798
ALBERT-chinese-Large	0.815	0.841
ALBERT-chinese-base	0.782	0.806
XLNet-chinese-mid	0.842	0.869
XLNet-chinese-base	0.822	0.846

本文还对语义解码层的Fusion-LSTM隐含层不同维度进行了分析。通过表5可以发现，当维度在160以下时，Fusion-LSTM的模型效果普遍偏低，而当维度到达了256维，模型呈现出了震荡的趋势，在320维的时候，到达了顶点，因此本文选用了320维作为最终模型的参数。

Table 5: 不同参数量下的模型效果

维度	P@1	MRR	维度	P@1	MRR
32	0.443	0.462	288	0.934	0.959
64	0.863	0.882	**320**	**0.954**	**0.988**
96	0.873	0.899	352	0.931	0.951
128	0.867	0.891	384	0.949	0.964
160	0.892	0.916	416	0.951	0.974
192	0.921	0.941	448	0.933	0.948
224	0.914	0.935	480	0.943	0.962
256	0.945	0.961	512	0.922	0.941

在ETSR数据集上，与最优的BERTCA-TL模型对比结果显示，去除解码层模块之后，模型的整体性能有明显的下滑，在P@1和MRR指标上分别下降了0.031和0.021，在没有解码层的情况下，语义交互的结果没法很有效的展现，我们曾将解码层替换成单纯的MLP网络，实验效果反而下降了，可见解码层在整个模型中的作用。另外，去除了语义交互层后，解码层得到了保留，但是在P@1和MRR指标上依然下降了0.065和0.063，可以发现实体与正文之间的语义交互层的重要性。对比“-Coattention”和“-Context”，“-Context”的实验结果反而有一定的提升，这是因为语义交互层的缺失，导致整体效果的下降，也从侧面表示了上下文编码层在整体结构上的提升偏小，而语义交互在整体结构上的重要性偏大。

[8] https://huggingface.co/models

(a) 无BERTCA　　(b) 有BERTCA

Figure 4: 基于语义相关度排序实例

为了更加形象的展示模型的成果，本文利用Solr开源搜索引擎框架对爬取的新闻进行可视化展示，如图4。该搜索引擎将搜索关键词与新闻标题进行比对，返回匹配结果，以关键词“360 公司”为例，搜索引擎返回了“东风小康风光360”、“经典落幕Xbox360正式停产！”这两条与“360 公司”关键词看似很匹配的结果，其实用户是想对360公司进行搜索，因此所有与360公司的有关的新闻应该在搜索结果的前列。

下面将BERTCA模型与搜索引擎结合，对搜索结果进行重排。首先通过训练好的BERTCA模型计算“360”与正文的语义相关度相关度，第一条新闻“360手机新掌门首次亮相否认360放弃手机业务”与“360”的语义相关度为0.964，说明实体与正文是密切相关度的，而在第3条新闻中，虽然标题中包含“360”，但是其实它是想表达“Xbox360”，在计算“360”与该条新闻的语义相关度时候，BERTCA模型发现了这种差别，语义相关度的结果为0.744，相较于上一条新闻，具有较大的差距。

6 结束语

针对搜索引擎存在“重形式，轻语义”，无法对搜索关键词和文本进行深层次语义理解的问题，本文提出了一种基于BERTCA的语义相关度计算模型，该模型融合了BERT动态语义编码方法和深度语义信息交互的协同注意力方法，并将排序方法与语义相关度方法进行结合，在ETSR、STS-B、QQP和MRPC这四个数据集上进行了对比实验，结果证明本文提出的模型能有效提升语义相关度的计算结果，并对搜索引擎的语义理解能力有较大的帮助。

参考文献

Qingyao Ai, Keping Bi, Jiafeng Guo, and W Bruce Croft. 2018. Learning a deep listwise context model for ranking refinement. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 135–144.

Christopher JC Burges. 2010. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11(23-581):81.

Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2016. Enhanced lstm for natural language inference. *arXiv preprint arXiv:1609.06038*.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-training with whole word masking for chinese bert. *arXiv preprint arXiv:1906.08101*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2333–2338.

Nal Kalchbrenner, Ivo Danihelka, and Alex Graves. 2015. Grid long short-term memory. *arXiv preprint arXiv:1507.01526*, 6:19–34.

Seonhoon Kim, Inho Kang, and Nojun Kwak. 2019. Semantic sentence matching with densely-connected recurrent and co-attentive information. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6586–6593.

Marius Köppel, Alexander Segner, Martin Wagener, Lukas Pensel, Andreas Karwath, and Stefan Kramer. 2019. Pairwise learning to rank by neural networks revisited: Reconstruction, theoretical analysis and practical performance. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 237–252. Springer.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Pengfei Liu, Xipeng Qiu, Jifan Chen, and Xuan-Jing Huang. 2016. Deep fusion lstms for text semantic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1034–1043.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Tie-Yan Liu. 2011. *Learning to rank for information retrieval*. Springer Science & Business Media.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Zhengdong Lu and Hang Li. 2013. A deep architecture for matching short texts. In *Advances in neural information processing systems*, pages 1367–1375.

Sean MacAvaney, Andrew Yates, Arman Cohan, and Nazli Goharian. 2019. CEDR: Contextualized embeddings for document ranking. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1101–1104.

Tao Qin, Tie-Yan Liu, Jun Xu, and Hang Li. 2010. Letor: A benchmark collection for research on learning to rank for information retrieval. *Information Retrieval*, 13(4):346–374.

Bo Shu, Fuji Ren, and Yanwei Bao. 2018. Investigating lstm with k-max pooling for text classification. In *2018 11th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, pages 31–34. IEEE.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Shengxian Wan, Yanyan Lan, Jiafeng Guo, Jun Xu, Liang Pang, and Xueqi Cheng. 2016. A deep architecture for semantic matching with multiple positional sentence representations. In *AAAI*, volume 16, pages 2835–2841.

Liang Wang, Sujian Li, Yajuan Lü, and Houfeng Wang. 2017. Learning to rank semantic coherence for topic segmentation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1340–1344.

Sheng Wang, Zhifeng Bao, J Shane Culpepper, Zizhe Xie, Qizhi Liu, and Xiaolin Qin. 2018. Torch: A search engine for trajectory data. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 535–544.

Qiang Wu, Christopher JC Burges, Krysta M Svore, and Jianfeng Gao. 2010. Adapting boosting for information retrieval measures. *Information Retrieval*, 13(3):254–270.

Wei Wu, Houfeng Wang, and Sujian Li. 2017. Bi-directional gated memory networks for answer selection. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, pages 251–262. Springer.

SHI Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. 2015. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, volume 8, pages 802–810.

Chenyan Xiong, Zhengzhong Liu, Jamie Callan, and Tie-Yan Liu. 2018. Towards better text understanding and retrieval through kernel entity salience modeling. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 575–584.

Chunlin Xu, Zhiwei Lin, Shengli Wu, and Hui Wang. 2019. Multi-level matching networks for text matching. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 949–952.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5753–5763.

Dawei Yin, Yuening Hu, Jiliang Tang, Tim Daly, Mianwei Zhou, Hua Ouyang, Jianhui Chen, Changsung Kang, Hongbo Deng, Chikashi Nobata, et al. 2016. Ranking relevance in yahoo search. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 323–332.

庞亮, 兰艳艳, 徐君, 郭嘉丰, 万圣贤, and 程学旗. 2017. 深度文本匹配综述. 计算机学报, 40(4):985–1003.

基于多任务学习的生成式阅读理解

钱锦[1]，**黄荣涛**[1]，**邹博伟**[1,2]*，**洪宇**[1]
苏州大学计算机科学与技术学院，苏州215000[1]
新加坡资讯通信研究院，新加坡138632[2]
{jaytsien,rthuang.suda}@gmail.com, zou_bowei@i2r.a-star.edu.sg,
tianxianer@gmail.com

摘要

生成式阅读理解是机器阅读理解领域一项新颖且极具挑战性的研究。与主流的抽取式阅读理解相比，生成式阅读理解模型不再局限于从段落中抽取答案，而是能结合问题和段落生成自然和完整的表述作为答案。然而，现有的生成式阅读理解模型缺乏对答案在段落中的边界信息以及对问题类型信息的理解。为解决上述问题，本文提出一种基于多任务学习的生成式阅读理解模型。该模型在训练阶段将答案生成任务作为主任务，答案抽取和问题分类任务作为辅助任务进行多任务学习，同时学习和优化模型编码层参数；在测试阶段加载模型编码层进行解码生成答案。实验结果表明，答案抽取模型和问题分类模型能够有效提升生成式阅读理解模型的性能。

关键词： 多任务学习 ； 生成式阅读理解

Generative Reading Comprehension via Multi-task Learning

Jin Qian[1],**Rongtao Huang**[1],**Bowei Zou**[1,2]*,**Yu Hong**[1]
School of Computer Science and Technology, Soochow University, Suzhou, 215000[1]
Institute for Infocomm Research, Singapore, 138632[2]
{jaytsien,rthuang.suda}@gmail.com, zou_bowei@i2r.a-star.edu.sg,
tianxianer@gmail.com

Abstract

Generative reading comprehension is a novel and challenging research in the field of machine reading comprehension. Compared with the mainstream extractive reading comprehension, generative reading comprehension model is no longer limited to extract answers from paragraphs, but can combine questions and paragraphs to generate natural and complete statements as answers. However, the existing generative reading comprehension model lacks the understanding of the boundary information of answers in paragraphs and the question type information. To solve such issues, this paper proposes a generative reading comprehension model based on multi-task learning. In the training phase, the model takes the answer generation task as the main task, and the answer extraction and question classification tasks as auxiliary tasks for multi-task learning. The model simultaneously learns and optimizes the parameters of the model encoding layer. Then it loads the encoding layer in the test phase to decode and generate the answers. The experimental results show that the answer extraction model and the question classification model can effectively improve the performance of the generative reading comprehension model.

Keywords: Multi-task Learning , Generative Reading Comprehension

* 通讯作者

1 引言

机器阅读理解在阅读和理解自然语言的基础上，根据文本内容回答用户提出的问题，是当前自动问答领域的研究热点之一。近年来，随着大规模阅读理解数据集的构建，如SQuAD (Rajpurkar et al., 2016)、HotpotQA (Yang et al., 2018)、CoQA (Reddy et al., 2018)，以及预训练模型的提出，如BERT (Devlin et al., 2018)、UniLM (Dong et al., 2019)、ENRIE-GEN (Xiao et al., 2020)，机器阅读理解技术取得了巨大发展。目前主流的机器阅读理解模型通常将答案设定为段落中的一个连续片段，这种抽取式阅读理解模型存在一定的局限性，其仅能直接以段落中的片段作为答案，导致在针对某些问题时，无法给出自然流畅的答案，例如表 1(a)中的True/False问题。此外，如果将问题与答案分离，仅根据答案无法获得完整清晰的信息。表 1(b)中例子所示，严格意义上说，抽取式模型给出的答案“Season 5(第5季)”并不通顺，在某些应用场景（如聊天机器人）中，会对用户体验造成影响。

表 1: 抽取式与生成式机器阅读理解

(a)	
段落	That all stops now, thanks to the creators of Sarcastic Font, which is italics, but in reverse. Simply genius! Here is their manifesto to support the need for a sarcasm font: For too long e-mails, instant messages, web pages and documents have been unable to fully communicate the subtleties of sarcasm. 这一切现在都停止了，感谢讽刺字体的创造者，这是斜体，但反过来。简直就是天才！以下是他们的宣言，以支持讽刺字体的必要性：太长时间以来，电子邮件、即时消息、网页和文档都无法充分传达讽刺的微妙之处。
问题	is there a font for sarcasm? 有讽刺字体吗?
抽取式答案	Sarcastic Font 讽刺字体
生成式答案	Yes, there is sarcastic font for sarcasm. 是的，讽刺有讽刺字体。
(b)	
段落	Longmire will ride again. Netflix announced on Friday that they had renewed the Western for a 10-episode Season 5, just seven weeks after the plucked-from-the-ashes fourth season debuted on the streaming service... 西镇警魂将再次上映。Netflix周五宣布，他们已经续订了这部西部片第5季，共10集。而就在7周前，脱胎换骨第四季才在Netflix上首播...
问题	how many seasons does longmire have? 西镇警魂有多少季?
抽取式答案	Season 5 第5季
生成式答案	Longmire has 5 seasons. 西镇警魂有5季。

与抽取式阅读理解相比，生成式阅读理解不再局限于直接从段落片段中抽取答案，而是参考段落、问题、甚至词表，生成更为自然和完整的表述作为答案。例如，表 1(a)中，生成式阅读理解模型给出的答案能够与问题更自然地衔接；而表 1(b)中的生成式答案与抽取式答案相比更完整，确保了答案在独立于问题和段落时仍能够保持完整的信息。而现有的生成式阅读理解模型通常基于整个段落生成答案，缺乏对答案边界和问题类型信息的理解，生成答案有时未参考段落中用于生成答案的片段以及问题的具体类型，导致生成的答案和真实答案之间存在差异。

为解决上述问题，本文提出一种基于多任务学习的生成式阅读理解框架。多任务学习能够学到多个关联任务的共享表示，并适应这些不同但相关的任务目标，使主任务获得更强的泛化性能。基于此，本文将答案生成任务作为主任务，将答案抽取和问题分类任务作为辅助任务，在训练阶段，通过多任务学习的参数共享机制，让模型生成答案的同时加强对答案边界和问题

类型的理解，从而让答案抽取和问题分类任务辅助答案生成任务，最终提升生成式阅读理解模型的泛化性能。

针对答案生成任务，本文提出的生成式阅读理解模型由编码层和任务层组成。其中，编码层基于深度双向Transformer (Vaswani et al., 2017)编码器，并借鉴UniLMv2 (Bao et al., 2020)模型中特殊设计的自注意力掩码机制控制答案生成过程中的可见信息；任务层分为答案生成模型、答案抽取模型和问题分类模型，答案生成模型在训练阶段通过预测被遮蔽答案单词的原始信息，增强模型的生成能力，在测试阶段直接采用训练好的编码层，以及束搜索（Beam search） (Sutskever et al., 2014)对问题和段落进行解码，生成答案；答案抽取模型采用指针网络 (Vinyals et al., 2015)识别答案在段落中的起始位置和结束位置；问题分类模型采用线性层判断问题的具体类型。

本文实验采用CoQA (Reddy et al., 2018)、MS MARCO (Bajaj et al., 2018)和NarrativeQA (Kočiský et al., 2018)三个阅读理解数据集验证模型性能。实验结果表明，本文模型在CoQA语料上取得了86.7%的F1值，比目前最好的生成模型提升了2.20%；在MS MARCO和NarrativeQA语料上的BLEU-1值分别为80.53%和57.94%，分别比目前最好的系统提升了2.39%和3.81%（绝对性能提升）。

本文的主要贡献如下：

- 提出基于多任务学习的生成式阅读理解模型，通过答案抽取模型和问题分类模型优化生成式阅读理解模型的性能；
- 本文在三个阅读理解数据集上进行详细实验，均取得了目前生成式模型的最佳性能。

2 相关工作

2.1 生成式机器阅读理解

近年来，随着如SQuAD (Rajpurkar et al., 2016)、TriviaQA (Joshi et al., 2017)、SearchQA (Dunn et al., 2017)、HotpotQA (Yang et al., 2018)和QuAC (choi et al., 2018)等大规模阅读理解数据集的构建，以及在以神经网络为代表的深度学习技术和计算资源的推动下，机器阅读理解领域获得了巨大发展。目前，MS MARCO (Bajaj et al., 2018)、NarrativeQA (Kočiský et al., 2018)和CoQA (Reddy et al., 2018)等数据集提供人工编辑生成的答案，要求机器能够理解问题和段落中相关句子的潜在联系，依赖一定的推理能力生成正确的答案，而非简单的文本匹配。随着生成式阅读理解数据集的发布以及自然语言生成技术的发展，研究者开始关注使用生成模型来解决阅读理解问题。McCann等 (2018)和Bauer等 (2018)采用基于RNN的指针生成机制进行单文档阅读理解答案的生成，Tan等 (2018)在多文档阅读理解中采用管道（Pipeline）的方法，先从多篇文档中抽取出最有可能成为答案的片段，然后将该片段作为答案合成模块（Seq2Seq生成模型）的一个特征,最后综合问题、文档和抽取特征合成答案。而本文所提出的是端到端的生成式阅读理解模型，旨在让答案生成、答案抽取以及问题分类共享模型编码层参数并进行优化，最终达到提升生成模型性能的目的。

目前，预训练模型如Mass (Song et al., 2019)、UniLM (Dong et al., 2019)、BART (Lewis et al., 2019)以及ERNIE-GEN (Xiao et al., 2020)等在各个自然语言生成任务中相继取得最佳性能，这些模型只需在特定任务(如阅读理解、文本摘要以及机器翻译等)进行微调就能取得令人满意的成绩。其中，Bao等 (2020)提出UniLMv2模型，其使用一种新颖的伪遮蔽语言模型(pseudo-masked language model, PMLM)将自编码模型和部分自回归模型统一起来训练，在问题生成、自动摘要等多个领域取得当前的最佳性能。本文将UniLMv2模型作为基线模型，并在此基础上进行多任务学习的实验。

2.2 多任务学习

多任务学习是一种提高泛化性能的迁移机制，现有研究表明它在提高模型泛化能力上十分有效。该机制同时学习多个相关任务，让这些任务同时共享知识，利用任务之间的相关性，提升每个任务的泛化性能。多任务学习的一般做法是，在所有任务上共享模型编码层，而针对特定的任务层有所区别。例如，Wang等 (2018)证明通过共享文档排序任务和多文档阅读理解任务的编码层能够提升整体的性能。Nishida等 (2019)在阅读理解、文档排序和问题

分类三种任务上共享问题和文章阅读模块，有效提升了模型的整体性能。Liu等 (2019)提出的MT-DNN模型在BERT (Devlin et al., 2018)的基础上对4种下游任务单句分类、成对文本分类、文本相似度打分和相关性排序进行联合微调，在性能上较BERT有了极大提升，证明了多任务学习能有效提升模型的泛化性能。此外，与MT-DNN模型在下游任务上进行多任务学习不同，ERNIE2.0 (Sun et al., 2020)在模型预训练阶段引入多任务学习，通过和多个先验知识库进行交互并采用增量学习的方式，使得模型能够学会多样化的语言知识，最终在各种下游任务上性能得到提升。

受到上述工作的启发，为了解决现有的生成式阅读理解模型缺乏对答案边界信息和问题类别信息的理解的问题，本文提出基于多任务学习的生成式阅读理解模型，通过答案抽取模型和问题分类模型优化生成式阅读理解模型性能。

3 基于多任务学习的生成式阅读理解模型

本章首先给出生成式阅读理解问题的形式化定义；然后介绍模型的编码层；最后介绍模型的任务层，其具体由答案生成模型、答案抽取模型和问题分类模型三部分组成。基于多任务学习的生成式阅读理解模型框架如图 1所示。

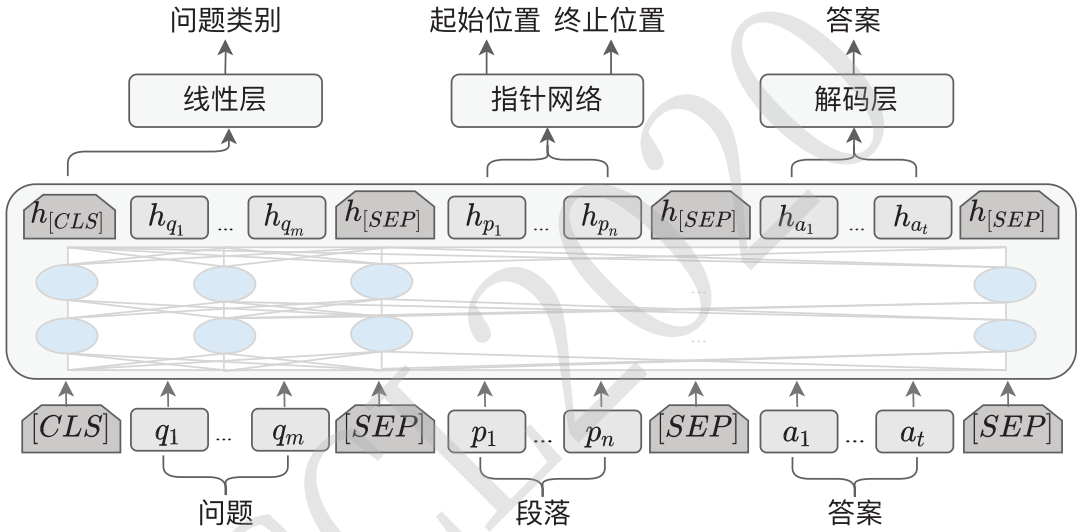


图 1: 基于多任务学习的生成式阅读理解模型框架

3.1 问题定义

给定问题和段落分别表示为$Q = \{q_i\}_{i=1}^{m}$和$P = \{p_i\}_{i=1}^{n}$，答案表示为$A = \{a_i\}_{i=1}^{t}$，其中m、n、t分别表示问题、段落和答案的长度。在生成式阅读理解中，本文将问题和段落组成源序列，答案作为目标序列，目标是根据Q和P，自动生成符合语义的目标答案A。该任务的目标$\bar{a}$表示为

$$\bar{a} = \arg\max_{a} P(a|Q,P) \tag{1}$$

其中$P(a|Q,P)$表示在给定Q和P的条件下，生成答案的对数条件概率。

3.2 编码层

本文基于预训练模型UniLMv2[1] (Bao et al., 2020)构建编码层，采用预训练的BERT进行问题和段落的交互，得到其表示，并在BERT的基础上改进了注意力遮蔽矩阵，采用伪遮蔽语言模型，使得模型能在阅读理解任务上根据问题和段落逐字或逐片段预测被遮蔽的答案。以下介绍编码层的具体工作原理和过程。

预处理阶段，采用WordPiece分词工具，将问题、段落和答案分词，得到子词（sub-word）级别的若干词项，其中对答案中的部分词项以一定概率进行遮蔽，并将其拼接后将

[1]https://github.com/microsoft/unilm

作为模型输入。每个词项表示为词向量$\mathbf{WE}(w_i)$、段向量$\mathbf{SE}(w_i)$和位置向量$\mathbf{PE}(w_i)$的和，维度均为d_w，其中词向量用于表示不同词项，段向量用于区分词来自源序列还是目标序列，位置向量用于表示词在输入序列中的绝对位置。词向量$\mathbf{X}_i$表示为：

$$\mathbf{X}_i = \mathbf{WE}(w_i) + \mathbf{SE}(w_i) + \mathbf{PE}(w_i)$$

其中w_i为第i个位置的词项。

本文将词向量集合表示为$\{\mathbf{X}_i\}_{i=1}^{|x|}$，则输入序列表示为$\mathbf{H}^0 = [\mathbf{X}_1, ..., \mathbf{X}_{|x|}]$，其中$|x|$为输入序列的长度。UniLMv2的编码层使用12层堆叠的Transformer网络，每经过一层Transformer网络都能得到不同抽象层次的上下文表示：

$$\mathbf{H}^l = [h_1^l, ..., h_{|x|}^l] = Transformer_l(\mathbf{H}^{l-1}), l \in [1, 12] \tag{2}$$

其中l为第l层Transformer网络，h_i^l为第i个词项的l层隐层表示。

Tranformer网络由多头自注意力机制和前向神经网络两个子层组成，每个子层均使用残差连接和层正则化，因此每个子层的输出可表示为：

$$LayerNorm(x + Sublayer(x))$$

第l层Transformer网络的自注意力头$\mathbf{A}_l$计算如下：

$$\mathbf{A}_l = softmax(\frac{\mathbf{Q}_l\mathbf{K}_l^\mathsf{T}}{\sqrt{d_k}} + \mathbf{M})\mathbf{V}_l \tag{3}$$

$$\mathbf{Q}_l = \mathbf{H}^{l-1}\mathbf{W}_l^Q, \mathbf{K}_l = \mathbf{H}^{l-1}\mathbf{W}_l^K, \mathbf{V}_l = \mathbf{H}^{l-1}\mathbf{W}_l^V \tag{4}$$

$$\mathbf{M}_{i,j} = \begin{cases} 0, & \text{允许被注意} \\ -\infty, & \text{不允许被注意} \end{cases} \tag{5}$$

其中$\mathbf{Q}_l$，$\mathbf{K}_l$和$\mathbf{V}_l$分别代表第l层注意力机制中的查询（query）向量、键（key）向量和值（value）向量，d_k为向量$\mathbf{K}_l$的维度，$\mathbf{W}_l^Q, \mathbf{W}_l^K, \mathbf{W}_l^V \in \mathbb{R}^{d_h \times d_k}$为可学习参数矩阵，$\mathbf{H}^{l-1} \in \mathbb{R}^{|x| \times d_h}$为$l-1$层的隐层表示，$\mathbf{M}$为生成式阅读理解模型注意力遮蔽矩阵，如图2所示。

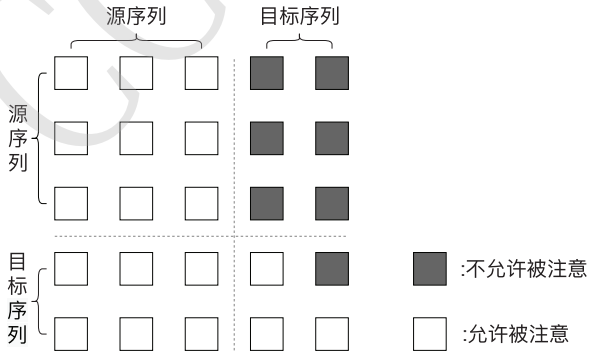


图 2: 注意力遮蔽矩阵

通过上述词嵌入层和Tranformer网络，得到输入序列的上下文表示$\mathbf{H}^1, \mathbf{H}^2, ..., \mathbf{H}^{12}$。本文使用最后一层输出$\mathbf{H}^{12}$作为整个序列的表示。$\mathbf{H}^{12}$中包含问题、段落和答案表示，其中，段落表示部分记作$\mathbf{H}^p$，答案表示部分记作$\mathbf{H}^a$，问题类别表示记作$\mathbf{H}^{cls}$。根据图 2所示的注意力遮蔽矩阵可知，问题和段落不会和答案进行交互，保证了训练和测试阶段$\mathbf{H}^p$和$\mathbf{H}^{cls}$所含信息的一致性。

3.3 任务层

作为基于多任务学习框架的核心部分，任务层由答案生成模型、答案抽取模型和问题分类模型三部分构成。

答案生成模型 训练阶段，真实答案会以一定概率被随机遮蔽，并且同时保留其原始位置信息来实现部分自回归（随机预测答案被遮蔽的片段），答案中被遮蔽的词项在经过编码后得到答案表示$\mathbf{H}^a$。答案生成模块通过解码层对原始答案中被遮蔽的词项进行预测来生成答案。具体来说，$\mathbf{H}^a$首先经过线性层并用Gelu函数激活后进行层正则化：

$$\mathbf{H}^a = LayerNorm(Gelu(Linear(\mathbf{H}^a))) \tag{6}$$

然后通过线性层将每个被遮蔽的词项映射到模型词表空间，获得预测分数。最后，使用SoftMax函数计算词的概率向量$\mathbf{a}$：

$$\mathbf{a} = SoftMax(Linear(\mathbf{H}^a)) \tag{7}$$

本文采用有标签平滑优化的交叉熵损失函数计算答案生成模型的目标函数：

$$L_{generate} = \sum_{t=1}^{T}\sum_{k=1}^{K} y_{\mathbf{a}_t^k} \cdot \log \mathbf{a}_t^k \tag{8}$$

其中T表示答案的长度，K表示词表的大小，$y_{\mathbf{a}_t^k}$表示答案中第t个位置经过标签优化的真实标签，$\mathbf{a}_t^k$表示答案中第t个位置的预测标签。注意，本文只对答案中被遮蔽的词项计算损失。

测试阶段，模型对于输入的问题和段落，每个时间步经解码层预测当前词的生成概率，同时使用束搜索每次保留生成概率最大的前k个序列，直至模型预测出[EOS]终止符结束解码。最后，模型将束搜索结果中生成概率最大的序列解码输出，其概率计算为：

$$P(A|Q,P) = P(a_1|Q,P)P(a_2|Q,P,a_1)...P([EOS]|Q,P,a_1,a_2,...) \tag{9}$$

答案抽取模型 经过编码层后，段落被表示为矩阵$\mathbf{H}^p$，答案抽取模型通过指针网络对答案的起始和终止位置进行识别。具体地，$\mathbf{H}^p$分别经过线性层得到对应起始位置分数和终止位置的分数，并通过SoftMax函数对分数进行归一化，得到相应的概率向量：

$$\mathbf{s}, \mathbf{e} = SoftMax(Linear(\mathbf{H}^p)) \tag{10}$$

其中$\mathbf{s}$为预测答案的起始位置概率向量，$\mathbf{e}$为答案终止位置概率向量，$\mathbf{s}$和$\mathbf{e}$由不同参数的线性层计算得到。

本文采用交叉熵损失函数计算答案抽取模型的目标函数：

$$L_{extract} = y_{\mathbf{s}} \cdot \log \mathbf{s} + y_{\mathbf{e}} \cdot \log \mathbf{e} \tag{11}$$

其中$y_{\mathbf{s}}$表示真实答案的起始位置概率向量，$y_{\mathbf{e}}$表示真实答案的终止位置概率向量。

问题分类模型 由于CoQA数据集中存在多种问题类型，包括事实型问题（Factoid question）、是非类问题（True/False question）和不可回答问题（Unanswerable question）。针对不同类型的问题，答案的模式通常差别较大，例如针对是非类问题，答案通常以“Yes/No”开头。本文采用4种问题类型标签{0: yes; 1: no; 2: unanswerable; 3: factoid}，以上四种问题类型（其中是非类问题被分为两种不同类型）。如图 1所示，输入经过编码后，取出[CLS]表示用于获得问题类型表示即$\mathbf{H}^{cls}$，并经过线性层为问题类型打分，最后将分数进行归一化后形成分类概率：

$$\mathbf{c} = SoftMax(Linear(\mathbf{H}^{cls})) \tag{12}$$

其中，$\mathbf{c}$代表问题类型的分数向量。

本文采用交叉熵损失函数计算问题分类模型的目标函数：：

$$L_{cls} = \sum_{k=1}^{K} y_{\mathbf{c}^k} \cdot \log \mathbf{c}^k \tag{13}$$

其中$K = 4$表示问题类别数，$y_{\mathbf{c}^k}$表示真实类别标签，$\mathbf{c}^k$表示预测类别标签。

多任务学习 本文采用多任务学习的方法，在训练阶段同时学习和更新答案生成、答案抽取和问题分类模块共享的编码层参数，让答案抽取和问题分类任务辅助答案生成任务提升阅读理解模型的性能。模型的损失由生成损失、抽取损失和分类损失三部分共同组成，整个模型的目标函数为：

$$LOSS = L_{generate} + \lambda_1 L_{extract} + \lambda_2 L_{cls} \tag{14}$$

其中λ_1和λ_2为调和系数，用于调节辅助任务权重。

4 实验

本章首先介绍生成式阅读理解任务数据集和实验设置，然后报告本文提出的基于多任务的生成式阅读理解模型性能，并针对实验结果进行分析。

4.1 生成式阅读理解任务数据集

现有阅读理解数据集大多针对抽取式模型，即答案为篇章中的一个片段，如SQuAD (Rajpurkar et al., 2016)、HotpotQA (Yang et al., 2018)等。采用这些数据集无法全面评价生成式阅读理解模型，与抽取式模型相比，其在答案的可读性、表述的完整性、以及应对多段答案的问题上，均有较大区别(请见本文第一章)。基于上述原因，本文实验中采用以下三个数据集。

- CoQA(Conversational Question Answering)[2]

  CoQA基于多个领域的多轮对话进行构建，并保持了人类对话简短的特征，存在大量指代和省略现象，问题和答案普遍偏短。值得注意的是，为了保证该数据集尽可能贴近自然对话，其中78%的答案经过人工编辑；此外，该数据集中存在较多的是非类问题(19.8%)和不可回答问题(1.3%)，部分问题无法仅采用抽取式阅读理解模型回答 (Reddy et al., 2018)。尽管如此，目前在CoQA评测榜单上排名较高的均为抽取式模型，而生成式模型，如UniLM和ERNIE-GEN，仅报告了在验证集上的性能，因此，本文将CoQA的验证集作为测试集评价系统性能，调参使用的验证集从CoQA训练集中划分。

- MS MARCO(Microsoft Machine Reading Comprehension)[3]

  MS MARCO是一个多文档问答数据集 (Bajaj et al., 2018)，其中特别提供了一个自然语言生成(NLG)子数据集，该数据集由人工编辑答案，其答案并非严格匹配文档中的片段，因此，本文采用MS MARCO(NLG)作为评价生成式阅读理解模型的数据集。注意，由于该数据集还包含了文档检索任务，而本文研究重点仅在于机器阅读理解，因此，仅采用人工编辑答案时依据的文档，即最佳文档(golden passage)；此外，由于MS MARCO评测榜单上，NLG数据集同样包含了文档检索任务，因此本文仅报告模型在MS MARCO(NLG)验证集上的结果。

- NarrativeQA[4]

  NarrativeQA是一个生成式阅读理解数据集，该数据集基于书本故事和电影脚本构建，答案由人工编辑 (Kočiský et al., 2018)。本文基于数据集的摘要子集进行阅读理解，并在其测试集上进行测试。

表 2列出了本文所采用三个数据集的统计数据。CoQA中存在28.7%的命名实体类问题、19.6%的名词短语类问题和9.8%的数字类问题；NarrativeQA中存在30.54%的人名类问题、9.73%的地点类问题和约10%左右的事件、实体、数字类问题，且CoQA和NarrativeQA明确允许简短、自然的答案，因此CoQA和NarrativeQA的答案普遍较短。MS MARCO(NLG)中存在53.12%的描述型问题，且答案会融入问题信息形成完整的表述，答案普遍较长。

4.2 实验设置

本文使用的模型为微软开源的unilm1.2-base-uncased[5]，该模型在大多数自然语言生成任务上取得了最佳性能。针对不同数据集，表 3列出了模型使用的超参数设置。

[2]https://stanfordnlp.github.io/coqa/
[3]https://microsoft.github.io/msmarco/
[4]https://github.com/deepmind/narrativeqa
[5]https://unilm.blob.core.windows.net/ckpt/unilm1.2-base-uncased.bin

表 2: CoQA、MS MARCO和NarrativeQA数据集(#问题数)

数据集	CoQA	MS MARCO(NLG)	NarrativeQA(Summary)
训练集#	108,647	153,725	32,747
验证集#	7,983	12,467	3,461
测试集#	-	-	10,557
段落平均长度	271	53	659
问题平均长度	5.5	6	9.83
答案平均长度	2.7	14	4.73

表 3: 参数设置

参数描述	CoQA	MS MARCO	NarrativeQA
max_src_len	470	176	470
max_tgt_len	42	40	42
λ_1	0.245	0.1	0.1
λ_2	1	0	0
batch size	48	48	48
学习率	2e-5	7e-5	2e-5
Warmup率	0.1	0.02	0.1
epoch	10	2	10

在CoQA多轮对话数据集中，当前问题可能存在指代或省略现象，因此本文选取当前问题之前的至多两轮问答对作为对话历史，并与当前问题进行拼接当作完整的问题Q，同时使用上一轮答案和当前问题的词在段落中出现的频率选取文章中最佳的段落作为段落P。训练时，根据答案A计算出其在段落P中的起始位置和终止位置(答案不在段落中时，起始位置和终止位置均设为0)。实验中，问题最大长度设为60，问题和段落(源序列)的最大长度为470，答案(目标序列)的最大长度为42，该数据处理与 (Dong et al., 2019)论文里的方法保持一致。模型的优化器为AdamW。

在MS MARCO多文档阅读理解数据集中，每个问题Q会给定10个参考段落，本文直接选取最佳的段落进行拼接作为段落P。训练时，根据答案A计算出其在段落P中的起始位置和终止位置(答案不在段落中时，起始位置和终止位置均设为0)。实验中，问题和段落(源序列)的最大长度为176，答案(目标序列)的最大长度为40。模型的优化器为AdamW。

在NarrativeQA数据集中，本文使用问题Q的词在段落中出现的频率选取摘要中最佳的段落作为段落P。训练时，使用F1值选取段落P中与答案A最为接近的片段作为抽取答案，并根据抽取答案计算出答案A在段落P中的起始位置和终止位置。实验中，问题和段落(源序列)的最大长度为470，答案(目标序列)的最大长度为42。模型的优化器为AdamW。

本文在CoQA数据集上使用F1值 (Rajpurkar et al., 2016)来评价模型的性能，在MS MARCO和NarrativeQA数据集上使用BLEU (Papineni et al., 2002)和ROUGE-L (Lin, 2004)来评价模型的性能。

4.3 实验结果与分析

为了验证本文基于多任务的生成式阅读理解方法的有效性，本文与以下阅读理解模型进行了比较:

- **UniLM**：由Dong等 (2019)提出，是第一个在CoQA数据集上报告实验性能的预训练生成模型，本文在实验设置上和它保持一致。

- **ERNIE-GEN**：由Xiao等 (2020)提出的基于多流（multi-flow）机制生成完整语义片段的预训练生成模型，在CoQA生成式阅读理解中达到了目前最好的性能。

- **Masque**：由Nishida等 (2019)提出的多风格生成式阅读理解模型，在MS MARCO(NLG)和NarrativeQA数据集的相关指标上达到了目前的最好性能。
- **UniLMv2**：由Bao等 (2020)提出，采用伪遮蔽语言模型的预训练生成模型，是UniLM的改进版本。本文使用UniLMv2分别在三个数据集上进行实现作为我们的基线模型，并简单修复了wordpiece分词在解码时出现的分词错误。
- **MLT-Model**：本文提出的基于多任务学习的生成式阅读理解模型，由答案抽取和问题分类任务辅助生成式阅读理解模型。

表 4: 模型在CoQA验证集上的性能

模型	F1
UniLM	82.5
ENRIE-GEN	84.5
UniLMv2	86.1
MLT-Model	**86.7**

表 5: 模型在CoQA验证集的消融实验

模型	F1
MLT-Model	**86.7**
w/o cls	86.0
w/o extract	86.2

表 4为本文提出的模型在CoQA验证集上的性能，我们的模型在F1指标上比当前性能最好的生成式模型ENRIE-GEN提升了2.2%，同时较基线模型UniLMv2提升了0.6%。本文针对预训练生成模型在答案解码时出现的子词结合不准确问题加以修复，实现的基线模型UniLMv2高于原始版本的性能，较ENRIE-GEN提升1.6%的F1值。表 5列出了本文模型在CoQA上的消融实验性能，在去除答案抽取任务和问题分类任务之后，性能较MTL-Model分别下降0.5%和0.7%的F1值。这是由于CoQA中存在20%左右的是非类问题和不可回答问题，这两类问题在训练阶段答案的起始和终止位置均设为0，因此仅用答案抽取任务辅助生成模型，会弱化模型对这两类问题的生成能力；而仅用问题分类任务来辅助生成模型，模型会缺少对答案在段落中边界信息的理解，所以只有将答案抽取和问题分类任务一起和答案生成任务进行多任务学习才能从整体上提升生成模型的性能。

表 6: 模型在MS MARCO(NLG)验证集(Golden Passage)上的性能

模型	BLEU-1	BLEU-4	ROUGE-L
Masque	78.14	-	78.80
UniLMv2	79.76	68.87	80.09
MLT-Model	**80.53**	**69.82**	**80.64**

表 6为本文提出的模型在MS MARCO (NLG)验证集上选取最佳文档的性能表现。本文模型较基线模型UniLMv2在BLEU-1指标上提升0.77%，BLEU-4指标上提升0.95%，ROUGE-L指标上提升0.55%。这是由于MS MARCO(NLG)数据集中答案和选定段落中的部分片段相似度较高，答案抽取任务能够辅助模型关注答案在段落中的边界信息，并增强生成模型对问题和段落中答案片段之间关系的理解，最终提升生成模型的性能。我们在同样设置下和Masque模型进行了对比，本文所提模型在BLEU-1指标上提升了2.39%，ROUGE-L指标上提升了1.84%。这主要由于Masque模型仅使用静态的预训练词向量并基于Transformer网络进行答案生成，而本文模型基于网络更加复杂的预训练模型UniLMv2生成答案，因此在实验性能上取得较大提升。

表 7为本文模型在NarrativeQA(summary)测试集上的性能表现。本文模型较基线模型UniLMv2在BLEU-1指标上提升0.39%，BLEU-4指标上提升0.61%，ROUGE-L指标上提升0.1%。NarrativeQA数据集的答案长度普遍偏短，因此我们的模型并未在ROUGE-L指标上有明显提升，但是BLEU指标证明了答案抽取任务有助于生成模型生成更准确的答案。此外本文模型较目前性能最好的Masque模型在BLUE-1指标上提升了3.81%，BLEU-4指标上提升了1.24%，但在ROUGE-L指标上下降了0.53%。可能的原因是Masque模型基于整个摘要生成答案，而本文的模型是基于规则选取的滑窗作为段落来进行生成式阅读理解，在选取滑窗时丢失了部分性能；Masque模型在该数据集使用MS MARCO数据进行多风格学习，而本文模型并未

表 7: 模型在NarrativeQA(summary)测试集上的性能

模型	BLEU-1	BLEU-4	ROUGE-L
Masque(NarrativeQA + MS MARCO)	54.11	30.43	**59.87**
Masque(NarrativeQA)	48.70	20.98	54.74
UniLMv2	57.53	31.06	59.24
MLT-Model	**57.92**	**31.67**	59.34

采用增加额外训练数据的方法训练模型。我们还比较了在相同训练数据的情况下，本文模型较Masque模型在BLEU-1指标上提升了8.83%,BLEU-4指标上提升了10.69%，ROUGE-L指标上提升了4.6%。该提升较在MS MARCO(NLG)数据集上更为显著，主要原因为NarrativeQA的答案更偏向于推理性质的概括总结，而MS MARCO(NLG)的答案则更偏向于基于段落中的答案片段进行完整的表述，这也表明了MS MARCO(NLG)的任务难度比NarrativeQA小，预训练模型在推理方法更占优势。

5 结语

本文针对生成式阅读理解模型缺乏答案边界和问题分类信息理解的问题，提出一种基于多任务学习的生成式阅读理解模型，通过答案抽取模型和问题分类模型优化生成式阅读理解模型。在三个阅读理解数据集上的实验结果表明，本文提出的基于多任务的生成式阅读理解模型能够有效地学习答案的边界信息和问题分类信息，在三个数据集上均取得了目前生成式模型的最好性能。在未来的工作中，我们将研究如何将该模型迁移至面向长文本的机器阅读理解任务上，使得该模型能够学习整个长文本的同时确定答案的边界信息，并以此生成答案。

致谢

本文工作得到国家自然科学基金（基金号61703293，61672368，61672367），江苏高校优势学科建设工程资助项目资助。

参考文献

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2018. MS MARCO: a human-generated machine reading comprehension dataset. *Computing Research Repository (CoRR)*, arXiv:1611.09268. Version 3.

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, NanYang, Xiaodong Liu, Yu Wang, Songhao Piao, Jian-feng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2020. Unilmv2: Pseudo-masked language models for unified language model pre-training. *arXiv preprint arXiv:2002.12804*.

Lisa Bauer, Yicheng Wang, and Mohit Bansal. 2018. Commonsense for generative multi-hop question answering tasks. In *Empirical Methods in NaturalLanguage Processing (EMNLP)*, pages 4220–4230.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the EMNLP*, pages 2174– 2184, Brussels, Belgium.

Jacob Devlin, MingWei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *Computing Research Repository (CoRR)*, arXiv:1810.04805. Version 1.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*.

Matthew Dunn, Levent Sagun, Mike Higgins, Ugur Guney, Volkan Cirik, and Kyunghyun Cho. 2017. SearchQA: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, and Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistic (TACL)*, 6:317-328

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Proc. ACL Workshop on Text Summarization Branches Out*.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *Computing Research Repository (CoRR)*, arXiv:1806.08730. Version 1.

Kyosuke Nishida, Itsumi Saito, Kosuke Nishida, Kazutoshi Shinoda, Atsushi Otsuka, Hisako Asano, and Junji Tomita. 2019. Multi-style generative reading comprehension. In *ACL*.

Kishore Papineni, Salim Roukos, Todd Ward, and WeiJing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Association for Computational Linguistics (ACL)*, pages 311–318.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Empirical Methods in Natural Language Processing(EMNLP)*, pages 2383–2392.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2018. CoQA: A conversational question answering challenge. *Computing Research Repository (CoRR)*, arXiv:1808.07042. Version 1.

Ilya Sutskever, Oriol Vinyals, and Quoc V.Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in neural information processing systems*, pages 3104–3112.

Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. In ICLR.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked Sequence to Sequence Pre-training for Language Generation. In *Proceedings of the 36th InternationalConference on Machine Learning*, volume 97, pages 5926–5936. PMLR.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie 2.0: A continual pre-training framework for language understanding. In *AAAI*.

Chuanqi Tan, Furu Wei, Nan Yang, Bowen Du, Weifeng Lv, and Ming Zhou. 2018. S-net: From answer extraction to answer synthesis for machine reading comprehension. In *Association for the Ad-vancement of Artificial Intelligence (AAAI)*, pages 5940–5947.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukaszKaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors,*Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 2692–2700.

Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerald Tesauro, Bowen Zhou, and Jing Jiang. 2018. Reinforced reader-ranker for open-domain question answering. In *Association for the Advancement of Artificial Intelligence (AAAI)*, pages 5981–5988.

Dongling Xiao, Han Zhang, Yukun Li, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2020. ERNIE-GEN: An Enhanced Multi-Flow Pre-training and Fine-tuning Framework for Natural Language Generation. *arXiv preprint arXiv:2001.11314*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W.Cohen, Ruslan Salakhutdinov, and Christopher D.Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In EMNLP.

基于多头注意力和 BiLSTM 改进 DAM 模型的中文问答匹配方法

秦汉忠	于重重*	姜伟杰	赵霞
北京工商大学	北京工商大学	北京工商大学	北京工商大学
793973698@qq.com	chongzhy@vip.sina.com	359206371@qq.com	zhaox@btbu.edu.cn

摘要

针对目前检索式多轮对话深度注意力机制模型 DAM（Deep Attention Matching Network）候选回复细节不匹配和语义混淆的问题，本文提出基于多头注意力和双向长短时记忆网络（BiLSTM）改进 DAM 模型的中文问答匹配方法，该方法采用多头注意力机制，使模型有能力建模较长的多轮对话，更好的处理目标回复与上下文的匹配关系。此外，本文在特征融合过程中采用 BiLSTM 模型，通过捕获多轮对话中的序列依赖关系，进一步提升选择目标候选回复的准确率。本文在豆瓣和电商两个开放数据集上进行实验，实验性能均优于 DAM 基线模型，R_{10}@1 指标在含有词向量增强的情况下提升了 1.5%。

关键词：检索式多轮对话；DAM；多头注意力；BiLSTM

Chinese question answering method based on multi-head attention and BiLSTM improved DAM model

Hanzhong Qin	Chongchong Yu*	Weijie Jiang	Xia Zhao
Beijing Technology and Business University	Beijing Technology and Business University	Beijing Technology and Business University	Beijing Technology and Business University
793973698@qq.com	chongzhy@vip.sina.com	359206371@qq.com	zhaox@btbu.edu.cn

Abstract

Aiming at the current problem of Deep Attention Matching Network(DAM) can not effectively match response details, and will cause semantic confusion, a Chinese question answering method based on multi-head attention and Bi-directional Long Short-Term Memory (BiLSTM) improved DAM model was proposed. This method can model longer multiple rounds of dialogue and handle the matching relationship between the response selection and the context. In addition, this paper uses the BiLSTM Network in the feature fusion process to improve the accuracy of multi-turn response selection tasks by capturing the time-dependent relation. In this paper, we test the improved DAM on two public multi-turn response selection datasets, the Douban Conversion Corpus and the E-commerce Dialogue Corpus. Experimental results show our model outperforms the baseline model by 1.5% in Recall-10-at-1 with the word vector enhancement.

Keywords: multi-turn response selection ,Deep Attention Matching ,multi-head attention ,Bi-directional Long Short-Term Memory

1 引言

人机对话系统是一个复杂的研究方向，构建人机对话系统的方法之一是检索式方法。检索式方法首先需提取输入对话的特征，随后在候选回复库匹配多个目标候选回复，按照某种指标进行排序，输出得分最高的回复。将之前的对话输出作为历史对话，即形成多轮对话的形式。

近年来，随着深度学习的发展，关于人机对话的研究重点逐渐由基于模板、规则的传统方法转变为基于端到端的深度学习模型方法。（Wu Y et al.,2016）提出序列匹配网络（Sequence Matching Network，SMN）模型，模型可分为“表示—匹配—融合”三个部分，整体上基于 CNN 和 RNN 实现以语义融合为中心的多轮对话回复选择。（Zhang Z et al.,2018）提出深度表达融合（Deep Utterance Aggregation，DUA）模型，针对 SMN 模型将历史对话直接拼接为上下文存在噪声和冗余的问题，采用注意力机制挖掘关键信息并忽略冗余信息，最终获得对话表达和候选响应的匹配得分。（Zhou X et al.,2018）提出深度注意力匹配（Deep Attention Matching，DAM）模型。在 SMN 模型的基础上，省去 CNN 和 RNN 等结构，仅依靠注意力机制完成多轮对话的回复选择，使模型参数大幅度减少，大幅提升了训练速度。而这类方法的局限性在于候选集中被选定的回复仅适用于本轮对话，与上下文并不能形成良好的匹配，或在匹配模型中没有学习到真正的语义关系，对多轮对话的内容产生了混淆，难以选择正确的候选回复。

在“表示—匹配—融合”这一框架下，同时优化三个部分是现阶段的研究难点。在前人研究的基础上，本文对比 DAM 模型，通过引入多头注意力机制，使模型更适合处理含有细微变化的数据，能让选定的目标候选回复与上下文形成良好的匹配关系。此外，本文在特征融合过程中采用 BiLSTM 模型，通过捕获多轮对话中的序列依赖关系，帮助模型建立每轮对话与前一轮对话、候选回复之间的匹配信息，使匹配模型学习到真正的语义关系，进一步提高选择目标候选回复的准确率，基于此建立基于多头注意力和 BiLSTM 改进的 DAM 模型 Ex-DAM，在豆瓣、电商这两个多轮中文对话数据集上进行研究。

论文的结构安排如下：第 1 节介绍了“检索式人机多轮对话”的概念及特点，概述了近几年深度学习模型方法；第 2 节介绍了深度注意力机制模型 DAM 的整体结构；第 3 节介绍基于多头注意力和 BiLSTM 改进的 DAM 模型 Ex-DAM，主要包括多头注意力模块、语义表示网络和双通道 BiLSTM 特征融合网络；第 4 节介绍实验数据、实验内容、实验结果及分析，验证 Ex-DAM 模型的有效性；最后在第 5 节进行总结。

2 DAM 模型

（Zhou X et al.,2018）提出的 DAM 模型的整体结构如图 1 所示，可以分为输入、表示、匹配、聚合四个部分。模型的输入是多轮对话和候选回复，输出是每个候选回复的得分。

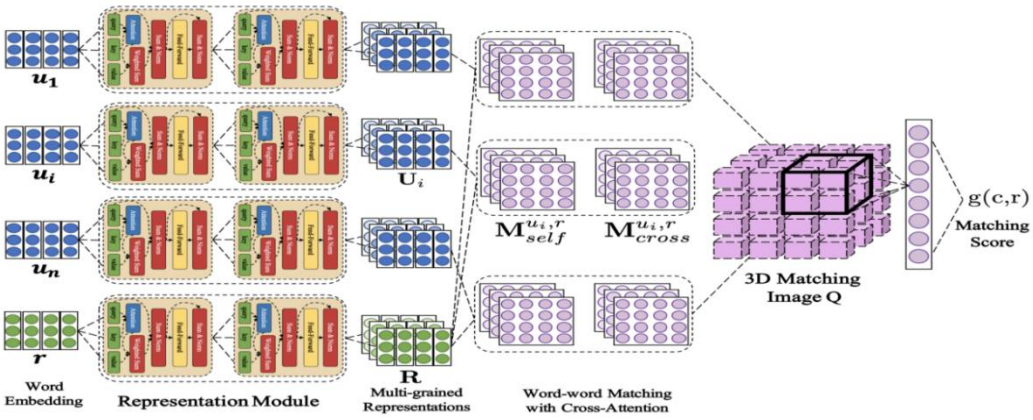


图 1. DAM 模型的整体结构

DAM 模型的注意力模块含有查询向量 Q、键向量 K 和值向量 V 三个输入。模块首先利用以下公式计算输入的缩放点积注意力：

$$\mathrm{V}_{\mathrm{att}} = Attention(Q,K,V) \tag{1}$$

之后模块将 V_{att} 和 Q 直接相加，产生的和包含二者的联合语义信息。为防止梯度消失或梯度爆炸，对 V_{att} 和 Q 相加的结果应用层归一化，结果记为 V_{ln}。接着将 V_{ln} 传入一个基于 ReLU 函数激活的双层前馈网络 FFN，进一步处理融合信息。FFN 的输出与输入进行一次残差连接，产生的结果再次应用层归一化，此时的 O 是整个注意力模块计算过程的最终输出，将其表示为：

$$O = AttentionModule(Q,K,V) \tag{2}$$

DAM 模型的语义表示网络由多个相同的注意力模块首尾相连，形成堆叠的网络结构。网络中每个注意力模块的三个输入相同，计算自注意力，公式表示为：

$$U_{\mathrm{i}}^{l+1} = AttentionModule(U_i^l,U_i^l,U_i^l) \tag{3}$$

$$R^{l+1} = AttentionModule(R^l,R^l,R^l) \tag{4}$$

其中 l 的范围从 0 到 $L-1$，L 表示模块堆叠的数量，$U_i^0 = u_i$ 以及 $R^0 = r$ 是原始输入。

DAM 模型中特征匹配的输入是语义表示网络的输出 U_i 和 R。针对不同粒度 l，网络将产生两种匹配矩阵，一种是自匹配矩阵 $M_{self}^{u_i,r,l}$，另一种是互匹配矩阵 $M_{cross}^{u_i,r,l}$。$M_{self}^{u_i,r,l}$ 是 U_i 和 R 的点积，矩阵中含有 U_i 和 R 元素间的语义依赖关系。$M_{cross}^{u_i,r,l}$ 的产生基于对注意力模块输入的修改，通过令 U_i 和 R 中对应元素互相关注，构造出新的语义表示 $\widetilde{U}_i^l$ 和 $\widetilde{R}^l$，用于捕捉跨越对话表达与候选回复之间的交叉关联特征。二者的计算公式为：

$$\widetilde{U}_{\mathrm{i}}^l = AttentionModule(U_i^l,R^l,R^l) \tag{5}$$

$$\widetilde{R}^1 = AttentionModule(R^l,U_i^l,U_i^l) \tag{6}$$

通过计算 $\widetilde{U}_i^l$ 和 $\widetilde{R}^l$ 的点积，得到 $M_{cross}^{u_i,r,l}$，DAM 模型中的特征融合网络将多个粒度的 $M_{self}^{u_i,r,l}$ 和 $M_{cross}^{u_i,r,l}$ 作为输入，在 i 和 l 两个维度拼接两个矩阵，得到矩阵 $P_{i,l}$ 如下：

$$P_{\mathrm{i,l}} = M_{self}^{u_i,r,l} \oplus M_{cross}^{u_i,r,l} \tag{7}$$

在 DAM 模型中，$P_{i,l}$ 被称为像素点，由 $P_{i,l}$ 组合形成的高维矩阵 P 被称为图像，这样的命名是为了方便使用 CNN。P 中的图像深度对应于多轮对话的轮次，图像宽度对应于每轮对话和候选回复在句子层级的匹配信息，图像高度对应于每轮对话和候选回复在单词层级的匹配信息。由于 P 含有三个维度的特征，DAM 模型采用 3D 卷积进行特征提取。经过两次 3D 卷积和最大池化，P 最终变成一维特征 f，再经过一个线性分类器即可获得匹配分数 $g(c,r)$。

3 基于多头注意力机制和 BiLSTM 网络的 Ex-DAM 模型

3.1 基于多头注意力和 BiLSTM 的 Ex-DAM 模型

为使 DAM 模型更适合处理含有细微变化的数据，进一步提高选择目标候选回复的准确率，本文利用多头注意力表示网络和双通道特征融合网络，结合 DAM 模型中的交互匹配网络，基于此构成一个新的端到端检索式多轮对话系统模型，将该模型命名为基于多头注意力和 BiLSTM 的 Ex-DAM 模型。模型的整体结构如图 2 所示。

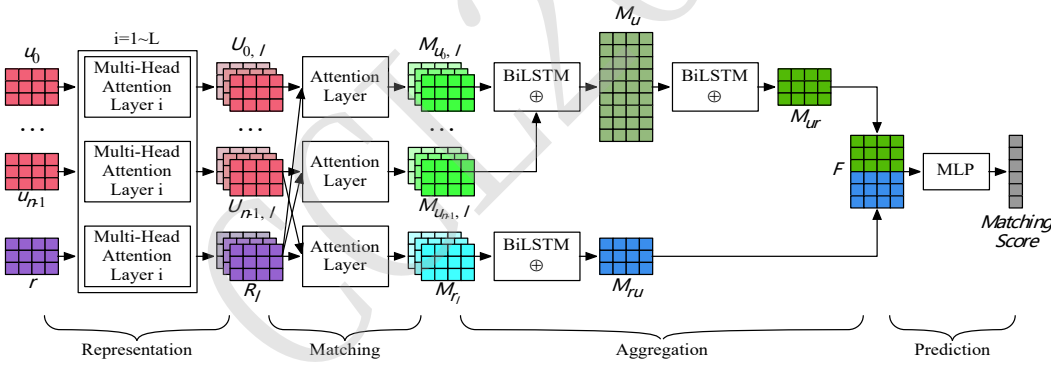


图 2. Ex-DAM 模型整体结构

模型的输入是词向量形式的多轮对话和候选回复，首先经过 L 个多头注意力层以获取它们的多粒度表示。在这些表示向量中，每一轮对话都和候选回复进行一次普通的注意力计算，得到多个主匹配矩阵。此外，候选回复再额外地与最后一轮对话计算一次注意力，以获得次匹配矩阵。随后，主、次匹配矩阵分别作为两个通道的输入进行特征融合。在这个过程中，所有的匹配矩阵经过 BiLSTM 和拼接操作依次进行序列特征提取和维度统一。最后，把两个通道的输出向量首尾拼接，经过多层感知器就能获得每个候选回复与多轮对话之间的匹配分数。

3.2 Ex-DAM 模型中的多头注意力模块和语义表示网络

普通的注意力机制在文本序列中可以很好地提取词向量角度的关键信息，但几乎无法识别对词向量进行统一修改的操作。多头注意力机制正好可以解决此类问题，在计算时首先输入多次映射，每个映射使用不同参数进行相同计算，最后将各个输出合并在一起，因此比缩放点积注意力更适合处理含有细微变化的数据，本文使用多头注意力模块结构如图 3 所示。

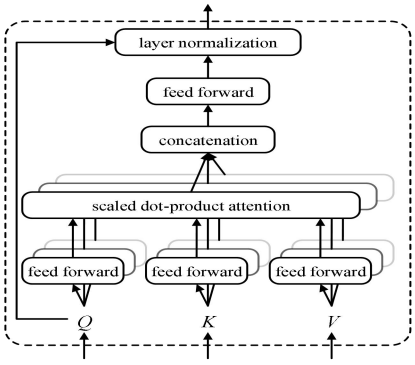


图 3. 多头注意力模块

多头注意力模块含有查询向量 Q、键向量 K 和值向量 V 三个输入，整体计算公式为：

$$MHAModule(Q,K,V) = LayerNorm(Q + MultiHead(Q,K,V)) \quad (8)$$

本文在模块中应用了残差连接，将输入 Q 与前馈层的输出恒等叠加，不会引入额外的参数，也不会增加模型的计算复杂度，在叠加过程中可强化输入中的重点内容，提升训练效果。多头注意力的头部能在不同子空间处理同一序列，从而获得更丰富的语义表示信息。由多头注意力模块组成 Ex-DAM 模型的语义表示网络，结构如图 2 中的 Representation 模块所示。

输入 $[u_i]_{i=1}^{i=T}$ 和 $[r_i]_{i=1}^{2\le i\le 10}$ 分别是每轮对话和候选回复的词向量，这些词向量已经过增强处理，同时含有意图信息和语义信息。词向量进入多头注意力模块计算语义表示的公式为：

$$U_{\mathrm{i}}^{l+1} = MHAModule(U_i^l, U_i^l, U_i^l) \quad (9)$$

$$R^{l+1} = MHAModule(R^l, R^l, R^l) \quad (10)$$

以 $U_i^0 = u_i$ 和 $R^0 = r$ 为多头注意力模块的原始输入，经 $L-1$ 个堆叠的相同模块可逐层获得多粒度的语义表示，即 $U_i = [U_i^0, ..., U_i^L]$ 和 $R = [R_i^0, ..., R_i^L]$。由于后续处理的计算量较大，为避免内存溢出需控制输入数量，本文针对 U_i 和 R 中的元素选择制定了以下三个处理策略：

第一个策略是当 L 数值较小时，保留所有 U_i 和 R，即使用所有粒度的语义表示作为特征匹配网络的输入，记作 Ex-DAM$_{\mathrm{L}}$；第二个策略是当 L 数值较大时，保留 U_i 和 R 中的后 m 个元素，即仅使用深层粒度的语义表示作为特征匹配网络的输入，记作 Ex-DAM$_{\mathrm{L\text{-}m}}$；第三个策略是当 L 数值较大时，保留 U_i 和 R 中的第一个元素和后 m 个元素，将原始输入同时作为语义表示网络和特征匹配网络的输入，而后者的输入还包含原始输入的多粒度语义表示，记作 Ex-DAM$_{\mathrm{L\text{-}0\text{-}m}}$。

3.3 Ex-DAM 模型中的双通道 BiLSTM 特征融合网络

本文提出的 Ex-DAM 模型的特征匹配网络仿照 DAM 模型，以语义表示网络的输出 U_i 和 R 作为输入，利用 DAM 模型中的注意力模块构造自匹配矩阵 $M_{self}^{u_i,r,l}$ 和互匹配矩阵 $M_{cross}^{u_i,r,l}$，如图 2

中的 Matching 模块，由于本文在该网络中未做改动，这里不做赘述。在 DAM 模型的特征融合网络中，3D 卷积作为注意力机制的一种辅助策略，在进一步提取特征的同时缩小数据维度。由于本文已将缩放点积注意力替换为多头注意力，因此本文在 Ex-DAM 模型中也将 3D 卷积进行了替换，Ex-DAM 模型的特征融合网络如图 2 中 Aggregation 模块所示。

在 Ex-DAM 模型的特征融合网络中，文本首先对 $M_{self}^{u_i,r,l}$ 和 $M_{cross}^{u_i,r,l}$ 进行加权求和，提取匹配矩阵中的重要匹配特征，其中 w_l 是共享权重，作用是增强模型的泛化性并减少计算开销。计算得到的 M_{self}^{i} 含有每轮对话与前一轮对话之间的匹配依赖信息，而 M_{cross}^{i} 中含有每轮对话与候选回复之间的匹配依赖信息。

接下来，本文利用两个不同的 BiLSTM 网络分别处理 M_{self}^{i} 和 M_{cross}^{i}，提取其中细微片段之间的匹配关系。与原始 DAM 模型中使用的 3D 卷积不同，Ex-DAM 模型放弃从“轮次—对话—候选回复”的角度入手，转而考虑以轮次为单一主线，分别对“对话—对话”以及“对话—候选回复”进行独立计算。同时，由于将在实验中引入基于意图识别的词向量增强，以累加形式嵌入到对话中的意图嵌入向量也较容易引起 BiLSTM 的关注。将代表不同轮次的隐藏状态矩阵首尾拼接，便可得到两种不同的融合特征矩阵——M_{self}^{agr} 和 M_{cross}^{agr}。特征融合网络的末尾部分与之前的相关研究保持一致，采用全连接神经网络处理 M_{self}^{agr} 和 M_{cross}^{agr}，将其中蕴含的融合特征表示为一组匹配分数，并使用 softmax 函数进行归一化处理，形成匹配概率。

4 数据集和实验设置

4.1 数据集与数据预处理

本文使用的中文多轮对话数据集是（Wu Y et al.,2016）提供的豆瓣对话数据集和（Zhang Z et al.,2018）提供的电商对话数据集。这两个数据集已由提供者进行了中文分词处理，每个数据集含有六个文件，其中 responses.txt 用于索引特定候选回复，word2vec.txt 用于预训练词向量，vocab.txt 用于索引特定单词，test.txt 作为测试集，train.txt 作为训练集，valid.txt 作为验证集。本文针对两个数据集中的数据组成进行了统计，统计结果如表 1 所示。

表 1. 数据集统计结果

统计指标	豆瓣对话数据集			电商对话数据集		
	训练集	验证集	测试集	训练集	验证集	测试集
多轮对话总数	495389	25000	1000	386478	5003	1000
候选回复总数	1000000	50000	10000	1000000	10006	10000
多轮对话轮次数最小值	3	3	3	1	1	1
多轮对话轮次数最大值	98	91	45	119	111	49
一轮对话单词数最小值	1	1	1	1	1	1
一轮对话单词数最大值	10624	5617	862	207	94	100

数据预处理的第一部分是数据清洗，利用 response.txt 中的候选回复表，可索引 i^T 和 i^F 得到

2~10 个候选回复，记作 $R=[r_i]_{i=1}^{2\le i\le 10}$。统计 R 中每个候选回复 r_i 的单词数 n_i^r，若满足

$$n_i^r \le W_{\max} \tag{11}$$

表示 r_i 中的单词数符合常规，其中 $W_{\max}$ 是本文自行设置的单词处理最大值，本文设为 100。本文利用 C 中的<EOS>将 C 分割成多个单轮对话，同时根据<EOS>的个数快速统计 C 的轮次数，得到 $C=[c_i]_{i=1}^{i=t}$，其中 t 是轮次数，若满足

$$t \le T_{\max} \tag{12}$$

表示 C 的轮次数符合常规，其中 $T_{\max}$ 是本文自行设置的轮次处理最大值。此外，还需针对满足条件的 C 统计其中 c_i 的单词数 n_i^c，若满足

$$W_{\min} \le n_i^c \le W_{\max} \tag{13}$$

表示 n_i^c 中的单词数符合常规，其中 $[W_{\min},W_{\max}]$ 是单词处理阈值区间，$W_{\max}$ 与本文对 n_i^r 的限制相同，$W_{\min}$ 是特别针对 c_i 设置的单词处理最小值，本文设为 2。

随后，本文将清洗后得到的数据进行数据规范。针对候选回复 $R=[r_i]_{i=1}^{2\le i\le 10}$，本文在 $[W_{\min},W_{\max}]$ 区间内设置一个表达长度值 W，通过处理所有 r_i，使其满足

$$n_i^r = W \tag{14}$$

若 $n_i^r < W$，将若干特殊标识符<PAD>增加至 r_i 尾部，使其长度达标。<PAD>本身不具备语义，因此也不会影响到 r_i 的语义。若 $n_i^r > W$，删除超出部分的单词。

经过上述处理，每个多轮对话都由 T 轮对话组成，每轮对话和每个候选回复都由 W 个单词组成。借助 vocab.txt 中单词与词序之间的对应关系表，本文将数据中的所有单词（包括<EOS>）转换为数字，实现文本数据的向量化过程。

4.2 基于意图识别的词向量增强

得到上述规范数据，其中将单词转化成的数字同时与 word2vec.txt 文件中的预训练词向量一一对应，根据这种对应关系可将每个数字转换成 200 维的词向量，其中<PAD>的词向量是 200 维的零向量。r_i 和 c_i 均是维度为 $(W,200)$ 的向量，C 的维度是 $(T,W,200)$。

受位置编码的启发，本文引入意图嵌入向量以处理意图识别结果。针对候选回复集合 $R=[r_i]_{i=1}^{2\le i\le 10}$，取其正确候选回复集合 R^T，对含有某种意图的 C 和 R^T 同时采用如下的编码方式：

$$ID(a)=\frac{1}{10}\sin(1/10000^{1-\frac{d}{dm}}) \tag{15}$$

$$ID(b)=\frac{1}{10}\sin(1/10000^{\frac{d}{d_m}}) \tag{16}$$

本文将 200 维的意图嵌入向量直接与相应的词向量相加，由<PAD>构成的单词和对话保持不变，得到的结果即为词向量增强的结果。这样设计的好处是，对于相同意图的 C 和 R^T，二者的意图嵌入向量完全相同，增强了二者的相关程度。此外，预训练词向量的数值范围是 0~1，而意图嵌入向量的数值范围是 0~0.1，二者直接相加的数据变化对预训练词向量影响很小。

4.3 实验设置与结果分析

4.3.1 评价指标及实验设置

本文采用检索式多轮对话系统常用的几种评价指标，用来衡量模型的性能。假设多轮对话数据集 C 由 N 个集合 c 组成，每个集合 c 包含正确回复 t 个、错误回复 f 个。在整个数据集上计算各项评价指标的平均值，得到平均精度均值（Mean Average Precision，MAP）、倒数排序均值（Mean Reciprocal Rank，MRR）、首位准确率（Precision-at-1，P@1）和计算召回率（Recall-n-at-k，$R_n@k$）：

$$MAP=\frac{1}{N}\sum_{c\in C}AP(c) \tag{17}$$

$$MRR=\frac{1}{N}\sum_{c\in C}RR(c) \tag{18}$$

$$P@1=\frac{1}{N}\sum_{c\in C}P@1(c) \tag{19}$$

$$R_n@k=\frac{1}{N}\sum_{c\in C}R_n@k(c) \tag{20}$$

其中 AP 为平均精度（Average Precision），RR 为倒数排序指数（Reciprocal Rank）。

本次实验中使用的数据均已经过数据预处理，训练过程相关配置使用 Adam 优化器调节模型参数，DAM 模型和 Ex-DAM 模型超参数的取值均如表 2 所示。

表 2.DAM(Ex-DAM)模型超参数表

超参数	参数含义	参数值
W	表达长度值	50
T	对话轮次值	9
epoch	数据集迭代次数	3
layer	堆叠多头注意力模块数	5
batch_size	单次训练样本数	128
learning_rate	初始学习率	0.001
decay_step	学习率衰减步长	500
decay_rate	学习率衰减率	0.9

4.3.2 DAM 模型实验结果

本文先在两个数据集的训练集上训练 DAM 模型，然后在测试集上评估模型的性能，实验结果如表 3 所示。

表 3. DAM 模型实验结果

实验内容	豆瓣对话数据集				电商对话数据集		
	MAP	MRR	P@1	R_{10}@1	R_{10}@1	R_{10}@2	R_{10}@5
DAM模型	0.550	0.601	0.427	0.254	0.406	0.547	0.810
+词向量增强	0.539	0.583	0.409	0.238	0.392	0.530	0.798
+数据预处理	**0.556**	**0.606**	**0.434**	**0.259**	**0.414**	**0.557**	**0.819**
+数据预处理+词向量增强	0.548	0.587	0.425	0.248	0.396	0.539	0.807

由表 3 可以看出，数据预处理有助于模型性能的提升，对原始数据集进行数据预处理，在豆瓣对话数据集上的各项评价指标获得了 0.5%~0.7%的提升，在电商对话数据集上的各项评价指标获得了 0.8%~1%的提升。还可以看出，将基于意图识别的词向量增强直接应用于 DAM 模型产生了不理想的效果，在两个数据集上的各项评价指标均下降了 1%以上，即使经过数据预处理，模型效果有了略微提升，但也始终低于基线水平。此结果的产生原因主要是 DAM 模型完全由自注意力机制构造而成，其中的计算过程依赖于词向量，本文在词向量层面进行的任何改动都将逐层干扰注意力机制的计算，从而导致 DAM 模型性能急剧下降。

4.3.3 Ex-DAM 模型实验结果

为了验证本文提出的 Ex-DAM 模型是否有效，将经过数据预处理的两种实验模型作为基线模型，从是否进行词向量增强的角度进行了独立实验。其中 Ex-DAM$_5$ 表示堆叠注意力模块数设置为 5，保留所有U_i和 R，即使用所有粒度的语义表示作为特征匹配网络的输入；Ex-DAM$_{5\text{-}4}$ 表示模块数设置为 5，保留U_i和 R 中的后 4 个元素；Ex-DAM$_{5\text{-}0\text{-}4}$ 表示模块数设置为 5，保留U_i和 R 中的第 1 个元素和后 4 个元素，实验结果分别如表 4 和表 5 所示：

表 4. 不含词向量增强的 Ex-DAM 模型实验结果

实验内容	豆瓣对话数据集				电商对话数据集		
	MAP	MRR	P@1	R_{10}@1	R_{10}@1	R_{10}@2	R_{10}@5
DAM模型	0.556	0.606	0.434	0.259	0.414	0.557	0.819
Ex-DAM$_5$	**0.562**	**0.610**	**0.441**	**0.264**	**0.423**	**0.563**	**0.822**
Ex-DAM$_{5\text{-}4}$	0.557	0.605	0.437	0.260	0.419	0.561	0.819
Ex-DAM$_{5\text{-}0\text{-}4}$	0.559	0.607	0.438	0.259	0.420	0.561	0.820

表 5. 含有词向量增强的 Ex-DAM 模型实验结果

实验内容	豆瓣对话数据集				电商对话数据集		
	MAP	MRR	P@1	R_{10}@1	R_{10}@1	R_{10}@2	R_{10}@5
DAM模型	0.548	0.587	0.425	0.248	0.396	0.539	0.807
Ex-DAM$_5$	0.568	0.607	0.442	0.265	0.425	0.564	0.830
Ex-DAM$_{5\text{-}4}$	0.565	0.605	0.440	0.262	0.421	0.561	0.828
Ex-DAM$_{5\text{-}0\text{-}4}$	**0.570**	**0.615**	**0.448**	**0.270**	**0.427**	**0.566**	**0.831**

由表 4 和表 5 看出，无论是否对数据集进行词向量增强，Ex-DAM 模型的实际表现都优于基线模型。即使词向量增强曾在之前的实验中导致基线模型的性能不升反降，却帮助 Ex-DAM 模型达到了最佳性能，说明多头注意力机制与 BiLSTM 的共同作用要优于普通自注意力机制。

在表 4 中，Ex-DAM$_5$ 模型性能优于其余模型，该模型与其余模型的不同之处在于语义表示网络的输出含有 5 种粒度的语义表示。若将最底层粒度语义表示去除或以原始输入替换最底层语义表示，都将损失一部分模型性能。然而在表 5 中，以原始输入替换最底层语义表示的 Ex-DAM$_{5\text{-}0\text{-}4}$ 模型却比 Ex-DAM$_5$ 模型性能更优。这是由于对原始输入进行了词向量增强，导致原始输入含有额外的意图特征，而语义表示网络中每个粒度的语义表示都源于原始输入，相当于在计算过程中不断强化这种意图特征，促使模型重点对意图特征建模。

本文进行的实验均将堆叠注意力模块数设置为 5，为探究 Ex-DAM$_{L\text{-}0\text{-}m}$ 模型中 L 和 m 的取值对模型性能的影响，本文使用经数据预处理和词向量增强的电商对话数据集进行了额外的实验，将评价指标 $R_{10}@1$ 结果绘制成折线图 4 所示。

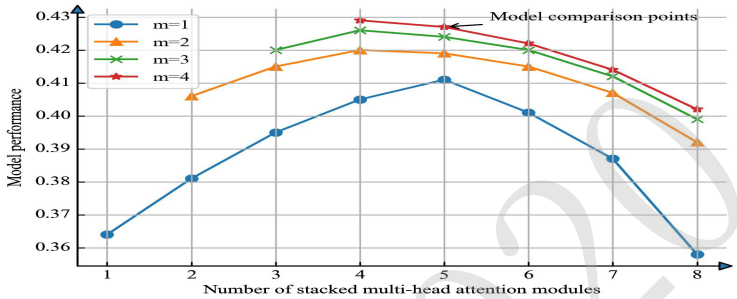


图 4. 不同参数搭配对 Ex-DAM$_{L\text{-}0\text{-}m}$ 模型的影响

本文在实验中始终保持语义表示网络的输出粒度不超过 5，这是由于模型占用的显存所致，若超出此值必须修改超参数，而计算方面的消耗将呈指数级增长，很难与之前的实验做对比。图 4 标记的模型对比点是上一实验中的 Ex-DAM$_{5\text{-}0\text{-}4}$ 模型，由图可知，多粒度语义表示确实能在一定程度上提升模型性能，当模型将 4 个堆叠多头注意力模块的输出与原始输入共同作为语义表示网络的输出时，通常能取得最高性能。若堆叠多头注意力模块数超过 5，无论如何选择 m 的值，模型都将逐渐出现过拟合现象，这是由于随着堆叠多头注意力模块数的增加，被多头注意力机制重点关注的信息会从前一层不断累加到下一层，导致这些信息在深层计算过程中基本保持不变，严重影响模型训练。

5 结论及后续工作

本文提出了基于多头注意力和 BiLSTM 改进的 DAM 模型 Ex-DAM，该模型用于处理中文多轮对话问答匹配问题。本文将 DAM 模型作为基线模型，利用多头注意力机制在多个不同子空间内计算特征，从而有能力建模较长的多轮对话。Ex-DAM 模型的卷积核使用 BiLSTM 来捕获序列上的依赖关系。实验证明，Ex-DAM 模型性能在电商和豆瓣的数据集上均优于基线模型。

在未来的研究中，将尝试加入命名实体识别、情感分析等多种辅助手段，使得 Ex-DAM 模型可以在文本片段中尽可能提取更多的特征。

致谢

*通信作者（chongzhy@vip.sina.com），本文承国家教育部人文社会科学研究规划基金资助项目（16YJAZH072）、国家社会科学基金重大项目（14ZDB156）、食品安全大数据技术北京市重点实验室资助。

参考文献

Ba J L, Kiros J R, Hinton G E. Layer normalization[J]. arXiv preprint arXiv:1607.06450, 2016.

Baeza-Yates R, Ribeiro-Neto B. Modern information retrieval[M]. New York: ACM press, 1999.

陈晨, 朱晴晴, 严睿, 等. 基于深度学习的开放领域对话系统研究综述[J]. 计算机学报, 2019, 42(7): 1439-1466.

Glorot X, Bordes A, Bengio Y, et al. Deep Sparse Rectifier Neural Networks[C]. international conference on artificial intelligence and statistics, 2011: 315-323.

He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C]. computer vision and pattern recognition, 2016: 770-778.

Kingma D P, Ba J. Adam: A Method for Stochastic Optimization[J]. Computer ence, 2014

Tran D, Bourdev L, Fergus R, et al. Learning Spatiotemporal Features with 3D Convolutional Networks[C]. international conference on computer vision, 2015: 4489-4497.

Voorhees E M. The TREC-8 question answering track report[C]//Trec. 1999, 99: 77-82.

Wu Y, Wu W, Xing C, et al. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots[J]. arXiv preprint arXiv:1612.01627, 2016.

Zhang Z, Li J, Zhu P, et al. Modeling multi-turn conversation with deep utterance aggregation[J]. arXiv preprint arXiv:1806.09102, 2018.

Zhou X, Li L, Dong D, et al. Multi-turn response selection for chatbots with deep attention matching network[C]. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018: 1118-1127.

左彬靖. 基于 word2vec 和自注意力机制的文本分类研究[D]. 广东工业大学, 2019.

基于Graph Transformer的知识库问题生成

胡月
华中师范大学
计算机学院
湖北　武汉
huyue2017@mails.ccnu.edu.cn

周光有
华中师范大学
计算机学院
湖北　武汉
gyzhou@mail.ccnu.edu.cn

摘要

知识库问答依靠知识库推断答案需大量带标注信息的问答对，但构建大规模且精准的数据集不仅代价昂贵，还受领域等因素限制。为缓解数据标注问题，面向知识库的问题生成任务引起了研究者关注，该任务是利用知识库三元组自动生成问题。现有方法仅由一个三元组生成的问题简短且缺乏多样性。为生成信息量丰富且多样化的问题，本文采用Graph Transformer和BERT两个编码层来加强三元组多粒度语义表征以获取背景信息。在SimpleQuestions上的实验结果证明了该方法有效性。

关键词： 问题生成；知识库；语义表征；知识库问答

Question Generation from Knowledge Base with Graph Transformer

Yue Hu
Central China Normal University
School of Computer
Wu Han　Hu Bei
huyue2017@mails.ccnu.edu.cn

Guangyou Zhou
Central China Normal University
School of Computer
Wu Han　Hu Bei
gyzhou@mail.ccnu.edu.cn

Abstract

Knowledge base question answering requires a large number of question answering pairs when relying on the knowledge base to infer answers. However, building a large-scale and accurate data set is not only expensive, but also limited by factors such as domain. To alleviate the problem of data labeling, the question generation from knowledge base has attracted the attention of researchers. This task is to use the triples of knowledge base to automatically generate the questions. However, existing methods only use a triple to generate questions that are short and lack diversity. To generate questions with rich and diverse information, this paper use two encoding layers, Graph Transformer and BERT, to enhance the multi-granular semantic representation of triples to obtain background information. Experimental results on the SimpleQuestions dataset prove the effectiveness of the method.

Keywords: Question generation , Knowledge base , Semantic representation , Knowledge base question answering

1 引言

问答(Question Answer，QA)系统利用信息检索，在海量的非结构化或者结构化的数据中推断自然语言问句的答案。问答任务是人工智能的核心研究之一，就目前所研究的QA而言，大

部分的训练数据都是以QA对作为标记数据，例如基于知识库的SimpleQuestions(Bordes et al., 2015)数据集，基于文本的Wiki QA(Vrandecic and Krötzsch, 2014)数据集，以及阅读理解所用的SQuAD(Rajpurkar et al., 2016)数据集等。这些数据集受大小等限制，且标注花费昂贵。并且随着问答系统、阅读理解等领域的发展，对标注数据的需求愈加强烈，因此，本文探索的问题生成(Question Generation，QG)任务可以为QA系统的研究提供一个扩充数据集途径。

当下，QG任务开始逐渐作为一支有力的研究方向。给定一篇包含多个事实/知识点的文本和标准答案，传统的QG任务根据答案与这些事实逆向生成内容丰富、角度多元化的问句。随着深度学习研究的推入，QG任务也开始具有了多样性，给定一些数据，例如结构化的知识图谱或SQL，半结构化的表格，甚至无结构化的文本等，QG系统都能根据输入的数据自动生成自然语言的问题。问题生成虽然不是新兴任务且其研究领域较为小众，但随着图像处理、自然语言处理等各领域研究的相互影响，使得研究者们开始将注意力放在了QG任务上。基于知识库的QA系统很难理解自然语言问题，依靠目前常用的知识图谱和数据库也不一定可以回答出该问题。由于QG系统可以从已有的知识库生成QA系统需要的标准“问题-答案”对数据，故当出现一个新问题时，QG系统生成的问句与已有的“问题-答案”对进行相似计算向用户推荐知识库内已存在的相接近的问题，间接解决问答的问题。并且，由于能够与QA任务共用数据集，如给定知识库时，生成的问句可以为基于知识库的问答系统提供更大的训练数据集，与基于知识库QA任务的可成为对偶任务。传统的QG系统使用人工设定的模板或规则来生成问答对解决特定领域的QA任务。本文着重于结构化知识库上的QG任务。传统的问题生成方法，多采取模板的方式。例如，Duma和Klein(2013)使用关系定义的模板来生成简短的描述，并相应地替换主语和宾语实体的占位符标记。另外Seyler等人(2015)从知识库三元组中生成问句，实体与谓词的表达由它们在知识库和给定词典中所存在标签决定。Seyler等人(2017)参考基于模板的方法来描述结构化查询并生成自然语言问题。然而，这种基于规则的方法无法识别单词的语义内容，且可扩展性较弱。

如今随着深度学习的热潮，尤其是随着Sequence-to-Sequence（Seq2Seq）框架(Sutskever et al., 2014)和编码-解码结构(Cho et al., 2014)为自然语言生成(张建华和陈家骏, 2006)任务带来了新的研究方向，并在机器翻译(Luong et al., 2015)、智能对话(贾熹滨等人, 2017)等任务获得优异的成果，一系列以基于神经网络的QG系统被提出。Serban等人(2016)采用基于注意力的编码-解码框架在SimpleQuestions数据集上训练模型，并生成了30M的标签数据。Khapra等人(2017)将给定知识库中的所有实体转换为一组关键字，然后以Seq2Seq结构进行建模。ElSahar等人(2018)采用Zero-shot使模型能够泛化未遇见过的谓词和实体的问题。Wang等人(2018)在Seq2Seq基础上加入复制机制(Gu et al., 2016)，使模型能解决OOV问题。上述生成模型只考虑到对未出现过的三元组或低频出现的生僻词的解决方法，但对于生成信息量丰富和多样化的问题的考虑有所欠缺。受Cai(2019)和Koncel-Kedziorski(2019)等人工作的启发，本文着重于加强对三元组的多粒度语义特征表示，采用双编码层：基于Graph Transformer的图编码层和基于BERT(Devlin et al., 2019)加强的词级编码层。本文预先将知识库中实体、关系构成知识图，赋予实体全局化的语义向量，然后针对该三元组，结合Transformer结构(Vaswani et al., 2017)的并行性对输入节点进行特征细化。同时，为了充分利用词语粒度的语义向量，三元组的词语序列先通过BERT预训练模型获取向量表征，再使用双向门控循环单元(Gated Recurrent Unit，GRU)(Cho et al., 2014)网络计算上下文向量。最后，本文将两个编码层联合获得更完善的三元组特征表示，再输入解码层生成问句。本文在英文数据集SimpleQuestions上进行了实验，实验的评测结果表明了该方法的有效性。

综上，本文的贡献如下：（1）基于知识库的QG模型，本文率先提出使用基于Graph Transformer的方法为实体获取丰富的背景信息。（2）本文为完善三元组的特征表示采用图和词级的表示，结合知识图和BERT分别进行初始化，再用神经网络对其泛化。（3）自动评测和人工评估的结果表明，本文模型生成的问题信息量更丰富且表达形式多样化。

2 模型设计

为了解决知识库中单个三元组存在背景信息量少，语义表达不够完善等问题，受Koncel-Kedziorski(2019)和Cai(2019)等人的工作启发，本文采用具有图神经网络特征的Transformer结构作为编码层，结合了知识图谱对实体和关系进行表示，以图的形式作为输入，并且还使用BERT预训练模型获得词语的语义表示。与Seq2Seq模型结合，从而针对任务目标规范适合本

数据集的语义向量表示，以获得更准确的潜在语义，使得模型生成的问题更加丰富流畅。

2.1 模型结构

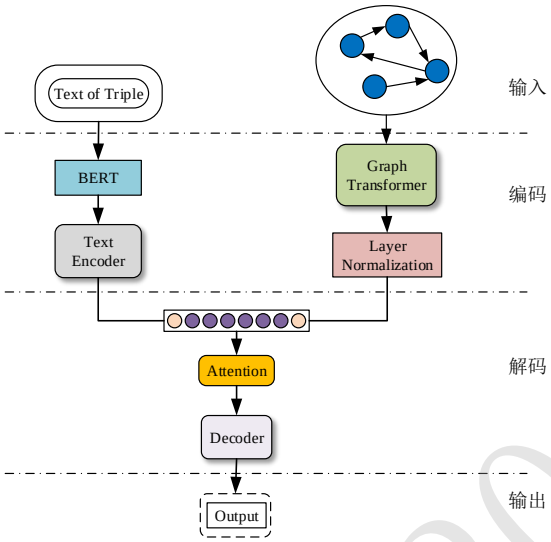


图 1. 模型结构图，双编码层在解码端连接

本节将介绍基于Graph Transformer(GT)的知识库问题生成模型（简称GT-KBQG）。GT-KBQG模型的结构如图1所示，由三个模块组成，分别为GT编码器，基于BERT预处理的文本编码器，以及基于Attention机制的GRU解码器。具体来看，首先三元组进行构图预处理，之后使用GT编码器对三元组在图中的向量表示$e_{triple} = (e_s, p_r, e_o)$进行编码，同时以自然语言文本形式表示的三元组$w = \{w_1, \ldots, w_i, \ldots, w_m\}$通过BERT预处理转换为向量形式$x_w = \{x_{w_1}, \ldots, x_{w_i}, \ldots, x_{w_m}\}$，通过双向GRU编码器编码。然后将以上两个编码层结果进行结合作为解码器的隐藏层初始化状态，解码器结合Attention机制(Luong et al., 2015)获得上下文向量，以此加强问题的生成效果。

2.2 基于Graph Transformer的图编码层

为了让实体之间更加紧密和结构化，有学者将其构成图的形式（实体为结点，关系为边），即知识图谱。本文利用已构成的知识图赋予数据集中的三元组丰富的背景网，结合Transformer结构，在丰富的知识背景网中抓取对本数据集针对任务有利的语义信息，使实体之间获得更紧密的关系，单个实体的语义向量也更加丰富。

2.2.1 知识构图预处理

英文知识库FreeBase(Bollacker et al., 2007)的实体原始表示为独立的id形式，且FreeBase主要由社区成员收集整理，数量庞大，常被视为可靠的知识图谱。SimpleQuestions数据集基于FreeBase得到，故其原始实体也是以id形式，为了模型能够获取更多的实体背景信息，本文将以大规模知识库预先训练获得的知识图作为SimpleQuestions数据集中实体的信息网背景图。与词嵌入的思想一致，本节主要是获得知识图嵌入（Knowledge Graph Embedding，KG-Emb）(Bordes et al., 2011)。

KG-Emb是为了将高维的知识图表示为低维谓词、实体的表示（P 和E）。为实现这一目标，Bordes等人(2011)提出基于翻译模型的TransE，通过训练P和E两个矩阵，使所有事实(s, r, o)的总距离$\sum ||e_s + p_r - e_o||_2^2$最小。在TransE的推动下，探索了一系列基于翻译的模型。例如，由Wang等人(2014)提出的TransH处理一对多或者多对一关系，与TransE直接测量e_s和e_o之间的距离不同，TransH将其投影到谓词特定的超平面中，在一定程度上解决了TransE不善于处理复杂关系的情况，使同一个实体在不同关系下的表示不同。TransH预设实体与关系处于相同的语义空间，Lin等人(2017)提出的TransR则将关系处于不同的语义空间的假设，该方法为每个谓词r定义一个转换矩阵M_r，以最小化$\sum ||e_s M_r + p_r - e_o M_r||_2^2$ 为目标。基

于TransE进行改进的类似算法还有很多，比如PTransE 等(Lin et al., 2015)多种图嵌入表示方法。

本文选用TransE方法对三元组事实进行图的预处理，将实体作为整体转为向量的表达。选择TrandE主要有以下几个考虑：首先，与它的改善结构如TransH、TransR等相比，TransE的参数较少；其次，TransE的主要层次关系的表示非常有效；最后，所有关系模型在多关系的实验分析中，TransE的效果良好。本文直接采用了Huang等人(2019)在FB2M数据集上所提供的基于TransE的KG-Emb。

2.2.2 Graph Transformer

本文使用预处理后的知识图作为KG-Emb，本编码层的输入是具有全局语义信息的实体和关系，并基于Transformer结构，在全局图的背景语义下进一步捕捉主语实体、关系、宾语实体之间的关联，使作为输入的三元组具有更加符合本任务的语义粒度表示。Vaswani等人(2017)提出的Transformer结构，通过全局上下文建模的多头自注意力机制来实现高效且并行的计算，具有并行性的优点，解决了RNN的在长序列上的顺序计算结构的缺点。

GT作为图编码器，参考Transformer编码层结构，由几个相同的网络块组成，根据其并行计算特性，输入节点之间可直接进行信息传递，如图2表示的为单个网络块，左边部分的e_o、e_s和p_r分别表示从KG-Emb中获取的主语实体、宾语实体以及谓词的语义向量，通过Graph Transformer结构计算抓取语义信息，使实体向量更加丰富和符合本文任务。

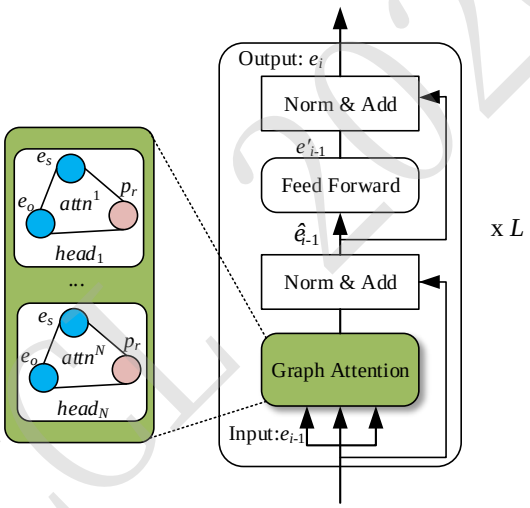


图 2. Graph Transformer结构

本文使用N头注意力机制在图2所示的网络块中堆叠L次，在输入残差网络之前进行拼接。Graph Transformer的具体计算如公式(1)所示：

$$\hat{e}_i = e_i + ||_{j=1}^{N} attn^j(q_i, k_i)v_i \tag{1}$$

在公式(1)中，其中||表示N个$attn$的连接操作($j \in N$)，$attn^j$为点积计算，如公式(2)所示，为了降低点积造成的阻碍梯度流的趋势，参照Vaswani等人(2017)的方法，将结果缩小$\sqrt{d_k}$之后再利用$softmax$函数进行归一化。

$$attn^j(q_i, k_i) = softmax(\frac{q_i k_i^\top}{\sqrt{d_k}}) \tag{2}$$

公式(1)和公式(2)中出现的q_i，k_i，v_i是第$i(i<L)$个堆叠块对输入进行线性变换后的d_k维的向量表示，如公式(3)所示。

$$\begin{cases} q_i &= e_i W_i^q \\ k_i &= e_i W_i^k \\ v_i &= e_i W_i^v \end{cases} \tag{3}$$

其中e_i由第$i-1$层第二个LN层标准化计算得来，即为第i层的输入,其中LN为标准化层。通过块状网络扩展多头注意力层，如图2所示，每个块状网络都具有衔接层转换，如公式(4)和(5)所示:

$$e_i^{'} = FF(LN(\hat{e}_i)) \tag{4}$$

$$e_{i+1} = LN(e_i^{'} + LN(\hat{e}_i)) \tag{5}$$

如公式(5)所示，第i层的最终输出作为第$i+1(\leq L)$层的输入。公式(4)中，$FF(\cdot)$为两层前馈网络，两层之间具有非线性变换，本文选取$ReLU$非线性变换以及残差网络的前馈进行计算。通过知识库构图之后，每个实体和关系拥有对应的向量化表示，实体与关系的拼接作为$i=1$时的初始输入，如公式(6)所示。

$$e_1 = Concat(e_s; p_r; e_o) \tag{6}$$

图1中的Layer Normalization层，对最终的输出结果e_N实行层标准化如公式(7)，与词级编码层进行融合。

$$e_N = LN_{output}(e_N) \tag{7}$$

多个块状网络的堆叠使得信息可以从图进行传播。从GT编码层输出的最终结果与下一节的文字编码结果进行组合作为解码器端的输入。

2.3 基于BERT增强词级表示的编码层

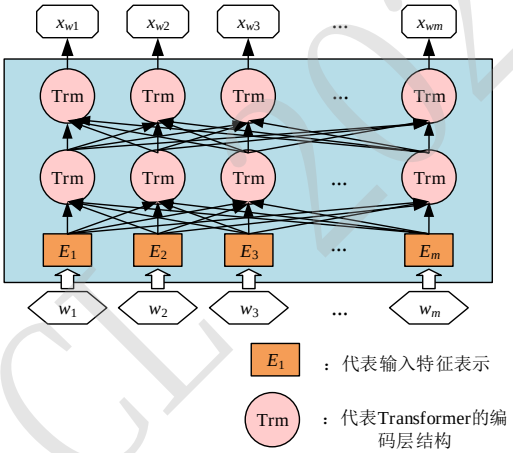


图 3. BERT预处理结构图

在2018年，Jacob Devlin等人(Devlin et al., 2019)提出BERT预训练模型在NLP领域获得极大的反响，在预训练和精调下，刷新了十多项NLP任务的记录。BERT是基于Transformer结构的Encoder，其并行结构擅长捕捉长距离依赖，避免了RNN的顺序结构带来的缺点。BERT为双向Transformer结构，未标记文本的深层双向表示由所有层的双向上下文上共同进行条件化来预训练。与已提出的预训练模型相比，经过巨量语料训练的BERT具有捕捉到真正意义上的双向上下文信息的能力。由于BERT的出色表现，在Python库中已经收纳整理了可以调用的BERT函数库，使用者安装好BERT环境[1]，再自行将所需数据放入就可以直接使用。SimpleQuestions数据集中实体原始为独立的id形式，而在目标问题中为自然语言的单词形式表示，且id形式尽管可以展现出实体的唯一性，但实体的描述词语中会出现一些语义相通的词汇有助于问题的生成，所以本节也将三元组的自然语言形式作为模型输入的一部分。本文主要考虑通过表现良好的语义表示来生成问句，以此来验证这些表示是否在本任务有效。考虑到BERT在编码方面的优异成绩，本文用其捕捉词语级别的表示。

由上文可以了解到Transformer擅长捕捉长距离信息适用于长文本任务，实体对应于科学术语，由多个词来表达的，三元组的词组输入为短字符串，因此我们采用双向GRU进行编码来嵌入三元组的词语粒度表示。我们通过BERT预处理泛化每个词的嵌入向量的语义信息，再采用

[1]https://github.com/hanxiao/bert-as-service

双向GRU的最后一个隐藏状态作为每个词生成d维嵌入，该状态位于实体词组中每个词的嵌入之上的。

如图3所示，每个输入三元组表示为单词型序列$w = \{w_1, \ldots, w_i, \ldots, w_m\}$，调用BERT函数进行预处理编码得到维度一致的向量矩阵$x_w = \{x_{w_1}, \ldots, x_{w_i}, \ldots, x_{w_m}\}$。然后，将经过BERT预处理后的语义向量输入文本编码器中，即图1的词级编码器(Text Encoder)，其由双向GRU对x_w进行正向和反向读取来抓取输入序列x_w的有效语义信息，详细结构如图4所示。公式(8)和(9)可表示该过程。

$$\overrightarrow{h}_i = GRU(\overrightarrow{h}_{i-1}, x_{w_i}) \tag{8}$$

$$\overleftarrow{h}_i = GRU(\overleftarrow{h}_{i+1}, x_{w_i}) \tag{9}$$

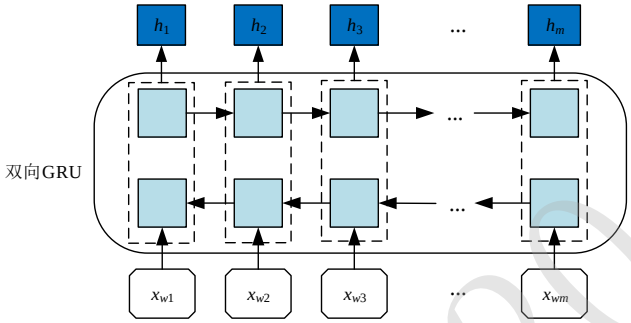


图 4. Text Encoder结构图

其中$\overrightarrow{h}_i$和$\overleftarrow{h}_i$分别为正向与反向的隐藏状态表示，再将每一步的正反隐藏状态拼接，得到整合的隐藏层状态$h = \{h_1, \ldots, h_i, \ldots, h_m\}$作为最后的编码层输出，其中$h_i = [\overrightarrow{h}_i; \overleftarrow{h}_i]$。

本文将e_N和h拼接作为多编码层的最终输出$\tilde{h} = [e_N; h; e_N]$。选择图编码层的最后一层的输出$e_N$和Text Encoder的最后一层隐藏状态$h_m$经过公式(10)成为解码层的隐藏状态的初始化值。

$$s_0 = ReLU([e_N; h_m]W_s) \tag{10}$$

2.4 基于Attention机制的GRU解码层

本文的解码器用来预测生成语义上流利且合适的简单问题，由于Transformer的并行结构性能效果非常好，我们观察了在Transformer结构上的任务，发现由于优良的并行结构使得模型计算时存在注意力过长，导致了生成的序列中出现与对应输入不相关的词汇。由于本文任务生成简单问题，而非长序列，顺序结构的RNN类网络每步的预测计算都依赖上一步的输出，其过程更贴合本文的任务目标。因此，本文使用基于注意力机制的GRU作为解码器。生成第t个单词的概率分布由公式(11)所示。

$$p(y_t|y_{\leq t}, x_w) = softmax(W_o\bar{s}_t) \tag{11}$$

其中，$\bar{s}_t$是经过注意力机制之后的t时刻隐藏状态，其由上下文向量c_t和GRU计算得到的隐藏状态s_t进行计算得到，如公式(12)所示。

$$\bar{s}_t = tanh(W_a[c_t; s_t]) \tag{12}$$

当前t时刻的隐藏状态s_t和通过注意力机制计算上下文向量c_t的详细计算由公式(13)和公式(14)给出。

$$s_t = GRU(y_{t-1}, c_{t-1}, s_{t-1}) \tag{13}$$

$$c_t = \sum_{r=1}^{m+2} \alpha_t \tilde{h}_r \tag{14}$$

其中，α_t为第t时间步的对齐向量，详细计算参照公式(15)。

$$\alpha_t = \frac{\exp(s_t^\top \tilde{h})}{\sum_{r=1}^{m+2} \exp(s_t^\top \tilde{h}_r)} \tag{15}$$

为了保证生成问句时句子主语的完整性，本文采用模板方法，将主语用占位符表示，在完成预测之后，再将问题对应的三元组主语补充上去。

3 实验结果与分析

3.1 实验数据

本文使用了具有id形式的英文数据集SimpleQuestions，表1所示为文本与id形式的问答对示例。本文的输入分为两部分，其一为对三元组进行构图预处理之后的实体-关系的向量表征。在实验过程中，对输入三元组进行构图操作，FreeBase被视为可靠的知识库，SimpleQuestions中所有的事实都是基于FreeBase的。由于FreeBase规模巨大，故在一些研究工作中，对FreeBase实行子集提取，如FB2M、FB5M等。SimpleQuestions中的三元组可在FreeBase的子集FB2M里查询到，故本文选取Huang等人(2019)所提供的基于FB2M训练TransE得到图嵌入作为本文KG-Emb。表2列出了FB2M与SimpleQuestions两者的具体信息。其二为自然语言形式下的文本三元组，通过BERT预处理之后的词向量表征。BERT预处理模型经过海量数据的预训练，可以赋予词语更完善的语义向量。并且，在编码端无需人为构造单词表，所有单词都可由BERT获得语义信息丰富的向量表征，这可以减轻诸多实体里出现的大量低频词汇所带来的单词量的压力，以加快模型的计算速度。

主语实体	谓词	宾语实体	目标问题
id:m/0ms5mg **text:** most of us are sad	Music/recording/ artist	**id:**m/0mjn2 **text:**Eagles	Which artist recorded most of us are sad?
id:m/086k8 **text:**warner bros entertainment	Film/production_ company/ films	**id:**m/0278x5r **text:** Saving Shiloh	What movie is produced by warner bros.?
id:m/02dtg **text:**Detroit	Location/location/ people_born_here	**id:**m/01s8mcb **text:** J. Moss	Who is a musician born in Detroit?

表 1. SimpleQuestions数据集id和文本形式样例

#	FB2M	SimpleQuestions
Training	14,174,246	75,910
Validation	N.A	10,845
Test	N.A	21,687
Predicates	6,701	1,837
Entities	1,963,130	131,681
Vocabulary	733,278	61,336

表 2. 数据集统计信息

3.2 评价标准

本文使用了一组在自然语言处理领域中较为完善的自动评估指标来评测模型效果：BLEU(Papineni et al., 2002)，METEOR(Banerjee and Lavie, 2005)，ROUGE-L(Lin, 2004)。考虑到语言表达的复杂性，本文还采用了人工评测的方法，邀请三位自然语言处理领域的研究生对生成的问题进行评估。

BLEU：BLEU在翻译任务中最先被使用，是将候选文本与一个或多个参考文本进行比较的评估标准。它是一种语法度量，用于计算生成的文本与参考文本之间的n元重叠。由于BLEU只考虑单词的连续匹配准确率，故其无法考虑到句子的语法以及语义方面的效果，如BLEU-1考虑的是单个单词的匹配，BLEU-2则考虑连续两个单词的匹配度等。

Meteor：Meteor是基于准确率与单词召回率上的加权调和平均数，其目的是解决BLEU标准固有的一些缺点。它还考虑到了一些其它指标没有发现的功能，例如，同义词匹配等，使其与人工评价的效果距离更近一步。

ROUGE-L：ROUGE是使用基于召回率的相似度量进行计算的指标，其基本思想是使用模型生成的文本和参考文本的n元组共现概率作为评价的基础，但无法评价句子是否流畅。在本文的评估中，选择ROUGE-L作为评估标准，其基本思想是匹配两个文本单元之间最长的公共序列。

目标问句	Who directed walter hill filmmaker ?	BLEU
生成句A	Who was the director of walter hill ?	58.06
生成句B	Who directed the film walter hill ?	77.58

表 3. BLEU值得分举例，表中BLEU分数值为BLEU-1至BLUE-4的得分均值

人工评估(Human Evaluation)：尽管自动评测在某种程度来说可以估量生成的问句与参考问句有多相近，但是当语义相近而语法表达结构上不同时，许多隐藏着的限制就显示出来了。由一个目标问句和两个模型预测的问句组成的一个例子可以更清楚的说明这个问题，如表3。尽管表中的三个句子在语义表达方面类似，但是利用自动评测标准来度量时分数差距却有将近20个百分点。此外，自动评测也不能确定是否有三元组中相关的关系词出现在预测的问句中。此时，人工来评测给出句子的流畅度与相关度是必不可少的。

3.3 基准模型

本文做了多个对比实验证明提出方法的有效性，具体的对比模型如下详释，Zero-shot与Enc-Dec模型的计算以及预处理保持与原始文章一致，其余模型保持与本文一致的预处理，但不加入图层次的编码层。

Transformer(Vaswani et al., 2017)：Transformer是当前在机器翻译任务上非常火热的模型。本文使用Transformer作为一个对比实验，观察它在此文任务中的表现。

Zero-shot(ElSahar et al., 2018)：Zero-shot利用特殊标识符替代关系词以解决在测试集可能出现的在训练集里未出现的关系词的问句。在该模型中，SimpleQuestions数据集三元组的实体和关系以TransE嵌入为初始化。

Enc-Dec(Serban et al., 2016)：Enc-Dec是根据Serban等人的工作来实现的。本文只使用了单占位符，Serban等人还使用了多类占位符（MP）作为输入序列，但由于在文章中并没有将这些占位符的类别罗列出来，并且MP所带来的贡献并不大，故本文的实验方法中，不能报告以MP作为输入的实验结果。

CopyNet(Wang et al., 2018)：Wang等人(2018)在基于知识库的QG任务上融入了复制机制，本文复现了融合复制机制的CopyNet模型。

3.4 实验设置

本文使用多编码层，在单词粒度上的编码器使用双向GRU结构，每个GRU隐藏节点数为500，由BERT初始化词向量，维度为768；基于图的编码器使用5个多头注意力机制和4个循环块的Transformer结构，使用TransE对三元组进行构图初始化，维度为250。解码器使用结合Attention的单层GRU结构，每个GRU隐藏节点数为1000，用GloVe(Pennington et al., 2014)预训练好词嵌入对词向量初始化。本文基于Tensorflow平台(Abadi et al., 2016)，采用Adam优化算法(Kingma and Ba, 2014)。学习率初始化为1e-3，每500步进行一次衰减。模型使用mini-batch方式训练，batch大小设置为64。

3.5 结果分析

（一）自动评测

如表4所示，GT-KBQG在BLEU-1、BLEU-2、Meteor和ROUGE-L指标上提升效果明显。Transformer模型虽然具有并行性的计算优势，但仅靠输入的单个三元组获取的语义背景信息始终不够完善，依赖并行结构的特征，其得到的效果虽然在BLEU-1上优于Zero-shot和Enc-Dec模型，但在BLEU-2，BLEU-3和BLEU-4指标上均逊色于Zero-shot模型。与Enc-Dec模型相

#	BLEU-1	BLEU-2	BLEU-3	BLEU-4	Meteor	ROUGE-L
Transformer	68.56	43.31	31.32	23.29	36.31	68.80
Zero-shot	62.44	**50.62**	**40.82**	**31.10**	36.24	61.32
Enc-Dec	60.92	46.05	36.32	27.36	35.07	60.28
CopyNet	69.74	46.28	34.92	27.12	37.48	69.80
GT-KBQG	**72.54**	47.37	35.81	27.86	**38.05**	**70.24**

表 4. 自动评估结果

比，GT-KBQG在Meteor（+2.98）和ROUGE-L（+9.96）指标上提升效果非常明显，从一方面说明GT-KBQG生成的问题与目标问题的语义和词语共现效果更好，但在3-gram和4-gram重叠（BLEU-3、BLEU-4）匹配上，效果却不明显说明了词语共现性提升，但可能表达形式更丰富，故在多元匹配上与目标问句相符的较少。CopyNet模型对比其他的基准模型效果提升也十分明显，但由于其着重于对三元组中出现的低频词处理，而未考虑信息表达的多样化，故在1-gram和2-gram上的词语共现不如GT-KBQG，且在BLEU-3和BLUE-4上的效果也逊色于Zero-shot模型。虽然在BLEU-1的匹配上，GT-KBQG效果最优，但是在多元匹配上与Zero-shot依然存在差距。根据后面的人工评价，我们推测是因为与参考问句的一致性导致的。多元匹配对词语组成问句序列具有严格要求，但由于GT-KBQG模型的BLEU-1效果最好，又说明本模型具有与参考问句相匹配的更多个词语，推测是因为在语句表达形式上与参考问句一致性没有Zero-shot高，但在总体的语义表达上，Meteor（+1.81）和ROUGE-L（+8.92）又优于Zero-shot，说明虽与参考问句表达形式一致性不高，但是表达目的应该是一致的。

（一）人工评价

#	Transformer	Zero-shot	Enc-Dec	CopyNet	GT-KBQG
Sim.	49.86%	51.34%	50.29%	51.27%	48.87%
Var.	42.90%	40.72%	38.72%	43.76%	57.43%

表 5. 人工评估结果

由于语言表述的复杂性，自动评测只能表现出针对目标问题这一单一的参考文本的效果，无法完全准确反映生成的问题是否合适，以及对生成问句与对应三元组的信息相关性、背景信息的描述等方面有所欠缺。因此，本文采用人工评价的方式对GT-KBQG模型生成问句的多样性进行评估，表5中Sim.表示similarity，该行表示与参考问句几乎完全相似（可能冠词不一致）的句子占所有被选问句的比例，表中的模型所占比例都在一半左右，具有较高的一致性。本文除了要考虑生成问句的一致性外，更加要关注是否具有多样性，对背景信息是否丰富进行标注，即表5中的Var.(variety)。

本文在测试集中随机抽取三份数据用于人工评价，每份有三百个三元组-问题对，同时每份以三个人参与，与标准问题以一对示例的形式进行标注。由表5所示，GT-KBQG 模型与目标问题的一致性不是最高，即在Sim.结果上，比Transformer（-0.99%）、Zero-shot（-2.47%）、Enc-Dec（-1.42%）和CopyNet（-2.4%）的结果都低，可以推测出目标问题较多比较简单，但也会出现不少具有实体描述信息的问句。但是在Var.上的评价结果GT-KBQG最好，比基准模型中表现最好的CopyNet高了13.67个百分点，这证明了本文对三元组特征表示的加强赋予模型更加丰富的背景信息，从而生成的问句中的实体的表述信息更多。

（三）样例分析

在评测的结果上，只能基于评估数据对生成的问题进行推测分析，无法直观感受生成问句的质量效果，表6选取目标问题和Enc-Dec模型的样例与GT-KBQG模型进行样例对比。

根据表6中所展示的样例可以看出，比起Enc-Dec模型，GT-KBQG生成出更加具有描述性的问题。如示例1和示例3中Enc-Dec所生成的问题正确但非常简洁，对于实体描述信息非常少，该类问题可以有非常多的回答，若与QA系统作为对偶学习任务，可能会降低系统的准确度。相比以上两个模型，GT-KBQG对于提问对象“Robert Drummond”的描述更进一步，该问题提供了“Robert Drummond”的之前的职业信息“football player”，比Human对实体所述的信

#	Facts	Models	Generated Questions
1	-Syracuse- -people born here- -Robert Drummond-	Human	Which football player was born in Syracuse New York?
		Enc-Dec	Who was born in Syracuse?
		GT-KBQG	Which former professional football player was born in Syracuse ?
2	-lady penelope creighton-ward- -character created by- -gerry anderson-	Human	Who created lady penelope creighton-ward?
		Enc-Dec	Who created lady penelope creighton-ward?
		GT-KBQG	Who created the fictional character lady penelope creighton-ward?
3	-marcus allen- -notable types- -american football player-	Human	which american sport does marcus allen play in?
		Enc-Dec	what is marcus allen?
		GT-KBQG	which sport is marcus allen known for?

表 6. 样例对比

息更细，但比起人工生成的问题，GT-KBQG的描述仅针对了对象实体，Human行对主题实体也进行了描述，可以让读者了解“Syracuse”处于New York。但总体上来看，与Enc-Dec模型相比，GT-KBQG生成问句时着重于对实体背景信息的描述，产生的问题更加精准化和多样化。在自动评测结果中，尽管在Enc-Dec模型进行了改善，但在多元重叠效果上并不太明显，对生成的问题进行人工评价，发现其原因除了模型自身依旧存在缺陷之外，与目标问题也有关联。例如样例2，目标问题较为简单，Enc-Dec模型本身存在生成句简短的特征，故与目标问题完全匹配，而进行改善的模型尽管生成了更加丰富的实体背景信息，但在自动评测指标上的结果却不如Enc-Dec效果好，而GT-KBQG预测出的背景信息越多越完善，BLEU等自动指标效果也会愈低。

4 总结与展望

本文提出了基于Graph Transformer的问题生成模型，缓解模型生成单一化的问题。与额外加入上下文信息作为输入的模型不同，本文仅对三元组本身的表示进行多样化加强，进而获得丰富的语义背景信息。在SimpleQuestions数据集上进行实验，与基准模型相比，人工评测与自动评测的结果证明了对三元组语义加强操作在一定程度上丰富了问题生成的多样性。本文尽管在实验结果上展现出较好的效果，但依旧存在许多问题待解决，比如与人工生成的问题依旧存在差距，且遇到未涉及的三元组或实体关系时，需要重新构造一个更完善的图，在后续工作上可以考虑自行构造出更大更完善的知识图。由于时间环境、设备等因素的限制，本文提出的GT-KBQG模型只基于SimpleQuestions上实验验证，接下来将会在多种数据集上进行研究，以及将进一步深入消融实验分析。

致谢

本文的工作作为硕士毕业论文的一部分，受到国家自然科学基金（No. 61972173)支持。感谢匿名审稿人对我们工作提出的建设性修改意见。

参考文献

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek Gordon Murray, Benoit Steiner, Paul A. Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu and Xiaoqiang Zheng. 2016. TensorFlow: A System for Large-Scale Machine Learning. In *12th USENIX Symposium on OSDI*, page 265–283.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72.

Kurt D. Bollacker, Robert P. Cook and Patrick Tufts. 2007. Freebase: A Shared Database of Structured General Human Knowledge. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, Vancouver, British Columbia, Canada*, pages 1962–1963.

Antoine Bordes, Jason Weston, Ronan Collobert and Yoshua Bengio. 2011. Learning Structured Embeddings of Knowledge Bases. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, San Francisco, California, USA*.

Antoine Bordes, Nicolas Usunier, Sumit Chopra and Jason Weston. 2015. Large-scale Simple Question Answering with Memory Networks. *http://arxiv.org/abs/1506.02075*.

Deng Cai and Wai Lam. 2019. Graph Transformer for Graph-to-Sequence Learning. *http://arxiv.org/abs/1911.07470*.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *EMNLP*, pages 1724–1734.

Jacob Devlin, Ming-Wei Chang, Kenton Lee and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT, Minneapolis, MN, USA, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Daniel Duma and Ewan Klein. 2013. Generating Natural Language from Linked Data: Unsupervised template extraction. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) – Long Papers*, pages 83–94.

Hady ElSahar, Christophe Gravier and Frédérique Laforest. 2018. Zero-Shot Question Generation from Knowledge Graphs for Unseen Predicates and Entity Types. In *NAACL-HLT, New Orleans, Louisiana, USA, Volume 1 (Long Papers)*, pages 218–228.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O. K. Li. 2016. Incorporating Copying Mechanism in Sequence-to-Sequence Learning. In *ACL (Volume 1: Long Papers)*, pages 1631–1640.

Xiao Huang, Jingyuan Zhang, Dingcheng Li and Ping Li. 2019. Knowledge Graph Embedding Based Question Answering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, Melbourne, VIC, Australia*, pages 105–113.

贾熹滨, 李让, 胡长建等. 智能对话系统研究综述[J]. 北京工业大学学报, 2017, 43(009):1344-1356.

Mitesh M. Khapra, Dinesh Raghu, Sachindra Joshi and Sathish Reddy. 2017. Generating Natural Language Question-Answer Pairs from a Knowledge Graph Using a RNN Based Question Generation Model. In *EACL, Valencia, Spain, Volume 1: Long Papers*, pages 376–385.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. In *http://arxiv.org/abs/1412.6980*.

Rik Koncel-Kedziorski, Dhanush Bekal, Yi Luan, Mirella Lapata and Hannaneh Hajishirzi. 2019. Text Generation from Knowledge Graphs with Graph Transformers. In *NAACL-HLT, Minneapolis, MN, USA, Volume 1 (Long and Short Papers)*, pages 2284–2293.

Yankai Lin, Zhiyuan Liu, Huan-Bo Luan, Maosong Sun, Siwei Rao and Song Liu. 2015. Modeling Relation Paths for Representation Learning of Knowledge Bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP, Lisbon, Portugal*, pages 705–714.

Hailun Lin, Yong Liu, Weiping Wang, Yinliang Yue and Zheng Lin. 2017. Learning Entity and Relation Embeddings for Knowledge Resolution. In *International Conference on Computational Science, ICCS, Zurich, Switzerland, Volume 108*, pages 345–354.

Chin Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of summaries. In *In Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*.

Thang Luong, Hieu Pham and Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP, Lisbon, Portugal*, pages 1412–1421.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *ACL, Philadelphia, Pennsylvania, USA*, pages 311–318.

Jeffrey Pennington, Richard Socher and Christopher D. Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP, Doha, Qatar*, pages 1532–1543.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev and Percy Liang. 2016. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP, Austin, Texas, USA*, pages 2383–2392.

Iulian Vlad Serban, Alberto García-Durán, Çaglar Gülçehre, Sungjin Ahn, Sarath Chandar, Aaron C. Courville and Yoshua Bengio. 2016. Generating Factoid Questions With Recurrent Neural Networks: The 30M Factoid Question-Answer Corpus. In *ACL, Germany, Volume 1: Long Papers*, pages 588–598.

Dominic Seyler, Mohamed Yahya and Klaus Berberich. 2015. Generating Quiz Questions from Knowledge Graphs. In *Proceedings of the 24th International Conference on World Wide Web, Florence, Italy*, pages 113–114.

Dominic Seyler, Mohamed Yahya and Klaus Berberich. 2017. Knowledge Questions from Knowledge Graphs. In *ICTIR, Amsterdam, The Netherlands*, pages 11–18.

Ilya Sutskever, Oriol Vinyals and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. In *NeurIPS*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA*, pages 5998–6008.

Denny Vrandecic and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. In *Commun. ACM, Volume 57*, 57(10):78–85.

Zhen Wang, Jianwen Zhang, Jianlin Feng and Zheng Chen. 2014. Knowledge Graph Embedding by Translating on Hyperplanes. In *roceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, Québec City, Québec, Canada*, pages 1112–1119.

Hao Wang, Xiaodong Zhang and Houfeng Wang. 2018. A Neural Question Generation System Based on Knowledge Base. In *NLPCC*, pages 133–142.

张建华, 陈家骏. 自然语言生成综述[J]. 计算机应用研究, 2006(08):1-3.

基于BERT与柱搜索的中文释义生成

范齐楠 孔存良 杨麟儿 杨尔弘
北京语言大学

摘要

释义生成任务是指为一个目标词生成相应的释义。前人研究中文释义生成任务时未考虑目标词的上下文，本文首次在中文释义生成任务中使用了目标词的上下文信息，并提出了一个基于BERT与柱搜索的释义生成模型。本文构建了包含上下文的CWN中文数据集用于开展实验，除了BLEU指标之外，还使用语义相似度作为额外的自动评价指标，实验结果显示本文模型在中文CWN数据集和英文Oxford数据集上均有显著提升，人工评价结果也与自动评价结果一致。最后，本文对生成实例进行了深入分析。

关键词： 中文释义生成 ； BERT ； 柱搜索

Chinese Definition Modeling Based on BERT and Beam Seach

Qinan Fan, Cunliang Kong, Liner Yang, Erhong Yang
Beijing Language and Culture University

Abstract

Definition modeling task refers to generate a corresponding definition for the target word. Previous study on Chinese definition modeling task did not consider the context of the target word. For the first time, this thesis uses the context information of the target word in Chinese definition modeling task and proposes a definition generation model based on BERT and beam search. For experiments, we construct the CWN Chinese definition modeling dataset containing context of the target word. In addition to BLEU score, semantic similarity is used as an additional automatic evaluation metric. The experimental results show that the model has a significant improvement in Chinese CWN dataset and English Oxford dataset, and the results of human evaluation are consistent with the results of automatic evaluation. At last, this thesis makes an in-depth analysis of the generated instances.

Keywords: Chinese Definition Modeling , BERT , Beam Seach

基金项目：北京语言大学研究生创新基金(中央高校基本科研业务费专项资金) (20YCX139)；北京语言大学语言资源高精尖创新中心项目(TYZ19005)；国家语委信息化项目(ZDI135-105)

1 引言

释义生成（又称释义建模，Definition Modeling）任务由 Noraset et al. (2017)首次提出，任务目的是为一个给定的目标词生成相应的释义。释义生成任务不论在自然语言处理（Natural Language Processing，简称NLP）领域还是实际应用场景中，都具有非常重要的研究意义和价值。**在NLP领域**：(1) 预训练的静态词向量经常会被用来初始化词嵌入，其质量好坏会对所做任务产生很大影响。目前常用的预训练词向量的质量评价方法有相似性、类比推理等，相比于这些评价方法，为预训练的词向量生成一句文本释义，能够更直观地反映词向量质量。(2) 低维密集词向量的可解释性问题，一直是深度学习领域关注的焦点。以人类可读的形式为低维词向量生成文本释义，可以对词向量捕获到的语义信息予以解释。(3) 词典释义经常被作为外部语义知识融入其它NLP任务中，本任务可以极大丰富词典释义资源。**在实际应用中**，释义生成任务也可以为词典编撰者及语言学习者提供很大帮助：(1) 不论是编撰新词典还是修订已有词典，都需要耗费大量的人力和物力，而释义生成系统可以作为词典编著者强有力的辅助工具，节省编撰成本。(2) 对于语言学习者，当他们需要查询陌生词汇时，受限于词典的收录能力，查询不到词语的情况时有发生。当遇到多义词时，他们也只能根据上下文去推断应取哪个义项，往往不能保证准确性。而释义生成任务不仅可以为新词语生成释义，也可以通过融合上下文的方法生成词语在特定语境下的释义。

Noraset et al. (2017)最早在英文上研究释义生成任务，出于评价预训练词向量质量的目的，这项工作使用目标词的预训练词向量作为输入来生成释义，根据生成释义是否准确来验证词向量是否包含正确的语义信息。考虑到预训练词向量会将多义词的多个义项合并的问题，Gadetsky et al. (2018)借鉴语义消歧任务，采用非参数贝叶斯的方法实现了动态多义，训练模型生成词语在给定上下文中的释义。Ishiwatari et al. (2019)后来将目标词预训练词向量和上下文向量直接拼接后用于释义生成，该方法达到了目前英文释义生成任务的最优结果。前人研究证明，上下文信息不仅可以对目标词进行消歧，也可以补充更多的语义信息，在释义生成任务中起到了非常重要的作用。在中文上，Yang et al. (2020)首次开展了释义生成任务研究，此项工作将HowNet中的义原作为外部语义知识融入模型来提升生成效果，但没有考虑目标词的上下文信息。

基于上述问题，本文首次将目标词的上下文引入中文释义生成任务，将任务重新定义为给定一个目标词及其所在上下文，为其生成相应的释义，图1中给出了数据示例：

被释义词：意外		
上下文：	1. 好在我们都已买了保险，如果发生**意外**，一切都由保险公司理赔。	2. 我亲口告诉她实情，令我**意外**的是，她出奇的平静，似乎早知这一刻。
释义：	料想不到的事件，指不幸的灾难变故。	形容人感到惊讶 。

Figure 1: 中文释义生成示例

由于词典资源的获取难度较高，且词典本身的容量有限，释义生成任务缺乏供模型训练的大量数据，属于低资源的文本生成任务。相较于前人工作中普遍使用的LSTM模型，参数更多、性能更好的模型（如Transformer）难以在释义生成任务上获得充分训练，因此无法取得很好的效果。使用预训练语言模型是解决这一问题的有效方法，可以将预训练语言模型在大规模语料上训练获得的先验知识迁移到释义生成任务中。因此，本文提出了基于预训练语言模型BERT与柱搜索的释义生成模型。如图2所示，该模型采用编码器-解码器框架，将预训练的BERT作为模型编码器，用于对目标词及上下文直接拼接后的序列进行编码，将Transformer作为模型解码器，用于生成释义。在测试阶段，为缓解陷入局部最优解的问题，我们将前人使用的贪心搜索（Greedy search）策略替换为柱搜索（Beam search）策略来扩大搜索空间，以兼顾模型解码的效率和性能，此策略进一步提升了生成效果。

为了验证模型的有效性，本文基于中文词汇网络（Chinese WordNet，简称CWN）构建了新的中文释义生成数据集，与Yang et al. (2020)使用的数据集不同，CWN数据集中每条数据包含被释义词、上下文及释义三项内容。除了BLEU指标之外，本文采用语义相似度作为额外的

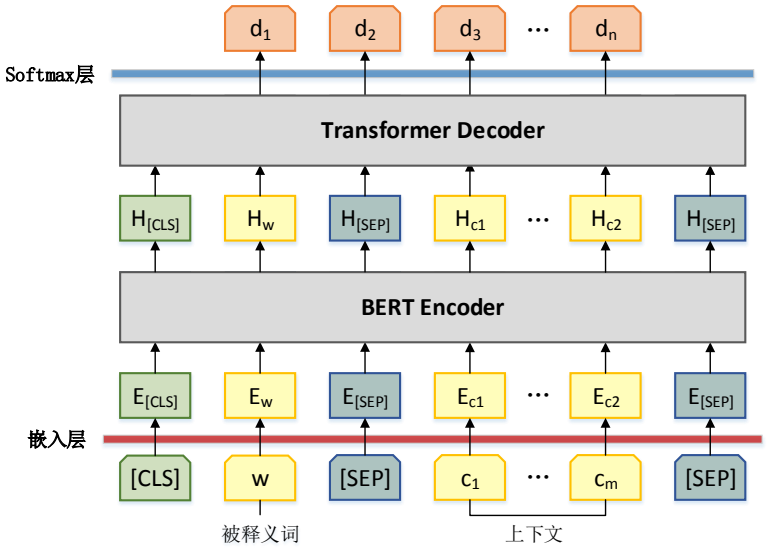


Figure 2: 模型图

评价指标，该指标使用余弦相似度计算生成释义和参考答案句向量在语义层面上的相似程度。本文提出的模型在中文CWN数据集上的实验结果相比基线模型提升显著，在Gadetsky et al. (2018)构建的英文Oxford数据集上实验结果同样明显超出基线模型。另外，我们对本文模型及基线模型在CWN数据集上的生成结果进行了人工评价，评价结果也与实验结果一致，表明了本文所提出方法的有效性。最后，本文分析了数据分布情况对释义生成结果的影响，并对模型的生成结果进行了实例分析。

本文的主要贡献有：

- 首次在中文释义生成任务中使用了目标词的上下文，更完整地定义了中文释义生成任务。
- 提出了基于BERT与柱搜索策略的释义生成模型，有效弥补了数据量不足的缺陷，获得了很好的效果。
- 对本文模型生成结果进行了深入分析，总结了中文释义生成任务仍待解决的四大问题。

2 融合上下文的中文释义生成模型

本文提出的中文释义生成任务，指的是生成目标词在特定上下文中的释义。如图1给出的数据示例，当给定相同词、不同上下文时，模型生成的释义也不同。形式化地，即给定一个词语w，以及包含该词语的一句上下文$C = [c_1, \ldots, c_m]$，为其生成一句相应的释义$D = [d_1, \ldots, d_n]$。模型的生成过程可以用条件概率表示为：

$$P(D|w, C) = \prod_{i=1}^{n} p(d_i|d_{<i}, w, C) \quad (1)$$

为了弥补缺乏训练数据的问题，本文在Transformer模型的基础上，提出了基于预训练语言模型BERT和柱搜索策略的模型，整体模型架构如图2所示。该模型使用BERT初始化编码器参数，使用Transformer作为模型解码器，然后在释义生成任务上进行微调，本节将对该模型进行详细介绍。

2.1 BERT编码器

由于Transformer模型的参数量庞大，需要借助大规模数据进行参数训练，而中文释义生成属于低资源任务，数据量远远未达到训练要求，因此难以达到理想效果。将预训练语言模型迁移到低资源任务上，是弥补数据量不足的有效方法。BERT (Devlin et al., 2018)是在大规模无标注语料上预训练的基于Transformer的多层双向编码器，近两年被应用于多项NLP任务中并刷新了最佳成绩。基于此，本文将BERT作为模型编码器，让模型能够获得BERT从大规模语料中学到的先验知识。

本文将目标词w和上下文序列C直接拼接后作为输入序列。在嵌入层，本文通过两种方式将目标词和上下文区分开。首先，使用特殊符号“[SEP]”将它们分隔开。其次，为它们分别加

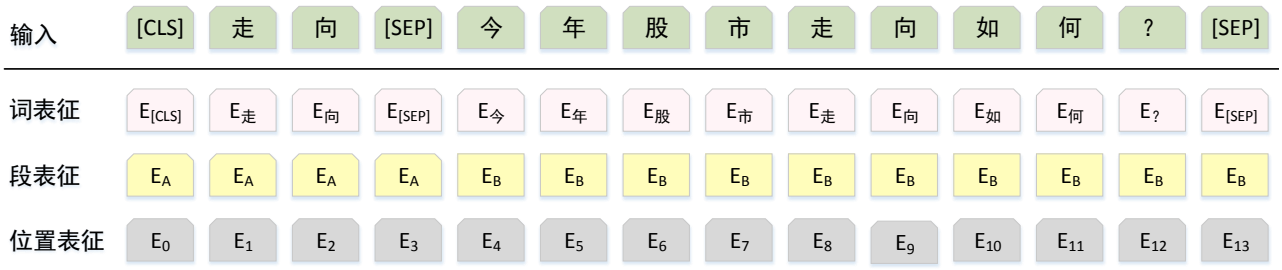


Figure 3: BERT嵌入层

上不同的段表征，将目标词的段表征置为0，上下文的段表征置为1。如图3所示，对于每一个词，其词嵌入由对应的词表征（Token embedding），段表征（Segment embedding）和位置表征（Position embedding）相加产生。经过BERT编码后得到最终的序列表征H：

$$H = \text{BERT}([\text{CLS}] \circ \text{w} \circ [\text{SEP}] \circ \text{C} \circ [\text{SEP}]) \tag{2}$$

其中∘表示连接操作，H由整个序列的上下文相关词向量构成，例如H_0是特殊符号“[CLS]”的词向量。H即为编码器的输出，传给Transformer解码器用于解码。

2.2 Transformer解码器

Transformer(Vaswani et al., 2017)模型是基于多头注意力机制的序列生成模型，近年来被广泛应用于NLP文本生成任务中。该模型的解码器是根据上一时间步的输出预测当前时间步的输出，最后将每个时间步输出的词语连起来得到最终的生成序列。

在本任务中，模型首先将之前时间步生成的释义序列通过嵌入层编码后再加上词的位置表征，得到的词嵌入作为Transformer解码器的输入。Transformer解码器由N层相同的模块构成，上层模块输出的隐状态是下层模块的输入。每个模块包含三个子层：一个掩码多头自注意力层、一个编码器-解码器多头注意力层和一个前馈神经网络层。其中多头注意力层由多个注意力层得到的向量拼接而成，每个注意力层采用缩放点积运算：

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{3}$$

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{Attention}_1, \text{Attention}_2, ..., \text{Attention}_\text{h}) \tag{4}$$

其中Q、K和V分别表示查询矩阵（Query）、键值矩阵（Key）和实值矩阵（Value），h表示注意力层的头数。掩码多头自注意力层的Q、K和V相同，都是释义的词嵌入经线性映射后的向量，掩码操作使模型在训练阶段的每个时间步不能看到未来信息。编码器-解码器多头注意力层的Q来自于上一层解码器的输出，K和V来自于编码器的输出。另外，这三个子层之后都会接一个归一化层和残差网络，归一化层能够加快模型训练速度，残差网络能够防止神经网络模型退化。

2.3 柱搜索策略

在解码阶段，Seq2seq模型常用的搜索算法有贪心算法和柱搜索算法。在释义生成任务中，前人都选用了贪心算法，而该算法具有一些弊端。在每个时间步都选取概率最大的词，很容易陷入局部最优解。另外，当某个时间步概率最大词错误时，该错误也会被继续传播。

柱搜索是一种平衡性能和消耗的搜索算法，目的是解码出相对较优的序列，能够一定程度上缓解上述贪心算法的问题。因此本文采取了柱搜索策略，与贪心算法在解码的每个时间步都选择概率最大的词不同，柱搜索算法会结合之前时间步已生成的序列，在当前时间步选择使得整体序列概率最大的前K个词，最后将K个序列中概率最大的作为最终输出，相比贪心算法能够进一步提升生成效果。

3 实验

3.1 数据集

不论在英文还是中文上，词典语料都非常稀缺。目前在中文上，释义生成任务还没有同时包含词语、上下文及释义的数据集。中文词汇网络（CWN）[0]是一个由台湾中研院开发的词汇语义关系知识库，其中的大部分义项都具有多条例句，我们选用CWN构建了高质量的中文释义生成数据集。本文使用opencc-python工具[1]将数据由繁体中文转换为简体，使用jieba工具[2]对全部数据进行分词，并对其中的特殊字符等做了预处理。然后按照被释义词数量8:1:1的比例，将数据集切分为训练集、验证集和测试集，最终每条数据包含一个被释义词、一条上下文和相应的释义。本文在英文Oxford数据集上也开展了实验，此数据集由Gadetsky et al. (2018)通过牛津在线词典[3]提供的API构建。CWN及Oxford数据集的规模统计如表1所示，其中上下文长度和释义长度是平均长度，中文CWN数据集按字统计，英文Oxford数据集按词统计。

数据集		被释义词数量	释义数量	数据条数	上下文长度	释义长度
CWN	训练集	6,574	21,736	67,861	34.49	14.76
	验证集	823	2,606	8,082	34.73	14.60
	测试集	824	2,774	8,599	34.06	14.72
Oxford	训练集	33,128	97,780	97,855	17.74	11.02
	验证集	8,867	12,230	12,232	17.80	10.99
	测试集	8,850	12,230	12,232	17.56	10.95

Table 1: 数据集规模统计

另外，由于CWN数据集具有多上下文的特点，本文对CWN切分后的数据集每条释义对应的上下文数量做了统计。如表2所示，在三个数据集中，上下文数量分布情况非常类似，超过90%的释义都有2条以上的上下文，有3条上下文的释义最多，达到60%以上。Oxford数据集中几乎全部的释义都只有1条对应上下文，相比之下，CWN数据集的上下文资源更加丰富。

数据集		上下文数量						
		1	2	3	4	5	6	7+
训练集	释义数量	794	3,342	13,671	1,896	768	1,063	202
	占比	3.65%	15.38%	62.90%	8.72%	3.53%	4.89%	0.93%
验证集	释义数量	78	424	1671	202	88	122	21
	占比	2.99%	16.27%	64.12%	7.75%	3.38%	4.68%	0.81%
测试集	释义数量	111	408	1777	229	96	134	19
	占比	4.00%	14.71%	64.06%	8.26%	3.46%	4.83%	0.68%

Table 2: CWN数据集释义包含的上下文数量统计

3.2 基线模型

本文将Transformer模型(Vaswani et al., 2017)和LOG-CaD模型(Ishiwatari et al., 2019)作为基线模型。Transformer模型是基于多头自注意力机制的模型，近年来在文本生成任务中被广泛应用，本文不再做详细介绍。LOG-CaD模型是针对英文释义生成任务提出的模型，该模型在四个英文数据集上都取得了很好的结果。LOG-CaD模型基于编码器-解码器框架，其中编码器共包含三个部分：

- **局部上下文编码器**：局部上下文是指给定的一句包含目标词的上下文。该模型采用双向LSTM模型对局部上下文进行编码。在解码的每个时间步，都通过注意力机制计算当前隐状态和局部上下文每个时间步隐状态的注意力系数，加权后得到最终的局部上下文向量表示。

[0]https://lope.linguistics.ntu.edu.tw/cwn2/
[1]https://github.com/yichen0831/opencc-python
[2]https://github.com/fxsjy/jieba
[3]https://en.oxforddictionaries.com/

- **全局上下文编码器**：全局上下文是指从大规模语料中获得的全局语义信息。CBOW是使用Google新闻语料预训练的静态词向量，该模型从CBOW中提取出目标词的预训练词向量作为目标词的全局上下文表示。
- **目标词字符级特征提取器**：由于英文单词中的词缀可以体现出重要的词义信息，例如以"-ist"结尾的通常是名词，表示专家或从事某活动的人。因此，该模型采用CNN模型提取了目标词的字符级特征表示，用于获取词缀中包含的语义信息。

模型将上述三个编码器的输出拼接后作为解码器的输入。该模型的解码器采用了单向LSTM模型，并在每个时间步增加了门控机制，对当前时间步输出的隐状态和编码器输出的拼接向量进行过滤，以更好地控制多种输入信息之间的交互。

3.3 实验设置

本文的Transformer模型基于FAIR开源代码库[4]实现，使用预训练的中文词向量(Li et al., 2018)和fastText词向量(Bojanowski et al., 2017)分别对中文和英文数据的词嵌入进行初始化，词表维数为300维，解码器的输入和输出词嵌入矩阵共享权重。模型的编码器和解码器均设置为6层，其中多头注意力层有5个注意力头，前馈层维度为2048。训练过程使用Adam优化器(Kingma and Ba, 2015)更新模型参数，初始学习率为1e-7，增长到5e-4后逐步下降，dropout设置为0.3。

本文基于BERT的模型采用的是base版本的BERT预训练模型，在transformers开源代码库(Wolf et al., 2019)基础上实现。本文的模型训练分为两个阶段：第一阶段固定编码器参数，仅训练解码器，学习率设置为5e-4，warm-up设置为4000；第二阶段同时微调编码器和解码器，学习率设置为2e-5，warm-up设置为2000。两阶段的dropout均设置为0.2。中文和英文释义的词嵌入使用了和上述相同的预训练词向量进行了初始化，Transformer解码器的超参数设置也与上述一致，优化器同样使用Adam。另外，在选择最优模型时采取了early-stop策略，每轮模型都会在验证集上计算PPL和BLEU值（考虑到效率问题，这里使用NLTK translate包[5]计算sentence BLEU，与测试时的BLEU指标不同但高度相关），当验证集上PPL超过10轮不再增长时，取这10轮中BLEU值最高的模型保存下来用于测试。

3.4 实验结果

本文分别在中文CWN数据集和英文Oxford数据集上评测了模型效果。由于前人使用的BLEU(Papineni et al., 2001)评价指标只能衡量生成释义与参考答案在字面上的相似性，因此本文将语义相似度作为额外的评价指标，从语义层面衡量生成释义和参考答案的相似性。该指标的计算方法是，首先使用sentence-transformers工具[6]分别对生成释义和参考答案句子进行编码(Reimers and Gurevych, 2019; Reimers and Gurevych, 2020)，然后使用scipy包[7]计算两个句向量的余弦相似度。表3和表4中分别给出了BLEU和语义相似度两个指标的实验结果。其中Transformer (Vaswani et al., 2017)和LOG-CaD (Ishiwatari et al., 2019)为本文的基线模型，ESD-sem为Li et al. (2020)提出的基于显式语义分解的模型。BERT-fix-encoder表示训练的第一阶段固定编码器参数仅训练解码器，BERT-fine-tune表示第二阶段同时微调编码器和解码器，这两个模型解码时均使用贪心算法。

模型	CWN		Oxford	
	验证集	测试集	验证集	测试集
Transformer (Vaswani et al., 2017)	21.16	20.77	17.03	17.02
LOG-CaD (Ishiwatari et al., 2019)	30.76	29.58	19.13	18.95
ESD-sem (Li et al., 2020)	–	–	–	20.86
BERT-fix-encoder (Greedy)	38.96	37.25	19.87	20.14
BERT-fine-tune (Greedy)	**43.25**	**40.05**	**21.95**	**22.01**

Table 3: 实验结果（BLEU）

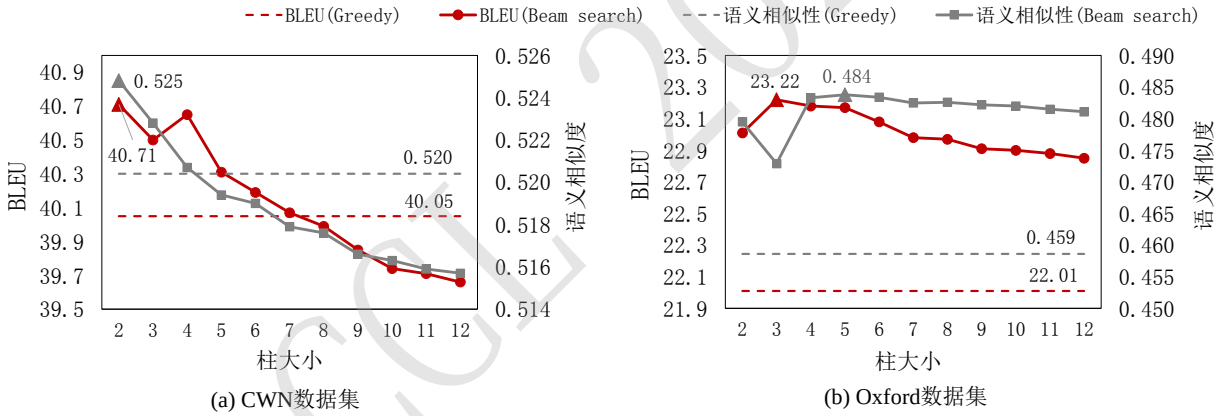
[4]https://github.com/pytorch/fairseq
[5]https://www.nltk.org/_modules/nltk/translate/bleu_score.html
[6]https://github.com/UKPLab/sentence-transformers
[7]https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.distance.cdist.html

模型	CWN		Oxford	
	验证集	测试集	验证集	测试集
Transformer (Vaswani et al., 2017)	0.273	0.269	0.369	0.368
LOG-CaD (Ishiwatari et al., 2019)	0.362	0.415	0.269	0.306
BERT-fix-encoder (Greedy)	0.508	0.486	0.443	0.443
BERT-fine-tune (Greedy)	**0.538**	**0.520**	**0.473**	**0.459**

Table 4: 实验结果（语义相似度）

可以看到，Transformer模型在中文CWN数据集上表现欠佳，BLEU和语义相似度两个指标均与LOG-CaD模型有较大差距。在英文Oxford数据集上，Transformer模型的BLEU值与LOG-CaD模型差距不大，语义相似度甚至超过了LOG-CaD模型。有了BERT的加持后，本文提出的BERT-fix-encoder（Greedy）模型在两个数据集上的结果都得到了非常显著的提升，经过第二阶段微调后的模型比起第一阶段也均有一定提升，验证了本文模型和两阶段训练策略的有效性。

本文在BERT-fine-tune (Greedy) 模型基础上，将贪心算法改进为柱搜索算法，对柱取2-12不同大小时的BERT-fine-tune模型结果进行了对比实验。如图4所示，在中文CWN数据集上，当柱取值较小时，两个评价指标都得到了提升，但继续增加柱的大小甚至会导致结果低于贪心算法。在Oxford数据集上，柱搜索策略带来的提升更明显，但随着柱的增大也会出现指标下降的情况。针对这一现象，Cohen and Beck (2019)指出柱搜索算法的柱取值越大，在解码过程较靠前的时间步会越倾向于选择低概率的词语，对生成效果产生影响，因此一味增加柱的大小并不能带来持续的提升。



(a) CWN数据集　　(b) Oxford数据集

Figure 4: 柱取不同大小的结果对比

4 质量分析

4.1 人工评价

为了更准确地评价生成释义的质量，本文从CWN测试集中随机采样了200条数据，其中被释义词没有重复，然后采用人工评价的方式对基线模型和本文模型的生成释义进行了质量评估。附录A.1中展示了部分用于人工评价的生成释义实例。我们请了四名语言学专业学生作为标注员，使用Likert量表(Likert, 1932)，按照1～5五个等级让标注员分别从语法和语义两个角度对模型的生成释义进行独立评分。其中语法角度仅衡量生成释义是否符合语法规则，完全符合为5分，完全不符合为1分；语义角度衡量生成释义与参考答案表示的语义是否一致，完全一致为5分，完全不一致为1分。表5中展示了四名标注员的人工评价结果。

可以看到，四名标注员对模型生成释义语法的评分都普遍较高，本文模型语法的平均分接近满分，说明模型具备了出色的生成流畅句子的能力。而五个模型在语义上的评分都相对较低，但本文模型的评分还是显著优于基线模型，这与上节中的自动评价结果也保持了一致。

	模型	标注员				平均分
		1	2	3	4	
语法	Transformer	4.985	4.890	3.905	4.760	4.635
	LOG-CaD	4.890	4.390	3.785	4.450	4.379
	BERT-fix-encoder(Greedy)	**5.000**	4.830	4.320	4.840	4.748
	BERT-fine-tune(Greedy)	**5.000**	4.920	4.525	4.905	4.838
	BERT-fine-tune(Beam=2)	**5.000**	**4.930**	**4.615**	**4.915**	**4.865**
语义	Transformer	1.575	1.605	1.815	1.435	1.608
	LOG-CaD	2.425	2.220	2.545	2.000	2.298
	BERT-fix-encoder(Greedy)	2.945	2.740	3.210	2.755	2.913
	BERT-fine-tune(Greedy)	3.315	2.955	3.615	**3.165**	3.263
	BERT-fine-tune(Beam=2)	**3.340**	**3.060**	**3.735**	**3.165**	**3.325**

Table 5: CWN数据集人工评价结果

为了衡量BLEU和语义相似度两个自动评价指标与人工评价指标的相关程度，本文计算了自动评价指标与人工评价指标的Pearson相关系数，如表6所示。可以看到，相比前人使用的BLEU指标，本文额外使用的语义相似度指标与人工评价指标具有更强的相关性。这说明语义相似度指标的结果更接近人类评价结果，更具有参考价值。

人工评价 / 自动评价	语法	语义
BLEU	0.245 ($p < 0.0001$)	0.482 ($p < 0.0001$)
语义相似度	0.298 ($p < 0.0001$)	0.639 ($p < 0.0001$)

Table 6: 自动评价与人工评价指标Pearson相关系数

4.2 数据分布情况对结果的影响分析

对于人类来说，义项越多的词语推断其意思的难度越大，而上下文可以帮助我们对多义词进行消歧，上下文中被释义词的搭配也能够为我们提供更多语义信息，那么上下文在模型中同样可以得到有效利用吗？本节在CWN数据集上，从释义、上下文两项数据内容的不同分布情况出发，对基线模型及本文模型的生成结果进行了对比分析。由于BLEU指标计算时，会将多义词的全部释义都作为参考答案，这会对我们的分析结果产生影响，因此本节选用语义相似度作为衡量指标，对BLEU指标的影响分析见附录A.2。如图5所示，两张子图中分别展示了不同释义数量以及上下文长度对模型语义相似度结果的影响。

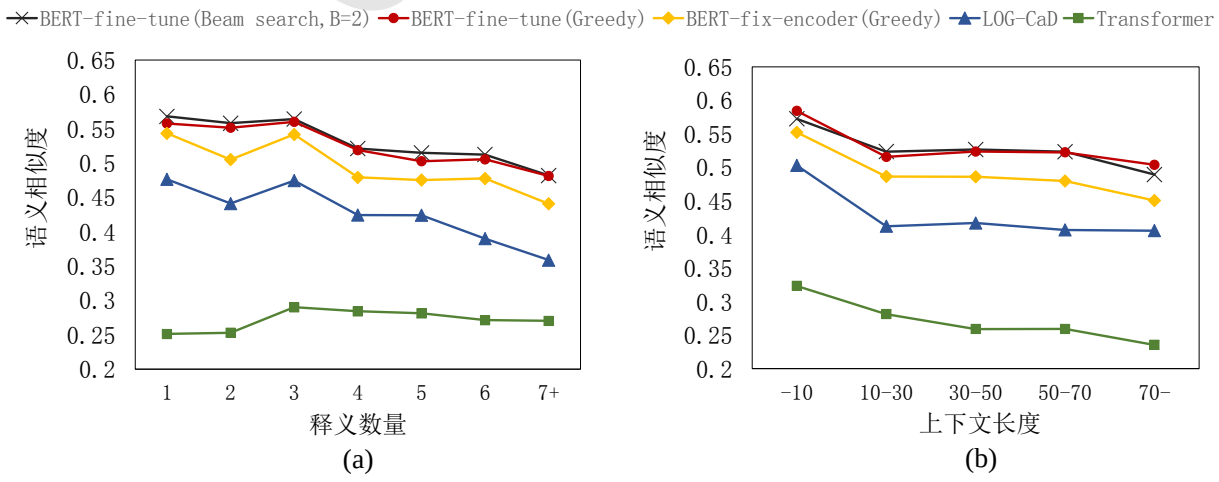


Figure 5: 数据分布情况对语义相似度结果的影响

可以看到，前两个子图中Transformer模型的折线趋势与其他四个模型有明显差异，本文认为这是由于Transformer模型在此任务上本身生成结果较差，由此带来的影响比本节分析的数据

因素要大得多。因此，本节主要对另外四个模型进行对比分析。

图5(a)中释义数量是指被释义词拥有的释义数量。可以看到随着释义数量增加，本文模型和LOG-CaD基线模型的语义相似度指标都呈现下降趋势。但当释义数量超过6条时，对LOG-CaD模型的结果影响显然更大，这可能是由于本文模型和LOG-CaD模型不同的编码方式造成的。本文是将被释义词和上下文同时编码，BERT编码器能够将输入序列编码为一组上下文相关的词向量，更好地捕获上下文信息。而LOG-CaD模型是将上下文和被释义词各自编码后再拼接，当义项数量过多时，此方法可能难以起到很好的消歧效果。

图5(b)中的上下文长度是按字统计的，整体上折线都呈下降趋势。这说明当上下文长度过长时，会对模型在上下文中定位重要信息产生干扰，因此释义生成任务中使用的上下文句子不宜过长。

4.3 生成释义的问题分析

本文从最优模型BERT-fine-tune(Beam=2)在CWN数据集上的生成结果中发现了一些典型问题，并将问题及相应实例分类整理在表7中。

- 第一类问题是模型生成的释义与参考答案的语义刚好相反，这一问题在英文释义生成任务中也会出现(Noraset et al., 2017)，是由于反义词的上下文语境通常极为相似，导致它们的词向量也会非常接近，这是通过大规模语料训练词向量方法的固有问题，这一问题也被转移到了释义生成任务上。
- 第二类问题是由于模型缺乏特定领域知识而导致生成错误释义，这一问题可以通过融入外部知识的方法得以缓解。
- 第三类问题是生成的释义中包含被释义词，本文认为这一问题是否归于错误不应一概而论。例如表7中针对该问题给出的第一个实例的情况是错误的，但对于第二个实例，释义中出现被释义词应该是被允许的。
- 第四类问题是如果被释义词的近义词在训练集中出现过，模型会倾向于生成与该近义词完全相同的释义。这种做法有时可以帮助模型生成完全正确的释义，例如表7中针对此问题给出的第一个实例；但有时由于近义词的语义有细微差别，也会导致生成释义不准确，例如表7中针对此问题给出的第二个实例。

问题一：生成相反释义	
近	参考答案：形容时间的距离短。 生成结果：形容时间的距离长。
问题二：缺乏特定知识	
河南	参考答案：位于黄河南岸的一省。介于湖北省与陕西省之间。 生成结果：中国省名，位于湖北、西藏之间的区域。
问题三：生成释义中包含被释义词	
解释	参考答案：说明特定事件的原因、理由使听话者明白。 生成结果：解释使听话者明白。
箱	参考答案：计算箱装物品的单位。 生成结果：计算箱子的单位。
问题四：生成与训练集中的近义词相同的释义	
聚集（近义词“聚”）	参考答案：多数的前述对象同一时间在同一地点出现。 生成结果：多数的前述对象同一时间在同一地点出现。
施暴（近义词“施虐”）	参考答案：以暴力对待。 生成结果：以不合人道，受事者无法忍受的方式对待。

Table 7: 生成释义的问题及相应实例

5 相关工作

5.1 释义生成任务

释义生成是近年来提出的一项文本生成任务，最初用于验证预训练静态词向量能否捕捉到正确且充分的语法、语义信息，或用于对低维密集词向量包含的语义信息予以解释，后来此任

务的研究目的逐渐落地到辅助语言学习者学习新词汇的实际应用场景。目前对该任务的研究基本都在英文上开展，对于中文释义生成的研究仅有一篇文章公开发表。

Noraset et al. (2017)首次提出了释义生成任务，用于直接评估预训练词向量的质量。文中将任务定义为给定目标词，为其生成相应的一句释义。方法上，除了目标词预训练词向量以外，还使用了CNN模型来提取目标词的字符级语义特征，解码器采用LSTM模型，并通过门控机制在解码的每一个时间步对输入向量进行信息过滤。但这项工作忽略了预训练词向量存在将多义词意义合并的缺陷，此后在英文上的工作基本都使用了上下文信息，让模型生成目标词在特定上下文中的释义。Gadetsky et al. (2018)提出了基于AdaGram对词向量进行消歧的方法。Mickus et al. (2019)提出了Select和Add两种编码机制对目标词及上下文进行编码，突出目标词在上下文序列中的重要性。Ishiwatari et al. (2019)直接将目标词预训练词向量、字符级特征向量和上下文向量拼接起来，作为解码器输入，进一步提升了生成效果。Li et al. (2020)提出将词的含义明确分解为若干个语义成分，并使用离散的潜在变量对语义成分建模后用于释义生成，该模型在英文数据集上取得了当前最优BLEU结果。

还有研究者从低维密集词向量的可解释性问题出发研究释义建模任务。Chang et al. (2018)将给定的目标词及其上下文嵌入高维稀疏空间，然后从中选择最能解释目标词语义的特定维，使用RNN模型生成目标词的文本释义，能够对目标词嵌入包含的语义信息进行直接解释。Chang and Chen (2019)随后又将释义建模任务重新定义为分类任务，即根据目标词及其上下文选择最合理的释义，来研究BERT、ELMO等预训练语言模型的上下文相关词向量捕获了什么语义信息。

释义生成任务在中文上的研究还很少。Yang et al. (2020)等人首次在中文上开展释义生成任务，使用基于Transformer的模型，并将Hownet中的义原序列融入模型，为模型提供更多外部语义知识信息，但这项工作没有考虑上下文信息。基于此，本文首次将上下文信息引入中文释义生成任务。

5.2 预训练语言模型BERT

近年来，面向NLP的预训练技术研究取得了长足进展。早期使用的Word2Vec预训练静态词向量(Mikolov et al., 2013a; Mikolov et al., 2013b)能够为NLP任务带来的提升十分有限，且无法解决一词多义的问题。后来提出的ELMo(Peters et al., 2018)是一种上下文相关的文本表示方法，可有效处理多义词问题。随后，GPT(Radford, 2018)和BERT(Devlin et al., 2018)等预训练语言模型被相继提出。其中BERT是迄今为止应用范围最广、效果最佳的预训练语言模型，在文本分类、语法改错等多项NLP任务中都展示出强大的性能(Adhikari et al., 2019; Kaneko and Komachi, 2019)。BERT是基于Transformer的双向编码表示模型，该模型的预训练使用了掩码语言模型和后句预测两个子任务，模型的优化目标函数是两个子任务目标函数的结合。将预训练后的BERT迁移到文本生成任务中，只需在BERT后增加一个解码器，即可进行微调训练。

本文将预训练语言模型BERT迁移到释义生成任务中，使用BERT初始化编码器的模型参数，使用Transformer作为模型解码器，此方法有效缓解了缺乏训练数据的问题。

6 总结

本文首次将上下文信息应用于中文释义生成任务，为了弥补缺乏训练数据的问题，提出了基于BERT与柱搜索策略的模型。为了验证模型的性能，本文分别在新构建的中文CWN数据集以及前人构建的英文Oxford数据集上开展了实验，结果表明，本文模型相比基线模型能显著提升释义生成的效果。本文还分析了数据分布情况对生成结果的影响，又通过实例分析总结了目前中文释义生成仍存在的四类重要问题。在未来的工作中，我们计划研究是否可以提出一种新的编码机制，更充分地利用多条上下文信息。此项工作使用的完整数据和代码公开于https://github.com/blcuicall/AutoDict/tree/ccl2020。

参考文献

Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and Jimmy Lin. 2019. Docbert: Bert for document classification. *ArXiv*, abs/1904.08398.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors

with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Ting-Yun Chang and Yun-Nung Chen. 2019. What does this word mean? explaining contextualized embeddings with natural language definition. In *EMNLP/IJCNLP*.

Ting-Yun Chang, Ta-Chung Chi, Shang-Chi Tsai, and Yun-Nung Chen. 2018. xsense: Learning sense-separated sparse representations and textual definitions for explainable word sense networks. *ArXiv*, abs/1809.03348.

Eldan Cohen and J. Christopher Beck. 2019. Empirical analysis of beam search performance degradation in neural sequence models. In *ICML*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Artyom Gadetsky, Ilya Yakubovskiy, and Dmitry Vetrov. 2018. Conditional generators of words definitions. In *ACL*.

Shonosuke Ishiwatari, Hiroaki Hayashi, Naoki Yoshinaga, Graham Neubig, Shoetsu Sato, Masashi Toyoda, and Masaru Kitsuregawa. 2019. Learning to describe unknown phrases with local and global contexts. In *NAACL-HLT*.

Masahiro Kaneko and Mamoru Komachi. 2019. Multi-head multi-layer attention to deep language representations for grammatical error detection. *ArXiv*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *ArXiv*, abs/1412.6980.

Shen Li, Zhe Zhao, Renfen Hu, Wensi Li, Tao Liu, and Xiaoyong Du. 2018. Analogical reasoning on chinese morphological and semantic relations. In *ACL*.

Jiahuan Li, Y. Bao, Shujian Huang, X. Dai, and Jiajun Chen. 2020. Explicit semantic decomposition for definition generation. In *ACL*.

Rensis Likert. 1932. A technique for the measurement of attitudes.

Timothee Mickus, Denis Paperno, and Mathieu Constant. 2019. Mark my word: A sequence-to-sequence approach to definition modeling. *ArXiv*, abs/1911.05715.

Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *ArXiv*, abs/1301.3781.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. *ArXiv*, abs/1310.4546.

Thanapon Noraset, Chen Liang, Lawrence Birnbaum, and Doug Downey. 2017. Definition modeling: Learning to define word embeddings in natural language. In *AAAI*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *NAACL*.

Alec Radford. 2018. Improving language understanding by generative pre-training.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *EMNLP*.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. *arXiv*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Liner Yang, Cunliang Kong, Yun Chen, Yang Liu, Qinan Fan, and Erhong Yang. 2020. Incorporating sememes into chinese definition modeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1669–1677.

A 附录

A.1 模型生成释义实例

被释义词：作用 **上下文**：一旦自动增词及印象增强的功能开始作用，这就不是一个可以重复的实验。	
参考答案	特定对象运作并发挥功能。
Transformer	比喻特定事件进行的快慢的程度。
LOG-CaD	特定对象的作用。
BERT-fix-encoder(Greedy)	特定对象在使用时发挥的效能。
BERT-fine-tune(Greedy)	特定对象所能发生的效能，常用于否定句。
BERT-fine-tune(Beam=2)	特定对象在使用时发挥的效能。
被释义词：开阔 **上下文**：真正的享乐，只在心灵的充实，胸襟的开阔	
参考答案	形容心胸宽大。
Transformer	享受物质上所带来的快乐，常隐含负面的涵义。
LOG-CaD	形容比喻思想或知识深刻且没有阻碍。
BERT-fix-encoder(Greedy)	形容比喻度量大，能容忍。
BERT-fine-tune(Greedy)	形容比喻度量大，能容忍。
BERT-fine-tune(Beam=2)	比喻拓展后述知识领域或经验。
被释义词：大打折扣 **上下文**：森林受到大量砍伐，植物吸收二氧化碳的能力大打折扣。	
参考答案	比喻将前述对象的功能或价值大幅度地降低。
Transformer	物体或有机体把组织外部的物质吸到内部。
LOG-CaD	以unk的方式使特定对象离开说话者的距离。
BERT-fix-encoder(Greedy)	形容特定对象的程度降低。
BERT-fine-tune(Greedy)	比喻前述对象的价值降低到负面评价。
BERT-fine-tune(Beam=2)	比喻前述对象的价值降低到负面评价。
被释义词：扯 **上下文**：阳台的墙缘上停着一只麻雀，仰着头、张嘴合嘴，扯着喉咙叫。	
参考答案	喉咙用力以大声量发声。
Transformer	前述物体移动时的力量带动后述物体。
LOG-CaD	用肢体的动作使声音达到最佳状态。
BERT-fix-encoder(Greedy)	使特定对象的毛发变得滑顺且触感好。
BERT-fine-tune(Greedy)	使特定对象的头发变得紧密。
BERT-fine-tune(Beam=2)	使特定对象离开其附着的表面。
被释义词：想要 **上下文**：现在的小朋友想要什么，就有什么。	
参考答案	提出后述需求或条件。
Transformer	表加强语气，强调后述对象。
LOG-CaD	表达成后述事件的愿望。
BERT-fix-encoder(Greedy)	因喜好而想要得到后述对象。
BERT-fine-tune(Greedy)	因分开而思念后述对象。
BERT-fine-tune(Beam=2)	认为符合自己心意而同意做后述事件。

Table 8: 模型生成释义实例

A.2 数据分布情况对BLEU结果的影响

如图6所示，两张子图中分别展示了不同释义数量及上下文长度对模型BLEU结果的影响。

与4.2节类似，(a)子图中Transformer模型折线趋势与其他模型差别较大，这主要是受到模型本身生成效果较差的影响。

从图6(a)中可以看到，不论是基线模型还是本文提出的模型，释义数量小于2条的词的BLEU结果都很不好，这与使用的BLEU指标的计算方式有关。该指标在计算时，会将多义词的全部释义都作为参考答案，因此当释义数量过少时，BLEU值会比较低。当释义数量超过5条时，本文模型和LOG-Cab模型的BLEU值都出现了不同程度的下降，但对LOG-Cab模型的结果影响显然更大。

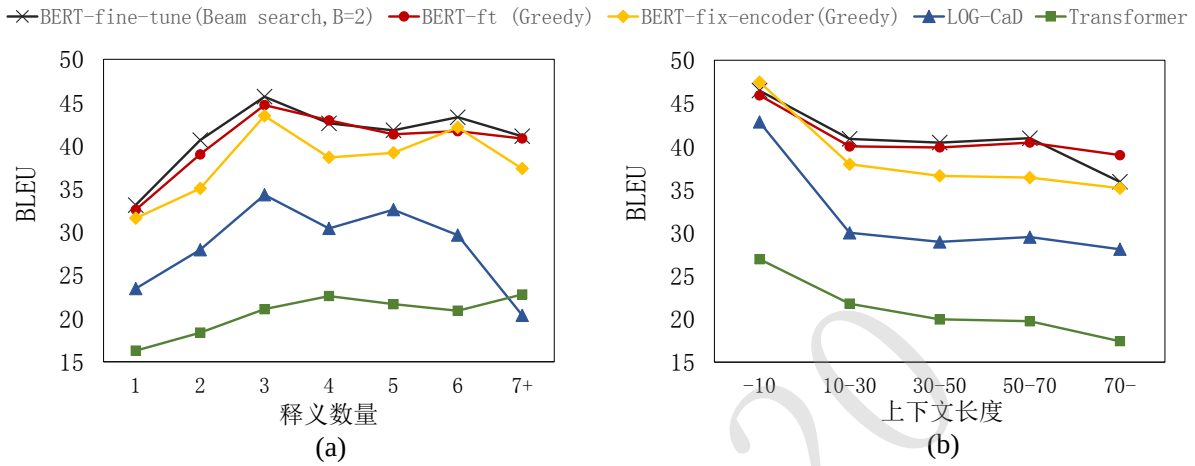


Figure 6: 数据分布情况对BLEU结果的影响

图6(b)中对不同上下文长度的BLEU结果进行了比较。可以看到当上下文长度超过10时，BLEU指标出现非常明显的下降，和对语义相似度指标的影响情况基本一致。

基于深度学习的实体关系抽取研究综述

夏振涛[1,3]**, 曲维光**[1]**, 顾彦慧**[1]**, 周俊生**[1]**, 李斌**[2]
(1.南京师范大学 计算机科学与技术学院，江苏省 南京市 210023;
2.南京师范大学 文学院，江苏省 南京市 210097;
3.南京擎盾信息科技有限公司，江苏省 南京市 210000)

摘要

作为信息抽取的一项核心子任务，实体关系抽取对于知识图谱、智能问答、语义搜索等自然语言处理应用都十分重要。关系抽取在于从非结构化文本中自动地识别实体之间具有的某种语义关系。该文聚焦句子级别的关系抽取研究，介绍用于关系抽取的主要数据集并对现有的技术作了阐述，主要分为：有监督的关系抽取、远程监督的关系抽取和实体关系联合抽取。我们对比用于该任务的各种模型，分析它们的贡献与缺陷。最后介绍中文实体关系抽取的研究现状和方法。

关键词： 关系抽取 ； 有监督方法 ； 远程监督方法 ； 实体关系联合抽取

Review of Entity Relation Extraction based on deep learning

XIA zhentao [1,3]**, QU Weiguang**[1]**,**
GU Yanhui[1]**, ZHOU Junsheng**[1]**, LI Bin**[2]
(1.School of Computer Science and Technology, Nanjing Normal University,
Nanjing , Jiangsu 210023, China;
2.School of Chinese Language and
Literature, Nanjing Normal University, Nanjing , Jiangsu 210097, China;
3.Aegis Data, Nanjing, Jiangsu 210000, China)

Abstract

As a core task of Information Extraction, Entity Relation Extraction plays an important role in many Natural Language Processing applications such as knowledge graph, intelligent question answering system and semantic search. Relation extraction tasks aim to find the semantic relation between a pair of entity mentions from unstructured texts. This paper focuses on the sentence-level relation extraction, introduces the main datasets for this task, and expounds the current status of relation extraction technology which can be divided into: supervised relation extraction, distant supervision relation extraction and joint extraction of entities and relations. We compare the various models for this task and analyze their contributions and defects. Finally, the research status and methods of Chinese entity relation extraction are introduced.

Keywords: relation extraction , supervised method , distant supervision method , joint extraction of entities and relations

基金项目：国家自然科学基金“汉语抽象意义表示关键技术研究”(61772278)；江苏省高校哲学社会科学基金“面向机器学习的汉语复句语料建设研究”(2019JSA0220);国家社会科学基金“中文抽象语义库的构建及自动分析研究”（18BYY127）。

1 引言

信息抽取是自然语言处理的一项重要任务，它的基本目的之一是从原始的非结构化文本中提取有意义的结构化信息，以用于智能问答、检索等自然语言处理应用。信息抽取本身是一项庞大的任务，包括命名实体识别、关系抽取、事件抽取等子任务。本文，我们聚焦关系抽取任务。

本文将关系定义为两个或多个实体之间的某种联系，关系抽取就是自动识别实体之间具有的某种语义关系。本文所研究的是二元关系抽取，关注两个实体之间的语义关系，得到(subject:主语，relation:关系，object:宾语)三元组，其中subject和object表示两个实体，relation表示实体之间的语义关系。例如，给出如下的句子：刘翔，1983年7月13日出生于上海，中国男子田径队110米栏运行员。我们可以抽取出实体关系三元组实例(刘翔，出生地，上海)。

这里我们主要研究基于深度学习的关系抽取方法，包括：有监督关系抽取、远程监督关系抽取和实体关系的联合抽取。基于传统方法的关系抽取研究综述可参考文献 (Pawar et al., 2017)。此外我们还探讨中文关系抽取的研究现状、方法等。

本文的结构如下：第1节介绍用于关系抽取的数据集；第2,3,4节分别探讨有监督关系抽取、远程监督关系抽取和实体关系联合抽取相关方法研究及其优缺点；第5节探讨中文实体关系抽取的研究现状和方法；第6节对本文进行总结并展望关系抽取未来的研究方向。

2 数据集

2.1 有监督数据集

有监督的数据集需要人工标注，意味着数据包含高质量的实体关系三元组，几乎没有噪音。但是人工标注数据集耗时耗力，因此这些数据集通常规模较小。下面介绍的两个数据集中，每个句子中的命名实体进行了标记并且实体对表达的关系可以被预测。

ACE 2005数据集： 自动内容抽取数据集(ACE)包含599个与新闻和电子邮件相关的文档，并划分为7种主要的关系类型。其中，6种主要关系类型包含足够的实例，每种关系类型平均有700个实例用于训练和测试。

SemEval-2010 Task 8 数据集： 该数据集 (Hendrickx et al., 2009)包含10717个样本，其中8000个用于训练，2717个用于测试。数据集中包含9种有序关系类型。关系的方向性实际上使关系的数量加倍，因为只有在顺序也是正确的情况下，才认为实体对被正确标记。最终数据集有19种关系类别(其他类型为2*9+1种)。

2.2 远程监督数据集

为了避免手工构建用于关系抽取数据集的繁重任务，远程监督方法 (Mintz et al., 2009)将文档与已知的知识库对齐，用于自动生成大量训练数据。这种方法基于距离监督假设：如果知识库中的实体对之间存在关系，那么每个包含该实体对的文档都会表达该关系。下面介绍的数据集是基于远程监督方法构造的。

New York Time(NYT)数据集： 远程监督假设是一个强假设并且会导致错误标签问题，即提到两个实体的句子不一定表达他们在知识库中的关系。为了缓解这一问题，降低噪声，可以将问题建模为一个多示例学习问题，放宽远程监督假设 (Riedel et al., 2010)我们在后续章节详细介绍多示例学习方法。该数据集通过将Freebase与纽约时报语料库(NYT)的关系对齐而形成。数据集包含53种可能的关系类别，包括一个特殊的关系类别NA(表示实体之间没有关系)。训练数据包含522611个句子，281270个实体对和18252个关系。

3 有监督的关系抽取方法研究

有监督的关系抽取方法被认为是一个多分类问题，通常分为基于特征工程的方法、基于核函数的方法和基于神经网络的方法。基于特征工程工程方法，许多研究的主要工作是如何能抽取复杂的特征。基于核函数的方法，直接以原始字符串作为处理对象，计算两个对象的核函数。利用神经网络可以自动从文本中获取表征关系的特征，完成关系抽取。本节中，我们主要介绍神经网络用于关系抽取的方法。

3.1 基于神经网络的方法

利用神经网络进行关系抽取，主要分为三个步骤：特征表示、神经网络的构建与学习和模型分类。其中，特征表示是将文本表示成分布式特征信息，比如将文本中的单词映射为为分布式词向量 (Collobert et al., 2008; Mikolov et al., 2013(a); Pennington et al., 2014; Mikolov et al., 2013(b))，这样就避免了传统特征工程中特征向量的稀疏性问题。我们把现有的研究方法总结为如下三类：融合词汇特征的方法、基于最短依存路径的方法和基于注意力机制的方法。

3.1.1 融合词汇特征的神经网络模型

MV-RNN (Socher et al., 2012)是最早的利用神经网络进行关系抽取的方法,该方法沿着成分解析树建立递归神经网络学习树结构中单词或短语的语义向量表示。树中每个节点由矩阵-向量(MV)表示，向量用来表示成分的内在含义，矩阵用来捕获它如何改变相邻单词或短语的含义。对于关系分类任务，该方法首先在解析树中找到要对其关系进行分类的两个实体之间的路径。然后选择路径的最高节点，并使用该节点的向量作为特征对关系进行分类，同时他们在模型中加入词性特征、命名实体标签特征和WordNet语义类别特征。该方法在SemEval-2010数据集上F1值达到82.4%。

MV-RNN包含特征学习过程，这种方法依赖递归过程中使用的语法树，句法分析中的错误会抑制学习高质量特征的能力。因此，现有方法利用卷积神经网络(CNN) (Zeng et al., 2014; Kim, 2014; Collobert et al., 2011)提取词汇和句子级层次特征，用于关系抽取任务。该方法不需要复杂的句法或语义预处理，模型的输入是一个带有两个标记实体名词的句子。然后通过词嵌入将单词映射为分布式词向量，分别提取词汇级特征和句子级特征，并直接将两种特征拼接形成最终的特征向量。其中，将带标记实体及其上下文词语所对应的词向量和WordNet中语义类别特征拼接作为词汇级特征向量,用最大池化卷积神经网络(CNN)自动提取句子级特征表示。更进一步，CR-CNN模型 (Santos et al., 2015)利用新的卷积神经网络处理关系分类任务,对于给定的输入文本，模型将其转换成分布式表示，然后通过卷积层构造句子的特征表示r，不同于Zeng使用softmax计算得分，该方法模型通过类别权重矩阵和r执行点积操作得到每个关系类别的得分，并且采用排名损失函数进行训练。该工作只使用词向量作为特征输入，没有使用任何其他外部资源。

3.1.2 基于最短依存路径的神经网络模型

上述方法在关系抽取任务上是有效的，但是，当主语和宾语之间的距离较长时，往往会受到其他不相关信息的干扰。考虑到句法特征在关系识别中起到至关重要的作用，许多研究通过神经网络从最短依存路径中学习关系表示。该方法基于最短路径假设：如果e1和e2是句子中的两个名词实体，我们假设e1和e2之间的最短路径，描述了它们之间的关系。这是因为,(1)如果e1和e2是属于同一个谓词的论元，那么它们的最短路径应该通过该谓词；(2)如果e1和e2属于不同的谓词论元结构，它们的最短路径将通过一系列谓词，并且任何连续的谓词都共享一个公共论元。路径上谓词的顺序表明了该关系的主语和宾语的正确分配。

基于最短路径假设，通过卷积神经网络可以从最短依存路径学习关系表示 (Xu et al., 2015(a)),该模型采用主语到宾语的最短路径作为输入，依存路径上的每个节点通过词嵌入矩阵生成局部特征向量，然后将这些特征向量组合，用CNN网络获取全局特征向量。同时，为了处理关系的方向性(即在关系中正确区分主语和宾语)，该方法在模型中引入负采样策略，采用宾语到主语的最短依存路径为负样本，研究表明负采样策略显著提升了模型的性能。

同年，基于SDP-LSTM神经网络方法 (Xu et al., 2015(b))用于关系分类，该方法利用两个实体之间的最短路径(SDP)，以LSTM为单元的多通道(四种信息：单词、词性、语法关系、WordNet语义特征)循环神经网络沿SDP提取多样化特征，最大池化层从每条路径中的LSTM节点收集信息；为了缓解过拟合，模型采用了dropout正则化策略。

然而，最短依存路径(SDP)是一种特殊的结构，在这个结构中，每两个相邻单词由依存关系分隔开。由于卷积神经网络和循环神经网络的局限性，以往的研究将依存关系等同于单词或词性等句法特征。循环卷积神经网络(RCNN) (Cai et al., 2016)利用基于LSTM的双通道循环神经网络对SDP中的全局模式进行编码，并利用卷积层捕获依存关系中每两个相邻单词的局部特征。基于RCNN模型，他们提出一种双向循环卷积神经网络(BRCNN)，可以同时学习沿着SDP前向和后向的双向信息表示，增强了实体间关系方向分类的能力。

神经网络模型直接在解析树上操作很难并行化，计算效率很低。基于主语和宾语之间的最短路径模型虽然在计算上更高效，但这种假设有很大的局限性，会丢掉一些关键信息。受到图卷积神经网络 (Marcheggiani et al., 2017; Kipf et al., 2016)的启发，一种适用于关系抽取的图卷积神经网络的扩展方法 (Zhang et al., 2018)利用高效的图卷积运算对输入语句的依存结构进行编码，然后提取以实体为中心的表示，实现关系预测。此外基于注意力引导的图卷积网络方法(AGGCNs) (Guo et al., 2019)提出用于关系抽取。该模型由M个相同的块组成，每个块包含注意力指导层、密集连接层和线性组合层。模型直接把整棵依存树作为输入，每个块以表示图的节点嵌入和邻接矩阵作为输入，在注意力引导层中，先将原始的图转化为邻接矩阵，然后通过多头注意力转化为全连接的基于注意力指导的邻接矩阵，矩阵中的每个元素对应相应节点之间边的权重，从而捕捉到领域的信息。密集连接层得到的矩阵被送入N个单独的密集连接层，产生新的表示。最后，应用线性组合将N个紧密连接的层的输出组合成隐藏的表示。经过上述注意力引导的图卷积模型，得到所有tokens的表征，然后将句子的表征和实体表征合并，运用前馈神经网络得到最终表示，最后用逻辑回归分类器预测关系。

3.1.3 基于注意力机制的神经网络模型

通过神经网络对依存树建模能提高关系抽取的性能，但这类方法还是需要依赖词汇资源如WordNet，或自然语言处理工具来提取特征。并且，对于关系重要的信息可以出现在句子的任意位置。为了解决这个问题，基于注意力机制的双向长短期记忆网络(AttBLSTM) (Zhou et al., 2016)用来捕获句子中重要的语义信息。该方法的注意力机制计算公式为：

$$M = tanh(H) \tag{1}$$

$$\alpha = softmax(W^T M) \tag{2}$$

$$r = H\alpha^T \tag{3}$$

其中，H为双向LSTM输出层组成的矩阵，w是学习的参数，r是输出向量的加权求和。最终的句子可以表示为

$$h^* = tanh(r) \tag{4}$$

模型用一个softmax分类器得到关系类别。实验表明，该方法不使用任何外部资源，可以得到很好的性能。

现有的基于注意力机制的关系抽取方法并没有充分利用实体信息，而实体信息可能是关系分类的最关键特征。因此，一种融合潜在实体类型的实体注意力机制模型 (Lee et al., 2019)用于关系抽取。在该方法中，为了捕获句子上下文信息，利用了自注意力机制(self-attention) (Tan et al., 2018; Vaswani et al., 2017)获得单词表示，并利用双向LSTM构建了神经网络模型。为了充分利用句子中实体对的信息，模型融合了实体相对位置特征和实体潜在种类特征，最终句子的表示可以通过注意力机制得到：

$$u_i = tanh(W^H[h_i, p_i^{e_1}, p_i^{e_2}] + W^E[h_{e_1}, t_1, h_{e_2}, t_2]) \tag{5}$$

$$\alpha_i = \frac{exp(v^T u_i)}{\sum_{j=1}^{n} exp(v^T u_j)} \tag{6}$$

$$z = \sum_{i=1}^{n} \alpha_i h_i \tag{7}$$

其中，h_i为第i位置双向LSTM的输出，h_{e_1};h_{e_2} 分别为实体e_1;e_2位置双向LSTM的输出，$p_i^{e_1}$;，$p_i^{e_2}$分别为对应于第i个单词相对于句子中的第一个实体e_1和第二个实体e_2的位置。该方法还使用实体类型信息提升性能，用于实体类型没有标注，他们使用主题聚类方法得到实体的潜在种类，通过注意力机制得到实体类型表示t。

3.2 小结

本章详细讨论了有监督方法的关系抽取，给出了现有的一些经典方法。目前，神经网络用于有监督的关系抽取成为主流方法，我们详细讨论了，神经网络方法中融合词汇特征的方法、基于最短依存路径的方法和基于注意力机制的方法。表1给出现有方法用于有监督关系抽取在SemEval-2010数据集上的性能对比。从图中可以看出，基于最短依存路径的方法总体上达到了较好的性能，原因在于利用最短依存路径对句子建模，可以丰富句子的全局语义信息，显著帮助关系分类。而融合词汇的方法，只能利用句子的局部特征，关系分类性能较低。基于注意力机制的方法不采用任何外部特征资源，利用注意力机制的可解释性自动挖掘出句子中的重要语义信息。

方法分类	模型	特征	F1值
融合词汇特征方法	MV-RNN (Socher et al., 2012)	Word embeddings+POS,NER, WordNet	82.4%
	CNN (Zeng et al., 2014)	Word embeddings + word position embeddings, WordNet	82.7%
	CR-CNN (Santos et al., 2015)	Word embeddings + word position embeddings	84.1%
基于最短依存路径方法	depLCNN (Xu et al., 2015(a))	WordNet, words around nominals	85.6%
	SDP-LSTM (Xu et al., 2015(b))	Word Embeddings, POS embeddings, WordNet embeddings, grammar relation embeddings	83.7%
	BRCNN (Cai et al., 2016)	Word embeddings+POS, NER, WordNet embeddings	86.3%
	C-GCN (Zhang et al., 2018)	Word embeddings	84.8%
	AGGCNs (Guo et al., 2019)	Word embeddings	85.7%
基于注意力机制方法	AttBLSTM (Zhou et al., 2016)	Word embeddings, position embeddings	84.0%
	BiLSTM with Entity-aware attention (Lee et al., 2019)	Word emneddings, Latent entity Typing	85.2%

Table 1: 有监督关系抽取方法在数据集SemEval-2010对比

4 基于远程监督的关系抽取方法研究

有监督的关系抽取需要依赖人工标注的数据集，限制了该方法的适用领域。因此，远程监督方法 (Mintz et al., 2009)将文档与已知的知识库对齐，用于自动生成大量训练数据。然而，远程监督假设是一个强假设并且会导致错误标签问题，即提到两个实体的句子不一定表达他们在知识库中的关系。因此可以将远程监督关系抽取任务作为一个多示例学习问题来放宽假设 (Riedel et al., 2010)。在用于关系抽取的多示例学习中，知识库(KB)中的每个实体对标记一个句子包。包中的所有句子都包含实体对的提及，但它们不一定包含直接关系。多示例学习是对包标签预测，而不是为每个句子预测关系标签。它假定，如果实体对存在关系，则包中至少有一个示例反映给定实体对的关系。

4.1 基于卷积神经网络的远程监督方法

Riedel的方法抽取文本特征时，需要依赖自然语言处理工具，会造成错误传播问题。因此，分段卷积网络(PCNN) (Zeng et al., 2015)用来提取特征，并利用多示例学习方法缓解数据

噪音问题。在多示例训练中把目标函数定义在包上，首先对包中的每个示例分别预测，得到相应的关系概率，然后选取概率最大的示例标签作为包的标签，并利用包的标签更新网络参数。

PCNN在远程监督数据集上得到了不错的效果，但这种方法仍然有缺陷。首先，PCNN将远程监督关系抽取看作一个单标签学习问题，并为每个实体对选择一个关系标签，而忽略了同一个实体对可能存在多个关系的事实。针对这个问题，利用两个不同的损失函数， (Jiang et al., 2016)处理多标签分类问题。此外，PCNN基于Riedel提出的假设来生成标记数据,根据这一假设，PCNN在训练和预测中只选择每个实体对可能的句子。然而，选择一个句子会丢失包含其他句子中的信息。对于这个问题，假设“两个实体间的关系可以自动从提到这两个实体的所有句子中显示表达或隐式推断” (Jiang et al., 2016)，在使用卷积神经网络自动提取每个句子的特征后，他们使用跨句最大池化来选择不同句子的特征，然后将最重要的跨征聚合为每个实体对的表示。由于结果表示有不同句子的特征组成，因此该方法充分利用这些句子中包含的所有可用信息。此外，利用句子级别的注意力机制 (Lin et al., 2016)来自动捕获不同句子的重要程度，过滤噪声句子。

上述方法使用的神经网络，模型大多是相对较浅的卷积神经网络，通常只涉及一个卷积层和一个全连接层，而且不清楚更深的模型结构是否能够从噪声数据中提取信号。一种基于残差学习的卷积神经网络。 (Huang et al., 2017)用于关系抽取，他们将词嵌入和位置嵌入合并到一个深度残差网络中，通过恒等映射到卷积层中。实验表明，该方法利用9层带残差学习的卷积神经网络可以显著提升远程监督关系抽取性能。

4.2 基于注意力机制的远程监督方法

现有方法在选择有效示例和缺乏实体背景知识方面存在缺陷，一个基于PCNN的句子级注意力机制模型(APCNNs) (Ji et al., 2017)用来选择有效示例，该模型充分利用了知识库中的监督信息。他们从Freebase和Wikipedia页面中提取实体描述来补充背景知识。对于一个包，模型首先使用PCNNs提取每个句子的特征向量v_{sen}。受到TransE模型的启发，在TransE模型中，用$e_1 + r \approx e_2$对一个三元组$r(e_1 1, e_2)$建模，在APCNNs中，用$(e_1 - e_2)$表示句子中e_1和e_2之间的关系。然后，模型通过一个隐含层用串联$[v_{sen}; e_1 - e_2]$的方式计算每个句子的注意力权重。最后，所有句子特征向量的加权求和就是包的特征。此外，为了将更多的背景知识融入到模型中，该方法使用卷积神经网络来提取实体描述的特征向量。

使用注意力机制，是一种通过学习多个示例的权值分布来选择有效示例的方法。但是，基于深度神经网络的远程监督学习中存在两个重要的表示学习问题：(1)在一个示例中，目标实体对上下文表示学习问题；(2)多个示例的有效示例选择表示学习。在先前的研究工作中，通常采用1-D向量的单词级和句子级注意力机制。1-D注意力向量的缺陷是，它只关注句子中一个或少量的方面，或一个或少量的示例。其结果是不同语义方面的句子，或者不同的多个有效句子被忽略。受结构化自注意力句子嵌入 (Lin et al., 2017)的启发，一种新的基于双向LSTM的多层结构化自注意力机制模型(MLSSA) (Du et al., 2018)用于缓解上述两个问题。针对第一个问题，他们提出一个基于二维矩阵的单词级注意力机制，该机制包含多个向量，每个向量都聚焦于句子的不同方面，从而更好地学习上下文表示。针对第二个问题，他们提出一种用于多示例学习的二维句子级注意力机制，其中包含多个向量，每个向量都集中在不同的有效示例上，以更好地选择句子。

4.3 融合知识库的方法

为了缓解远程监督中错误标注问题，许多研究利用现有知识库添加信息。首先，一种无标签的远程监督方法 (Wang et al., 2018)在距离假设不充分的条件下，不使用关系标签，只利用知识库(KG)的先验知识直接、柔和的监督分类器的学习。除了关系示例外，知识库中还包括其他相关信息，比如关系的别名,现有的关系抽取模型通常忽略这些可用的信息。一种远程监督关系抽取方法—RESIDE (Vashishth et al., 2018)利用知识库中附加的边信息改进关系抽取。具体的，它使用实体类型和关系别名信息在预测关系时施加软约束，使用图卷积神经网络从文本中编码语法信息，即使在有限额外信息可用时也能提高性能。

远程监督可以自动标注足够数量的训练数据;然而，这些数据通常只覆盖关系的有限部分。许多关系都是长尾关系，数据仍然不足。目前的远程监督模型忽略了长尾关系问题，难以从纯文本中提取出全面的信息。受在尾部的数据和在顶部的数据之间丰富的语义关联的启发，一种用于长尾不平衡数据的远程监督关系提取方法 (Zhang et al., 2019)利用分布顶部数据丰富的类

的知识来提高尾部数据贫乏类的性能。首先，他们提出利用知识图嵌入的类标签间的隐式关系知识，利用图卷积网络学习显式关系知识。其次，通过粗到细的知识感知注意机制，将关联知识集成到关联抽取模型中。

4.4 小结

在本节中，我们详细讨论了一些用于远程监督关系抽取的经典方法，下面表2给出这些方法在NYT数据集上抽取关系示例前100(TOP-100)、前200(Top-200)、前300(TOP-300)前500(Top-500)的对比结果,这里使用Precision@N(P@N)为评估指标。远程监督关系抽取只需要手动标注少量的关系实例，适用于没有标注语料库的关系抽取，但其实现过程在数据集中引入了噪声，使得该方法的性能低于有监督的关系抽取方法。许多后续的工作都试图利用选择性注意力机制、融合知识库等方法来处理噪声和放宽远程监督假设，通过去噪进一步提高性能。远程监督关系抽取方法利用的是弱标注数据，一般的神经网络方法都是以数据驱动模型，但纯数据驱动模型并不能充分挖掘数据中的潜在信息，从表中我们可以发现，融合知识库的方法相对有着姣好的性能，这类方法能够更好地将结构化知识融入神经网络模型中，用知识指导模型。融合知识库的方法，不仅仅是关系抽取任务，在其他自然语言处理任务中也具有重要意义。

方法分类	模型	Top-100	Top-200	Top-300	Top-500
基于卷积神经网络方法	PCNN (Zeng et al., 2015)	72.3	69.7	64.1	-
	PCNN+MIL (Zeng et al., 2015)	86.0	80.0	-	69.0
	PCNN+ATT (Lin et al., 2016)	76.2	73.1	67.4	-
	MIMLCNN (Jiang et al., 2016)	69.0	64.0	59.0	53.0
	ResCNN-9 (Huang et al., 2017)	79.0	69.0	61.0	-
基于注意力机制的方法	APCNN+D (Ji et al., 2017)	87.0	83.0	-	74.0
	MLSSA (Du et al., 2018)	90.0	81.5	77.0	-
融合知识库的方法	LFDS (Wang et al., 2018)	90.0	88.0	-	83.0
	RESIDE (Vashishth et al., 2018)	84.0	78.5	75.6	-

Table 2: 远程监督关系抽取方法在数据集NYT数据集对比

5 实体和关系联合抽取方法研究

实体和关系联合抽取在于从非结构化文本中同时进行实体识别和关系抽取。传统方法以流水线方式处理抽取实体关系三元组任务，即首先提取实体，然后识别他们之间的关系。这个独立的框架使任务易于处理，并且每个组件都可以更加灵活。但它忽略了这两个子任务之间的相关性，在这种方式下，每个子任务都是独立的模型。这样，实体识别的结果可能会影响关系分类的性能，导致错误传递。与流水线方法不同，联合学习框架能利用单个模型提取实体和关系，能够有效地集成实体和关系的信息。

5.1 基于共享参数的联合抽取方法

最早的联合框架模型 (Li et al., 2014)利用结构化感知机和集束搜索方式同时提取实体及其关系。该框架采用了一种基于半马尔科夫链思想的分段解码器，克服了传统的基于字符的标注方式。此外，考虑到不精确的搜索，他们提出一些新的有效的全局特征作为约束来捕捉实体和关系之间的相互依赖性。

在有监督的关系分类任务中，可以用基于LSTM的神经网络来表示实体之间的关系，但这些方法只使用有限的语言学结构，并且不能对实体和关系联合建模。因此，一种端到端的模型 (Miwa et al., 2016)用于提取词序列和依存树结构上实体之间的关系。该方法通过使用双向顺序(从左到右和从右到左)和双向树形结(自底向上和自顶向下)的LSTM-RNNs在单个模型中对实体和关系联合建模。与传统的增量端到端关系抽取模型不同，该模型在训练中做了两个改进：实体预训练(对实体模型进行预训练)和scheduled sampling(以一定的概率用正确的标签替换不可靠的预测标签)，这些改进缓解了早期训练阶段实体识别性能低的问题，并且允许实体信息进一步帮助下游关系分类。Miwa提出的模型是局部训练的，没有考虑到增量决策之间的结构性对应。因此，一个全局优化的神经网络模型 (Zhang et al., 2017)用于端到端关系抽取，为了更好地学习上下文表示，提出了新的LSTM特征。Miwa的方法依靠外部句法解析器获取句法信息，这对于关系抽取至关重要，但解析错误可能导致树LSTM的编码不准确，从而降低关系抽取性能。在Zhang的方法中，使用双仿射注意力解析器 (Dozat et al., 2016)的LSTM隐藏层来增强输入表示。由于解析器是预训练的，它包含了关于每个单词的丰富语法信息，但不显式地表示解析决策，从而避免了由于不正确的解析而导致的问题。此外，Miwa的方法在预测实体边界或做出关系分类决策时没有明确学习片段的表示，在Zhang的方法中，采用LSTM-Minus方式，将一个片段建模为最后一个和第一个LSTM隐藏层向量之差。

为了不使用任何依存树信息，Katiyar等人 (Katiyar et al., 2017; Zheng et al., 2017)用多层双向LSTM对句子建模，对实体识别和关系抽取都看作是序列化标注任务。在关系抽取中，对于每个词，使用指针网络找到当前词和相关之前词的关系类型。同样，Zheng把实体和关系抽取任务看作是序列标注任务，与Katiyar方法不同，他们提出一种新的序列化标注方式，用双向LSTM和单向LSTM分别编码和解码，输出层同时对实体和关系标注，完成关系抽取任务。

5.2 基于全局优化的联合抽取方法

上述联合学习的方法是通过共享参数的方式实现，这样的好处是不需要在两个子任务上附加约束。但是由于独立的子模型解码器，子模型之间的联系没有得到充分利用。一个联合最小化风险训练的方法用于实体和关系联合抽取 (Sun et al., 2018)。在这个模型中，把实体识别作为序列化标注任务，关系抽取作为分类任务。两个任务的模型之间共享参数，并且优化一个全局损失函数，弥补了训练和测试之间的差异。通过共享参数联合学习，一方面，在实体类型和关系类型判定时没有显示交互，一些复杂的解码算法可以同时判断实体边界和类型，但是通过在ACE05数据集上的实验发现 (Sun et al., 2019)，边界的识别正确率很高，相对的类型的判定就低一些。因此，将实体关系联合抽取分为两个子任务，分别是实体范围检测（Entity Span Detection）和实体关系类型推导（Entity Relation Type Deduction）。在实体范围检测中使用序列标注的方法，在实体关系类型推导中使用一种基于图卷积网络的联合模型，同时两个模型进行联合训练。

在句子中，实体三元组会存在重叠问题，Zeng等人 (Zeng et al., 2018)首先在神经网络建模中利用拷贝机制 (Gu et al., 2016; He et al., 2017)解决重叠问题。他们把重叠问题分为三种类型：Normal,EntityPairOverlap(SEO),SingleEntityOverlap(SEO)。他们的方法基于sequence-to-sequence模型 (Dong et al., 2016)，在模型中有两个主要部分：编码器(endcoder)和解码器(decoder)。编码器首先把句子转换成固定长度的向量。然后，解码器读取这个向量并生成三元组。

实验表明Zeng的方法对训练数据依赖性强，不能提取多个单词实体的情况。一个层次化的强化学习框架 (Takanobu et al., 2019)通过一个高层强化学习过程识别关系指示词，用低层的强化学习过程识别实体。高层的过程在某个特定位置检测关系指示词，如果确定了某个关系，将触发低层过程识别该关系对应的实体。当低层任务完成后，高层强化学习过程继续搜索句子中下一个关系。Li等人 (Li et al., 2019)将实体关系抽取任务转换为多轮问答问题，即将实体和关系的提取转换为从上下文确定答案的任务。这种方法提供了一个比较好的捕捉标签层次依赖的方法。但这中间方法计算效率低，因为它需要在单个句子中扫描所有实体模板问题和相关的关系模板问题。

此外一种用于实体识别和关系提取的端到端联合模型（(GraphRel)(Fu et al., 2019)通过关系加权GCN来考虑命名实体和关系之间的交互来解实体重叠问题。GraphRel通过堆叠Bi-LSTM句子编码器和GCN依赖树编码器来学习自动提取每个单词的隐藏特征。然后GraphRel标记实体提及单词并预测连接提及的关系三元组，这是第一阶段预测。为了在考虑到三元组之间

的相互作用的情况下进行预测，该模型在GraphRel第二阶段添加了一个新颖的关系加权GCN。第一阶段GraphRel接收到实体损失和关系损失，沿着依赖关系链接提取节点隐藏特征，同时建立具有关系加权边的新全连接图。然后，通过对中间图进行操作，第二阶段GCN在最终分类每个边之前有效地考虑实体之间的相互作用以及（可能重叠的）关系。

5.3 小结

在本节中，我们详细讨论了一些用于实体关系联合抽取的经典方法，表3给出实体和关系联合抽取方法在不同数据集上的结果对比。许多较早的实体关系抽取都采用了流水线框架，流水线框架具有集成不同数据源和学习算法的灵活性，但其缺点也很明显。首先，他们受到错误传播的严重影响，实体提取阶段的错误会传播到关系分类阶段。其次忽略了实体提取和关系分类的相关性。第三，流水线框架导致计算效率低下。在实体提取阶段后，将每个实体对传递到关系分类模型，以识别它们之间的关系。由于大多数实体对没有关系，这中两阶段的方式是低效的。在本节中讨论的联合建模技术采用了实体识别和关系识别任务之间的双向信息流，很好的解决了流水线方式的缺陷。从实际的角度来看，联合抽取方法非常重要，因为良好的实体提取性能是实现良好关系提取性能的必要条件。

datasets	model	Entity			Relation		
		P	R	F1	P	R	F1
ACE04	(Li et al., 2014)	0.835	0.762	0.797	0.608	0.361	0.453
	(Miwa et al., 2016)	0.808	0.829	0.818	0.487	0.481	0.484
	(Katiyar et al., 2017)	0.812	0.781	0.796	0.502	0.488	0.493
	(Li et al., 2019)	0.844	0.829	0.836	0.501	0.487	0.494
ACE05	(Li et al., 2014)	0.852	0.769	0.808	0.654	0.398	0.495
	(Miwa et al., 2016)	0.852	0.769	0.808	0.572	0.540	0.556
	(Zhang et al., 2017)	-		0.836	-		0.575
	(Katiyar et al., 2017)	0.840	0.813	0.831	0.605	0.553	0.578
	(Sun et al., 2018)	0.839	0.832	0.836	0.649	0.551	0.596
	(Sun et al., 2019)	0.861	0.824	0.842	0.681	0.523	0.591
CONLL04	(Zhang et al., 2017)	-		0.856	-		0.678
NYT	(Zheng et al., 2017)	0.59	0.479	0.529	0.597	0.451	0.514
	(Sun et al., 2018)	-			0.652	0.406	0.500
	(Zeng et al., 2018)				0.610	0.566	0.587
	(Takanobu et al., 2019)				0.714	0.586	0.644
	(Fu et al., 2019)				0.639	0.600	0.619

Table 3: 实体关系联合抽取在不同据集的结果

6 中文实体关系抽取研究现状

在中文研究方面，由于标注语料的短缺，关系抽取的研究相对于英文数据集上的研究较少。本文聚焦神经网络的中文实体关系抽取研究，传统方法可参考 (武文雅 et al., 2018)的工作，远程监督方法可参考 (白龙 et al., 2019)的工作。

6.1 中文实体关系抽取数据集

COAE2016： 该数据集来源于第八届中文倾向性分析评测（COAE2016）的面向知识抽取的关系分类任务。该数据集包含988句训练集、483句测试集，以及10种关系类型(人物的出生日期, 人物的出生地, 人物的毕业院校, 人物的配偶, 人物的子女, 组织机构的高管, 组织机构的员工数, 组织机构的创始人, 组织机构的成立时间, 组织机构的总部地点)

ACE2005： ACE 2005数据集收集自新闻专线、广播和网络日志。关系分为 6 大类和18个小类，包含8023个关系事实和18个关系子类型。

DuIE： 2019年，中国计算机学会、中国中文信息学会联合百度公司举办的语言与智能技术竞赛开放了基于百度百科和百度信息流的大规模中文信息抽取数据集 (Li et al., 2019)。该数据集

包含49个关系种类数和458184个关系实例数。

6.2 基于神经网络方法研究现状

为了在中文语料中获得更丰富的高级特征，高层语义注意力机制的分段卷积神经网络模型 (武文雅 et al., 2019)用于中文关系抽取。在模型的向量表示中，添加了HowNet中的上位词向量特征。该方法在ACE2005数据集上的实验结果F1值达到73.94%，COAE2016数据集上F1值达到78.41%。由于中文句式和语法结构复杂，并且汉语有更多歧义，会影响中文实体关系分类的效果，一种基于多特征自注意力的实体关系抽取方法 (李卫疆 et al., 2019)，充分考虑词汇、句法、语义和位置特征，使用基于自注意力的双向长短期记忆网络来进行关系预测。

此外，利用多粒度信息和外部知识进行中文关系抽取 (Li et al., 2019)。具体的，多粒度信息主要包含三部分：字向量、词向量和词义向量。除了会用到字向量和词向量外，还会用到词义向量。使用HowNet作为外部知识库，对于给定词，通过检索HowNet可以获得词的所有词义信息。然后通过SAT模型 (Niu et al., 2017)将每个词义转换为实值向量。将字向量、词向量和词义向量通过Lattice LSTM编码层，最后经过关系分类层，将编码层得出的隐藏层状态作为输入，经过注意力计算，进行关系分类。通过在不同领域的三个数据集（Chinese SanWen，ACE 2005 Chinese corpus and FinRE）上进行的实验表明，他们的模型具有显著的优越性，在ACE2005数据集上F1达到78.71%。

基于DuIE中文信息抽取数据集，一个端到端的框架 (Liu et al., 2019)用于关系抽取，该框架首先在具有关系提及层的原始文本中捕获关系提及，然后进行实体标记，其目的是使用给定的关系提及对相应的三元组实体进行解码。此方法在验证集下的f1值达到84.8%。

6.3 小结

目前基于神经网络的中文实体关系抽取在公共数据集上的研究还较少，现有方法为了提取中文的丰富语义特征，都融合了知识库资源。2019年，中国计算机学会、中国中文信息学会联合百度公司举办的语言与智能技术竞赛开放了基于百度百科和百度信息流的大规模中文信息抽取数据集DuIE，有利推动了中文关系抽取研究的发展，中文关系抽取方法将不断涌现、性能不断提高。表3给出中文数据集上的实验结果对比。

数据集	方法	F1值
COAE2016	(武文雅 et al., 2019)	78.41%
	(李卫疆 et al., 2019)	81.49%
ACE2005	(武文雅 et al., 2019)	73.94%
	(Li et al., 2019)	78.71%
DuIE	(Liu et al., 2019)	84.8%

Table 4: 关系抽取在中文数据集上的结果

7 总结与展望

关系抽取作为信息抽取不可或缺的部分，是知识图谱、文本内容理解的重要支撑技术之一。根据领域的划分，可分为限定域关系抽取和开放域关系抽取。本文详细讨论了限定域关系抽取的三大类方法：有监督方法、远程监督方法和实体关系联合抽取方法。根据本文的论述，前沿的关系抽取技术在英文数据集ACE2004、ACE2005、SemEval-2010和NYT-10做了许多工作，在中文数据集上相对较少。

本文通过对现有关系抽取研究方法的总结，提出以下关系抽取未来的研究路线：

(1)前沿的关系抽取技术在主流英文数据集ACE2004、ACE2005、SemEval-2010和NYT-10做了许多工作。NYT-10数据集是自动构建的，通过将Freebase知识库与纽约时报语料库(NYT)的关系对齐而形成，此数据集没有手动注释，存在着数据噪声的问题。SemEval-2010数据集通过引入手动注释达到了相对较高的质量，但数据规模依然太小。未来工作可以开发出高质量的基于中文的关系抽取数据集，并且不断提升关系抽取技术在中文数据集上的性能。

(2)由于大量的关系事实都是通过多个句子来表达，句子级的关系抽取受到了不可避免的限制，因此，未来关系抽取的研究方向会从句子级推广到篇章级，通过读取和推理一个文档中的多个句子，能够有效提升关系抽取性能。

(3)关系抽取是一项复杂的任务，无论是有监督的数据，还是远程监督数据，纯数据驱动模型是远远不够的。如何从现有数据中挖掘和学习有用信息，以及如何将结构化知识、语言知识、领域知识融合进关系抽取模型中，这是两个重要的课题对于语言理解有着重要意义。

(4)多模态学习在关系抽取任务中的应用和研究。在互联网上存在着多种形式的数据，如自然语言、图片、结构化文本，这每一种数据形式可以称为一种模态。在构造数据集时，可以融入这些多模态数据。通过利用多模态之间的互补性，剔除模态间的冗余性，从而可以学习到更好的特征表示。

参考文献

Kurt D. Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. *International Conference on Management of Data, pages 1247-1250.*

白龙, 靳小龙, 席鹏弼. 2019. 基于远程监督的关系抽取研究综述. 中文信息学报, *33(10):10-17.*

Rui Cai, Xiaodong Zhang and Houfeng Wang. 2016. Bidirectional recurrent convolutional neural network for relation classification. *The 54th Annual Meeting of the Association for Computational Linguistics, pages 756-765.*

Ronan Collobert, Jason Weston, Michael Karlen, Koray Kavukcuoglu and Pavel P. Kuksa. 2011. language processing (almost) from scratch. *Journal of Machine Learning Research, pages 2493-2537.*

Ronan Collobert and Jason Weston. 2008. A Unified Architecture for Natural Language Processing. *The Twenty-Fifth International Conference on Machine Learning, pages 160-167.*

Li Dong and Mirella Lapata. 2016. Language to logical form with neural attention. *The 54th Annual Meeting of the Association for Computational Linguistics, pages 7-12.*

Timothy Dozat and Christopher D. Manning. 2016. Deep biaffine attention for neural dependency parsing. *The 5th International Conference on Learning Representations, pages 24-26.*

Jinhua Du, Jingguang Han, Andy Way and Dadong Wan. 2018. Multi-level structured self-attentions for distantly supervised relation extraction. *The 2018 Conference on Empirical Methods in Natural Language Processing, pages 2216-2225.*

Tsu-Jui Fu, Peng-Hsuan Li and Wei-Yun Ma. 2019. GraphRel: Modeling Text as Relational Graphs for Joint Entity and Relation Extraction. *The 57th Annual Meeting of the Association for Computational Linguistics, pages 1409-1418.*

Jiatao Gu, Zhengdong Lu, Hang Li and Victor O. K. Li. 2016. GraphRel: Modeling Text as Relational Graphs for Joint Entity and Relation Extraction. *the 55th Annual Meeting of the Association for Computational Linguistics, pages 199-208.*

Zhijiang Guo, Yan Zhang and Wei Lu. 2019. Attention Guided Graph Convolutional Networks for Relation Extraction. *The 57th Annual Meeting of the Association for Computational Linguistics, pages 241-251.*

Shizhu He, Cao Liu, Kang Liu and Jun Zhao. 2017. Generating natural answers by incorporating copying and retrieving mechanisms in sequence-to-sequence learning. *The 55th Annual Meeting of the Association for Computational Linguistics, pages 199-208.*

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, Stan Szpakowicz. 2009. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. *The 5th International Workshop on Semantic Evaluation, pages 15-16.*

Yi Yao Huang and William Yang Wang. 2017. Deep residual learning for weakly-supervised relation extraction. *The 2017 Conference on Empirical Methods in Natural Language Processing, pages 1803-1807.*

Guoliang Ji, Kang Liu, Shizhu He and Jun Zhao. 2017. Distant supervision for relation extraction with sentence-level attention and entity descriptions. *The 31th AAAI Conference on Artificial Intelligence, pages 3060-3066.*

Xiaotian Jiang, Quan Wang, Peng Li and Bin Wang. 2016. Relation extraction with multi-instance multi-label convolutional neural networks. *The 26th International Conference on Computational Linguistics: Technical Papers, pages 1471-1480.*

Kambhatla N. 2016. Combining lexical, syntactic and semantic features with Maximum Entropy models for extracting relations. *The Meeting on Association for Computational Linguistics, pages 22-28.*

Arzoo Katiyar and Claire Cardie. 2017. Going out on a limb: Joint extraction of entity mentions and relations without dependency trees. *The 55th Annual Meeting of the Association for Computational Linguistics, pages 917-928.*

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *The 2014 Conference on Empirical Methods in Natural Language Processing, pages 25-29.*

Thomas N. Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *The 5th International Conference on Learning Representations, pages 24-26.*

Joohong Lee, Sangwoo Seo and Yong Suk Choi. 2019. Semantic Relation Classification via Bidirectional LSTM Networks with Entity-Aware Attention Using Latent Entity Typing. *Symmetry, pages 785.*

Qi Li and Heng Ji. 2014. Incremental joint extraction of entity mentions and relations. *The 52nd Annual Meeting of the Association for Computational Linguistics, pages 402-412.*

Shuangjie Li, Wei He, Yabing Shi, Wenbin Jiang, Haijin Liang, Ye Jiang, Yang Zhang, Yajuan Lyu and Yong Zhu. 2019. DuIE: A Large-Scale Chinese Dataset for Information Extraction. *The CCF International Conference on Natural Language Processing and Chinese Computing, pages 791-800.*

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou and Jiwei Li. 2019. Entity-Relation Extraction as Multi-Turn Question Answering. *The 57th Annual Meeting of the Association for Computational Linguistics, pages 1340-1350.*

Ziran Li, Ning Ding, Zhiyuan Liu, Haitao Zheng and Ying Shen. 2019. chinese Relation Extraction with Multi-Grained Information and External Linguistic Knowledge. *The 57th Annual Meeting of the Association for Computational Linguistics, pages 4377-4386.*

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. *The 54th Annual Meeting of the Association for Computational Linguistics, pages 2124-2133.*

Zhouhan Lin, Minwei Feng, Cícero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *The 5th International Conference on Learning Representations, pages 24-26.*

李卫疆, 李涛, 漆芳. 2019. 基于多特征自注意力BLSTM的中文实体关系抽取. 中文信息学报, *33(10):47-56.*

Zhenhua Liu, Tianyi Wang, Wei Dai, Zehui Dai and Guangpeng Zhang. 2019. A Relation Proposal Network for End-to-End Information Extraction. *The Natural Language Processing and Chinese Computing, pages 782-790.*

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. *The 2017 Conference on Empirical Methods in Natural Language Processing, pages 1506-1515.*

Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean. 2013(b). Efficient estimation of word representations in vector space. *The 1st International Conference on Learning Representations, pages 2-4.*

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado and Jeffrey Dean. 2013(a). Distributed Representations of Words and Phrases and their Compositionality. *The 27th Annual Conference on Neural Information Processing Systems, pages 3111-3119.*

Melanie Reiplinger, Michael Wiegand and Dietrich Klakow. 2009. Distant supervision for relation extraction without labeled data. *The 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP, pages 2-7.*

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. *The 54th Annual Meeting of the Association for Computational Linguistics, pages 7-12.*

Yilin Niu, Ruobing Xie, Zhiyuan Liu and Maosong Sun. 2017. Improved Word Representation Learning with Sememes. *The 55th Annual Meeting of the Association for Computational Linguistics, pages 2049-2058.*

Sachin Pawar, Girish K. Palshikar and Pushpak Bhattacharyya. 2017. Relation Extraction : A Survey. *arXix:1712.05191v1 [cs.CL] .*

Jeffrey Pennington, Richard Socher and Christopher D. Manning. 2014. Glove: Global vectors for word representation. *The 2014 conference on empirical methods in natural language processing, pages 1532-1534.*

Sebastian Riedel, Limin Yao and Andrew McCallum. 2010. Modeling Relations and Their Mentions without Labeled Text. *European Conference on Machine Learning & Knowledge Discovery in Databases, pages 20-24.*

Bryan Rink and Sanda M. Harabagiu. 2010. UTD: Classifying semantic relations by combining lexical and semantic resources. *The 5th International Workshop on Semantic Evaluation, pages 15-16.*

Cícero Nogueira dos Santos, Bing Xiang and Bowen Zhou. 2015. Classifying relations by ranking with convolutional neural networks. *The 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, pages 626-634.*

Richard Socher, Brody Huval, Christopher D. Manning and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. *The 2012 joint conference on empirical methods in natural language processing and computational natural language learning, pages 1201-1211.*

Changzhi Sun, Yeyun Gong, Yuanbin Wu, Ming Gong, Daxin Jiang, Man Lan, Shiliang Sun and Nan Duan. 2019. Joint Type Inference on Entities and Relations via Graph Convolutional. *The 57th Annual Meeting of the Association for Computational Linguistics, pages 1361-1370.*

Changzhi Sun, Yuanbin Wu, Man Lan, Shiliang Sun, Wenting Wang, Kuang-Chih Lee and Kewen Wu. 2018. Extracting Entities and Relations with Joint Minimum Risk Training. *The 2018 Conference on Empirical Methods in Natural Language Processing, pages 2256-2265.*

Ryuichi Takanobu, Tianyang Zhang, Jiexi Liu and Minlie Huang. 2019. A Hierarchical Framework for Relation Extraction with Reinforcement Learning. *The Thirty-Third AAAI Conference on Artificial Intelligence, pages 7072-7079.*

Zhixing Tan, Mingxuan Wang, Jun Xie, Yidong Chen and Xiaodong Shi. 2018. Deep semantic role labeling with self-attention. *The Thirty-Second AAAI Conference on Artificial Intelligence, pages 4929-4936.*

Shikhar Vashishth, Rishabh Joshi, Sai Suman Prayaga, Chiranjib Bhattacharyya and Partha P. Talukdar. 2018. Improving distantly-supervised neural relation extraction using side information. *The 2018 Conference on Empirical Methods in Natural Language Processing, pages 1257-1266.*

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems, pages 5998-6008.*

Oriol Vinyals, Meire Fortunato and Navdeep Jaitly. 2015. Pointer networks. *Advances in Neural Information Processing Systems, pages 2692-2700.*

Guanying Wang, Wen Zhang, Ruoxu Wang, Yalin Zhou, Xi Chen, Wei Zhang, Hai Zhu and Huajun Chen. 2018. Label-free distant supervision for relation extraction via knowledge graph embedding. *The 2018 Conference on Empirical Methods in Natural Language Processing, pages 2246-2255.*

武文雅, 陈钰枫, 徐金安. 2019. 基于高层语义注意力机制的中文实体关系抽取. 广西师范大学学报, *33(01):32-41.*

武文雅, 陈钰枫, 徐金安. 2018. 中文实体关系抽取研究综述. 计算机与现代化, *000(008):25-31.*

Jason Weston, Antoine Bordes, Oksana Yakhnenko and Nicolas Usunier. 2013. Connecting language and knowledge bases with embedding models for relation extraction. *The 2013 Conference on Empirical Methods in Natural Language Processing, pages 1366-1371.*

Kun Xu, Yansong Feng, Songfang Huang and Dongyan Zhao. 2015(a). Semantic relation classification via convolutional neural networks with simple negative sampling. *The 2015 Conference on Empirical Methods in Natural Language Processing, pages 17-21.*

Yan Xu, Lili Mou, Ge Li, Yunchuan Chen, Hao Peng and Zhi Jin. 2015(b). Classifying relations via long short term memory networks along shortest dependency paths. *The 2015 Conference on Empirical Methods in Natural Language Processing, pages 1785-1794.*

Dmitry Zelenko, Chinatsu Aone and Anthony Richardella. 2003. Kernel Methods for Relation Extraction. *Journal of Machine Learning Research, pages 1083-1106.*

Daojian Zeng, Kang Liu, Yubo Chen and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. *The 2015 Conference on Empirical Methods in Natural Language Processing, pages 1753-1762.*

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou and Jun Zhao. 2014. Relation classification via convolutional deep neural network. *The 2015 Conference on Empirical Methods in Natural Language Processing, pages 23-29.*

Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu and Jun Zhao. 2018. Extracting relational facts by an end-to-end neural model with copy mechanism. *The 56th Annual Meeting of the Association for Computational Linguistics, pages 506-514.*

Meishan Zhang, Yue Zhang and Guohong Fu. 2017. End-to-end neural relation extraction with global optimization. *The 2017 Conference on Empirical Methods in Natural Language Processing, pages 1730-1740.*

Ningyu Zhang, Shumin Deng, Zhanlin Sun, Guanying Wang, Xi Chen, Wei Zhang and Huajun Chen. 2019. Long-tail Relation Extraction via Knowledge Graph Embeddings and Convolution Networks. *The 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 3016-3025.*

Yuhao Zhang, Peng Qi and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. *The 2018 Conference on Empirical Methods in Natural Language Processing, pages 2205-2215.*

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou and Bo Xu. 2017. Joint extraction of entities and relations based on a novel tagging scheme. *The 55th Annual Meeting of the Association for Computational Linguistics, pages 1227-1236.*

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. *The 54th Annual Meeting of the Association for Computational Linguistics, pages 207-212.*

小样本关系分类研究综述

胡晗　　刘鹏远*
北京语言大学　信息科学学院
国家语言资源监测与研究平面媒体中心
北京市海淀区学院路15号, 100083
201821198609@stu.blcu.edu.cn　liupengyuan@blcu.edu.cn

摘要

关系分类作为构建结构化知识的重要一环，在自然语言处理领域备受关注。但在很多应用领域中(如医疗、金融等领域)，收集充足的用于训练关系分类模型的数据十分困难。近年来，仅需要少量训练样本的小样本学习逐渐应用于关系分类研究中。本文对近期小样本关系分类模型与方法进行了系统的综述。根据度量方法的不同，将现有方法分为原型式和分布式两大类。根据是否利用额外信息，将模型分为预训练和非预训练两大类。此外，除了常规设定下的小样本学习，本文还梳理了跨领域和稀缺资源场景下的小样本学习，探讨了目前小样本关系分类方法的局限性，并分析了跨领域小样本学习面临的技术挑战。最后，展望了小样本关系分类未来的发展方向。

关键词： 关系分类；小样本学习；元学习

Few-Shot Relation Classification: A Survey

Han Hu　　Pengyuan Liu*
Beijing Language and Culture University, School of Information Science
Language Resources Monitoring and Reserch Center
15 Xueyuan Road, Haidian District, Beijing, 100083, China
201821198609@stu.blcu.edu.cn　liupengyuan@blcu.edu.cn

Abstract

As an important part of constructing structured knowledge, relation classification has attracted much attention in the field of natural language processing. However, in many application fields (medical and financial fields), it is very difficult to collect sufficient data for training relation classification model. In recent years, few-shot learning research which only needs a small number of training samples is emerging in various fields. In this paper, the recent models and methods of few-shot relation classification are systematically reviewed. According to the different measurement methods, the existing methods are divided into prototype and distributed. According to whether to use additional information, the model is divided into two categories: pretraining and non-pretraining. In addition to the regular setting of few-shot learning, we also comb the cross domain few-shot learning and few-few-shot learning, and discusse the limitations of current few-shot relation classification methods, and analyze the technical challenges faced by cross domain few-shot models. Finally, the future development of few-shot relation classification is prospected.

Keywords: Relation Classification , Few-shot Learning , Meta Learning

* 通讯作者　Corresponding Author

1 引言

关系分类是自然语言处理领域中的一项重要任务，它致力于判断给定语句中两个目标实体之间的预定义关系，为构建结构化知识(如，知识图谱)提供了基础。当前用于该任务的主流深度学习模型以大量监督数据为驱动，导致模型泛化能力依赖于监督数据的数量和质量。尽管正则技术被广泛用来降低深度学习模型对训练数据的过拟合，但其并不能为模型提供额外的监督信息。因此当监督数据不足时，简单地对模型加以正则并不能真正解决泛化问题(Wang et al., 2019b)。为了缓解训练数据不足的问题，节省人工标注成本，Mintz et al. (2009)采用了远程监督的方法。文章假设“两个实体如果在知识库中存在某种关系，则包含这两个实体的语句在某种程度上能表示出这种关系”，启发式地将语句中的目标实体与知识库中的实体对齐，达到自动标注语句的目的。但这个假设也带来了后续的问题：(1)同一实体对在不同语句中所蕴涵的关系可能不同，利用远程监督方法会产生噪声数据(如Figure 1所示)；(2)很多领域的知识库并不完善(如，医疗领域)，且大部分实体对和关系呈长尾分布，通过这种方法获取的可用于训练的数据仍然不足(如Figure 2所示)。

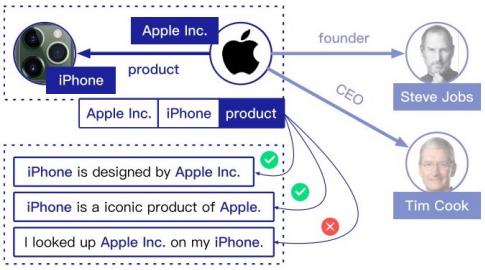


Figure 1: 远程监督方法引入了噪声数据。(Han et al. (2020))

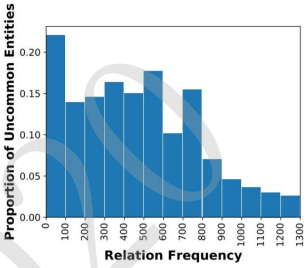


Figure 2: DBpedia中关系出现的频率与对应未见实体占比分布图。(Wang et al. (2019c))

相比之下，人类拥有利用过去所学知识快速学习新概念的能力。因此，研究者们希望构建一种新的训练方法，使模型仅在少量训练样本的情况下学习，并具备良好的泛化能力。Li Fei-Fei et al. (2006)首次提出单样本学习(One-Shot Learning)，采用贝叶斯模型，利用已学习的类别知识帮助模型在每个新类别仅有单个训练样本的情况下进行学习。至今，已有大量研究工作投入到单/小样本学习(One/Few-Shot Learning)领域，其中最具有代表性且主流的方法是元学习(Meta Learning)方法。元学习，或称“学会学习”，是系统地观察模型在不同的学习任务中的表现，从这种经验或元数据(Meta Data)中学习，然后以更快的速度学习新任务的方法(Vanschoren, 2018)。

目前，小样本学习的研究主要集中于计算机视觉领域。启发于人类的记忆，研究者们提出记忆网络，将先验知识存储在记忆模块中以供检索与更新(Weston et al., 2015; Sukhbaatar et al., 2015)。从优化的角度出发，一些研究者训练一个元优化器，帮助模型高效搜索合适的任务参数(Andrychowicz et al., 2016; Li and Malik, 2016)。另一些研究者则通过学习一个与任务无关的通用初始化参数，使得模型仅在少量训练样本情况下快速适应新任务(Finn et al., 2017)。Vinyals et al. (2016)从度量的角度提出了匹配网络，并首次提出了训练与测试过程相匹配的Episode训练原则(如Figure 3所示)。

在自然语言处理领域，小样本学习刚刚兴起。Yu et al. (2018)利用多个度量函数来解决任务多样性小样本分类问题。Geng et al. (2019)和Geng et al. (2020a)提出静态和动态记忆的归纳网络来解决因类别样本过少带来的样本方差问题。Han et al. (2018)首次将小样本学习引入关系分类任务，构建了小样本关系分类数据集FewRel，并尝试了几种典型的小样本学习方法与人类基准作比较。许多研究者在此基础上进行了探索，Baldini Soares et al. (2019)提出的无监督句子匹配方法在这一任务上的表现甚至超越了人类基准。针对小样本关系任务的多样性及任务中可能存在的噪声样本，Gao et al. (2019a)利用层级注意力来增强模型对小样本任务多样性以及噪声样本的鲁棒性。Xie et al. (2020)则通过异构图网络与对抗训练减少模型对噪声样本的敏感性。Obamuyide and Vlachos (2019)将监督式关系分类任务视为元学习的一个例

子，提出模型无关的元学习方案，力求模型在数据充足与数据稀缺两种情况下都有良好表现。由于一些领域的元数据不足以训练一个在该领域任务间有较好泛化能力的元模型，Gao et al. (2019b)在FewRel数据集的基础上提出了FewRel2.0数据集，探索元学习跨领域泛化以及非预定义类别检测问题。Geng et al. (2020b)则提出了更严苛的元训练条件，探索元学习模型在有限元数据情况下的学习能力。

在这篇综述文章中，我们系统地回顾了小样本关系分类任务具有代表性和启发性的工作(如Figure 4所示)。探讨了这些工作在当前用于解决该任务的元学习设定下的优势与不足。并给出了未来小样本关系分类的一些发展方向。

2 问题定义

2.1 N-way K-shot小样本分类

小样本学习是监督式机器学习的一种特殊情况，它的目标是在限制了目标任务训练数据数量的情况下，训练出对该任务新数据具有良好泛化能力的模型。

对于一个*N-way K-shot*小样本分类任务$T = \{D^{train}, D^{test}, \ell\}$，其中训练集(或称支持集)$D^{train} = \{(x_1^1, y_1), ..., (x_1^k, y_1), ..., (x_n^1, y_n), ..., (x_n^k, y_n)\}$ 包含N个类别(N一般为5或10)，每个类别K个训练样本(K一般为5或10)，测试集(或称问题集)$D^{test} = \{(x^m, y^m)\}_{m=1}^{M}$，$\ell$为损失函数。假设输入$x$和输出$y$的联合概率分布为$p(x, y)$，学习的最终目的是通过训练集$D^{train}$与损失函数$\ell$发现从$x$映射到$y$的最优假设$o^*$，且在测试集$D^{test}$上有良好的泛化能力。

由于训练集所提供的训练数据有限，用经验风险近似期望风险不够精准。当前以数据驱动为主的深度学习方法在这种任务上会出现过拟合的现象。尽管正则技术被广泛用来降低深度学习模型对训练数据的过拟合，但其并不能为模型提供额外的监督信息。因此，正则方法并不能提高小样本情况下用经验风险替代期望风险的可靠性(Wang et al., 2019b)。为了提高小样本情况下模型的泛化能力，结合先验知识至关重要。

2.2 元学习

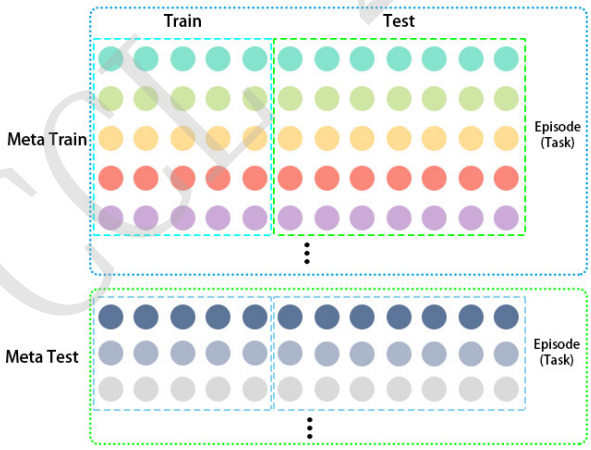


Figure 3: 元学习Episode训练框架

元学习，或称“学会学习”，是元学习器(Meta Learner)系统地观察基学习器(Base Learner)在不同的学习任务(Task)中的表现，从这种经验或元数据(Meta Data)中学习，然后以更快的速度学习未曾见过的新任务(Novel Task)的方法(Vanschoren, 2018)。这个过程中，存在两个层面的学习：(1)元学习器迭代地学习不同任务间的元知识(Meta Knowledge)；(2)基学习器基于元知识以及新任务中的特定信息快速学习并处理该任务。

对于一个小样本分类任务，基学习器的目标是找到最优假设o^*。为了接近o^*，基学习器确定了假设空间$\mathcal{H}$，其中包含了由φ参数化的假设$h(\cdot, \varphi)$。优化算法通过搜索假设空间$\mathcal{H}$来找到一个对于D^{train}最优的假设h。Wang et al. (2019b)系统地分析了经验风险的可靠性与样本复杂度和假设空间之间的联系。作者指出，为了使经验风险对期望风险的近似以一定概率达到一定精度，模型决定的假设空间越复杂，所需要的训练样本就越多。

在元学习中，为了减少训练所需的样本，元学习器需要从大量相似任务中学习元知识。然后，根据元知识构建假设空间$\mathcal{H}$的草图，以限制假设空间的大小。而基学习器则通过新任务特定的信息完成$\mathcal{H}$的具体构建。假设$p(T)$为小样本任务$T = \{D^{train}, D^{test}, \ell\}$的分布。在元训练阶段，元学习器$f_\theta(\cdot)$从包含$N_{meta}^{train}$个独立同分布任务的元训练集$D_{meta}^{train} = \{T_s^i \sim p(T)\}_{i=1}^{N_{meta}^{train}}$中学习。基学习器$g_\varphi(\cdot)$根据元知识从$D_{T_s}^{train}$中学习，并在$D_{T_s}^{test}$上评估损失。通过最小化基学习器在一系列任务上的损失来优化元学习器的参数θ:

$$\theta = \arg\min_\theta \mathbb{E}_{T_s \sim p(T)} \ell_\theta(D_{T_s}) \tag{1}$$

在元测试阶段，与元训练集类别互斥的元测试集$D_{meta}^{test} = \{T_t^j \sim p(T)\}_{j=1}^{N_{meta}^{test}}$被用来测试元学习器对新的小样本任务的泛化能力。

从监督式机器学习的角度，Chao et al. (2020)给出了更直观的解释。作者认为可以将元训练阶段视为元学习器根据一组$\{(D^{train}, h^*)\}$元样本对进行监督式训练的过程。给定一个小样本任务，假设基学习器在该任务上的最优假设为h^*，而基学习器根据元知识及训练集学习到的假设为h。给定该任务的测试集$D^{test} = \{(x^m, y^m)\}_{m=1}^M$，元学习器在该任务上的损失为$\ell_{meta}(g_\varphi(D^{train}|\theta), h^*) = |\mathcal{L}^{test}(h) - \mathcal{L}^{test}(h^*)|$，其中$\mathcal{L}^{test}(h) = \frac{1}{M}\sum_{m=1}^M \ell(h(x^m), y^m)$。假设$h^*$最小化$\mathcal{L}^{test}$，那么$\ell_{meta}(g_\varphi(D^{train}|\theta), h^*) = \mathcal{L}^{test}(h)$，即我们可以用$D^{test}$和$\ell$代替$h^*$与$\ell_{meta}$。因此，用于训练元学习模型的训练集由$\{(D^{train}, h^*)\}$变为了$\{(D^{train}, D^{test})\}$。最终，式(1)可以改写为:

$$\theta = \arg\min_\theta \sum_{i=1}^{N_{meta}^{train}} \sum_{m=1}^{M} \ell(g_\varphi(D_i^{train}|\theta)(x_i^m), y_i^m) \tag{2}$$

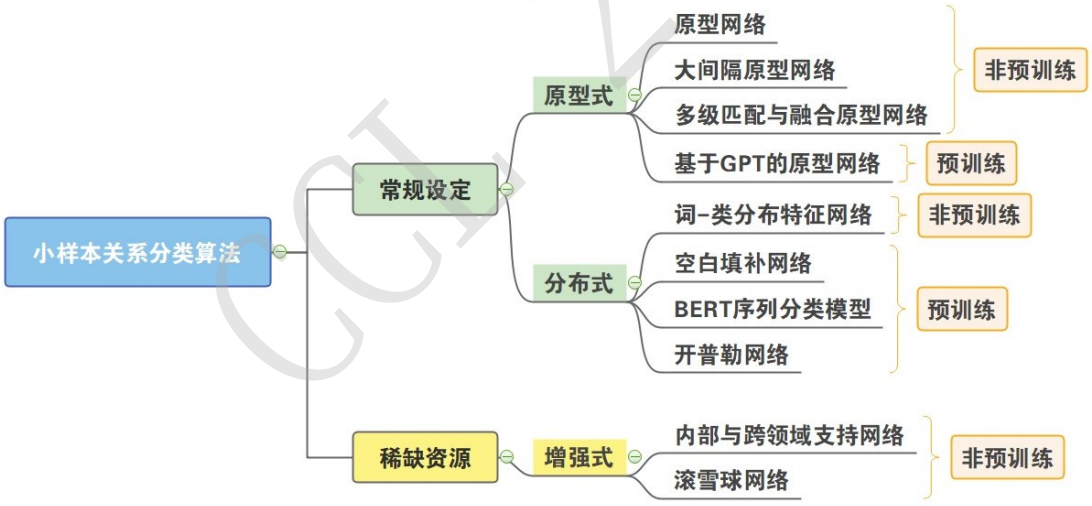


Figure 4: 小样本关系分类算法分类导图

3 常规小样本关系分类

3.1 数据集

FewRel(Han et al., 2018)是第一个英文小样本关系分类数据集，它包含100种关系，每种关系700个样本。作者以Wikipedia为数据库，Wikidata为知识库，通过远程监督的方法将数据库中的句子与知识库中的事实对齐。为了扩大实体集，作者首先利用命名实体识别技术挖掘文章中的非锚点实体，然后通过实体链接技术将挖掘出的实体与Wikidata中的实体进行匹配。由于对于表达某种关系的一组句子来说，其包含的可能是同一对实体。为了避免模型机械地根据句子中出现的实体对而不是句子本身的语义来进行关系分类，作者在每种关系中，对于同一对实体只保留一个样本。之后，去除样本量不足1000的关系，对剩余的关系，每种关系随机抽

取1000个样本。经过标注人员的筛选标注，去除正样本不足700的关系，以kappa值对剩余的关系进行降序排列，保留前100种关系。最终，数据集以64:16:20的比例被划分为训练集、验证集和测试集。

3.2 常规小样本关系分类算法

在常规小样本关系分类算法中，基于度量和优化的元学习方法最为常见。Han et al. (2018)测试了基于参数生成的元学习MetaNet(Munkhdalai and Yu, 2017)，基于图网络的元学习GNN(Satorras and Bruna, 2017)，基于时序卷积的元学习SNAIL(Mishra et al., 2017)。但这些复杂的方法在小样本关系分类任务上的表现并不如简单的基于度量的方法。后续的研究者在此基础上进行探索，我们将这些模型分为基于原型和基于分布式表达两大类。

3.2.1 原型式小样本关系分类算法

原型式小样本关系分类算法是基于度量的一类算法。度量方法将样本嵌入到一个更小的空间中，使得相似的样本聚在一起，不相似的样本分离。这些方法的不同点在于用于生成类别原型的向量表示以及生成类别原型的方法。

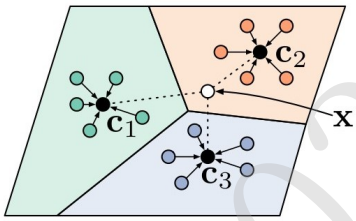


Figure 5: 原型网络(图片来自Snell et al. (2017))

- **原型网络(Prototypical Network(Snell et al., 2017))**假设存在一个嵌入空间，在这个空间里，每个类别的点都围绕着该类别的原型聚集(如Figure 5所示)。它利用卷积神经网络(CNN)作为编码器f_θ，将$x_n^k \in D^{train}$和$x^{test} \in D^{test}$非线性映射到该嵌入空间，然后构造其类别原型:

$$c_n = \frac{1}{K}\sum_{k=1}^{K} f_\theta(x_n^k) \tag{3}$$

  最后衡量$f_\theta(x^{test})$与c_n之间的距离$d(f_\theta(x^{test}), c_n)$（如，欧式距离）对其作最近邻分类。最终，通过分类损失对嵌入空间进行优化。

  在原型网络中，编码器$f_\theta(\cdot)$既是元学习器，也是基学习器。类似于多任务学习，它假设如果学习到的嵌入空间能够处理很多任务，那么这个空间也有足够的能力处理新任务。支持集不再用于基学习器的参数更新，而是作为嵌入空间中的类别锚点。

- **大间隔原型网络(Large Margin Prototypical Network(Fan et al., 2019))**在原型网络的基础上，采用了更加细粒度的特征表示以及额外的目标函数。除了利用句子级别的表示$f_{sentence}(x) = f_\theta(x)$以外，作者根据关系分类的特点，将句子分为五个部分，头实体之前的部分r_f，头实体e_h，头实体和尾实体之间的部分r_m，尾实体e_t，尾实体之后的部分r_b，利用多个CNN对其分别作嵌入得到嵌入表示$\mathbf{r}_f$，$\mathbf{e}_h$，$\mathbf{r}_m$，$\mathbf{e}_t$和$\mathbf{r}_b$。之后，将得到的表示拼接起来送入一个全连接层并用ReLU激活，获取这些表示的非线性关系:

$$f_{phrase}(x) = \mathbf{ReLU}(f_\varphi(\mathbf{r}_f \oplus \mathbf{e}_h \oplus \mathbf{r}_m \oplus \mathbf{e}_t \oplus \mathbf{r}_b)) \tag{4}$$

  然后将的句子级表示和短语级表示拼接起来得到最终的表示:

$$f(x) = f_{sentence}(x) \oplus f_{phrase}(x) \tag{5}$$

为了在嵌入空间中增加类间距离，缩短类内距离，作者额外采用了三元组损失函数作为辅助：

$$\mathcal{L}_{triplet} = \sum_{i=1}^{N} \mathbf{max}(0, \gamma + \|f(a_i) - f(p_i)\|^2) + \|f(a_i) - f(n_i)\|^2)) \tag{6}$$

其中，N为Episode/Task的大小，$a_i = c_n$是锚点，p_i是正样例，n_i是负样例。平衡交叉熵损失与三元组损失得到最终的损失函数：

$$\mathcal{L} = \mathcal{L}_{softmax} + \lambda \mathcal{L}_{triplet} \tag{7}$$

- **多级匹配与融合原型网络(MLMAN(Ye and Ling, 2019))**利用注意力机制，通过考虑支持集D^{train}和问题样本x^{test}的局部和实例两个层面的匹配信息，对两者作交互式编码。作者首先利用CNN对D^{train}和x^{test}进行上下文编码，得到$\{\mathbf{S}_k \in \mathbb{R}^{T_k \times d_c}; k = 1, ..., K\}$与$\mathbf{Q} \in \mathbb{R}^{T_q \times d_c}$。然后，将支持集拼接得到整个支持集的矩阵表示$\mathbf{C} \in \mathbb{R}^{T_s \times d_c}, T_s = \sum_{k=1}^{K} T_k$。

  对支持集与问题样例作局部匹配与融合：

$$\mathbf{A} = \mathbf{Q}\mathbf{C}^\top \in \mathbb{R}^{T_q \times T_s} \tag{8}$$

$$\widetilde{\mathbf{Q}} = \mathbf{Softmax}(\mathbf{A})\mathbf{C} \in \mathbb{R}^{T_q \times d_c} \tag{9}$$

$$\widetilde{\mathbf{C}} = \mathbf{Softmax}(\mathbf{A}^\top)\mathbf{Q} \in \mathbb{R}^{T_s \times d_c} \tag{10}$$

$$\bar{\mathbf{Q}} = \mathbf{ReLU}([\mathbf{Q}; \widetilde{\mathbf{Q}}; |\mathbf{Q} - \widetilde{\mathbf{Q}}|; \mathbf{Q} \odot \widetilde{\mathbf{Q}}]\mathbf{W}_1) \in \mathbb{R}^{T_q \times d_h} \tag{11}$$

$$\bar{\mathbf{C}} = \mathbf{ReLU}([\mathbf{C}; \widetilde{\mathbf{C}}; |\mathbf{C} - \widetilde{\mathbf{C}}|; \mathbf{C} \odot \widetilde{\mathbf{C}}]\mathbf{W}_1) \in \mathbb{R}^{T_s \times d_h} \tag{12}$$

将$\bar{\mathbf{C}}$还原为独立的类别矩阵$\{\bar{\mathbf{S}}_k \in \mathbb{R}^{T_k \times d_h}\}_{k=1}^{K}$，并用单层双向长短时记忆网络(BiLSTM)编码所有的$\bar{\mathbf{S}}_k$与$\bar{\mathbf{Q}}$，得到最终的局部表示$\hat{\mathbf{S}}_k$ 和$\hat{\mathbf{Q}}$。对局部表示进行最大池化和平均池化并拼接，得到最终向量表示：

$$\hat{\mathbf{s}}_k = [\mathbf{max}(\hat{\mathbf{S}}_k); \mathbf{ave}(\hat{\mathbf{S}}_k)], \forall k \in \{1, ...K\} \tag{13}$$

$$\hat{\mathbf{q}} = [\mathbf{max}(\hat{\mathbf{Q}}); \mathbf{ave}(\hat{\mathbf{Q}})] \tag{14}$$

除了对支持集与问题样例作局部匹配与融合，作者还采用了实例级匹配来构造类别原型。不同于原始原型网络通过平均向量的方式构造类别原型，MLMAN通过多层感知机度量$\hat{\mathbf{s}}_k$与$\hat{\mathbf{q}}$匹配程度，赋予不同$\hat{\mathbf{s}}_k$不同的权重来构造带权平均类别原型：

$$\beta_k = \mathbf{v}^\top(\mathbf{ReLU}(\mathbf{W}_2[\hat{\mathbf{s}}_k; \hat{\mathbf{q}}])) \tag{15}$$

$$\hat{\mathbf{s}} = \sum_{k=1}^{K} \frac{e^{\beta_k}}{\sum_{k'=1}^{K} e^{\beta_{k'}}} \hat{\mathbf{s}}_k \tag{16}$$

最终进行类别匹配，对问题样例作出分类：

$$f(\{s_k\}_{k=1}^{K}, q) = \mathbf{v}^\top(\mathbf{ReLU}(\mathbf{W}_2[\hat{\mathbf{s}}; \hat{\mathbf{q}}])) \tag{17}$$

为了生成更具代表性的类别原型，除了分类损失外，作者额外加入了非一致性度量损失，保证同一类别中的样本不会互相偏离：

$$\mathcal{L}_{incon} = \frac{1}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} \|\hat{\mathbf{s}}_k^i - \hat{\mathbf{s}}^i\|_2^2 \tag{18}$$

$$\mathcal{L} = \mathcal{L}_{softmax} + \lambda \mathcal{L}_{incon} \tag{19}$$

- **基于GPT的原型网络(Prototypical GP-Transformer(Eberts, 2019))**采用预训练语言模型GPT替代原始原型网络中的CNN作为编码器，以获得更好的类别原型的表示。在GPT中每个句子首尾有标记符标记句子的开始$< Start >$和结束$< End >$，由于Transformer是自注意力模型，$< end >$能够注意到整个句子，因此其嵌入表示$\mathbf{h}_{<end>}$被用于后续的分类。为了标示出句子中的目标实体，作者尝试了不同的标记目标实体的方法：(1)在目标实体两侧添加标记符(常用于RNN)；(2)位置嵌入(常用于CNN)；(3)将目标实体的平均嵌入表示与句子的平均嵌入表示拼接；(4)根据目标实体划分句子作分段编码并拼接；(5)将实体的平均嵌入表示与$\mathbf{h}_{<end>}$拼接。为了加速模型的收敛，作者在任务微调阶段加入了语言模型作为辅助目标函数：

$$\mathcal{L} = \mathcal{L}_{softmax} + \lambda \mathcal{L}_{LM} \tag{20}$$

3.2.2 分布式小样本关系分类算法

分布式小样本关系分类算法主要分为两类，一类是建模句子间的分布式表示，另一类是建模句子中词对类的分布式表示。

- **空白填补网络(Matching The Blanks(Baldini Soares et al., 2019))**将Harris分布式假设拓展到关系领域，利用预训练语言模型BERT，从无标注非结构化文本中学习任务无关的关系表示。作者假设，对于任意一对关系陈述句$\mathbf{r}$和$\mathbf{r}'$，如果它们表示的关系语义相似，那么两者的内积$f_\theta(\mathbf{r})^\top f_\theta(\mathbf{r}')$应该很大，否则很小。作者观察到，在网络文本中，任意一对实体之间的每种关系都可能被陈述多次。利用这一冗余特性，作者运用实体链接方法构建了无监督数据集，提出了名为Matching The Blanks的方法来学习判断两个关系陈述句是否表达同一关系的编码器f_θ：

$$p(l = 1|\mathbf{r}, \mathbf{r}^{'}) = \frac{1}{1 + e^{f_\theta(\mathbf{r})^\top f_\theta(\mathbf{r}')}} \tag{21}$$

$$\mathcal{L}(\mathcal{D}) = -\frac{1}{|\mathcal{D}|^2} \sum_{(\mathbf{r},e_1,e_2)\in\mathcal{D}} \sum_{(\mathbf{r}',e_1',e_2')\in\mathcal{D}} \alpha \mathbf{log} p(l = 1|\mathbf{r}, \mathbf{r}^{'}) + (1 - \alpha)\mathbf{log}(1 - p(l = 1|\mathbf{r}, \mathbf{r}^{'})) \tag{22}$$

其中，$l = 1$表示$\mathbf{r}$与$\mathbf{r}'$表示表示同一种关系，否则表示不同关系。$\alpha = \delta_{e_1,e_1'}\delta_{e_2,e_2'}$，$\delta_{e,e'}$为克罗内克函数，当且仅当$e = e'$时为1，否则为0。为了避免模型只是机械地记忆目标实体，而忽略了句子的语义，作者以概率β将目标实体随机替换为空白符[**BLANK**]。在如何标记句子目标实体问题上，作者采用了与基于GPT的原型网络相同的方法：(1)在目标实体两侧添加标记符；(2)位置嵌入。同时探索了如何从BERT的输出中得到固定长度的关系表示向量：(1)利用BERT原始的[**CLS**]；(2)拼接两个目标实体的池化表示；(3)在目标实体两侧添加标记符的基础上，拼接标记符[$\mathbf{E1}_{start}$]与[$\mathbf{E2}_{start}$]作为最终的关系表示向量。由于数据集过大，不可能比较所有的$\mathbf{r}$与$\mathbf{r}'$。作者采用了噪声对比估计训练方法(noise-contrastive estimation)，将所有包含同一对实体的关系陈述句视为正例对，从所有关系陈述句中随机选取一对句子或者选取只共享其中一个实体的句对构建负例对。最终，与BERT相似，作者平衡两种损失函数对模型进行无监督训练：

$$\mathcal{L} = \mathcal{L}_{match} + \lambda \mathcal{L}_{MLM} \tag{23}$$

- **词-类分布特征网络(Distributional Signatures(Bao et al., 2020))**通过学习在任务间具有一致性的词对类的分布特征来迁移任务间共享的元知识，同时根据词对类的重要程度构造句子表示，避免池化带来的信息丢失。模型分为两个部分，一是注意力权重生成器，另一个是用于分类的任务特定的岭回归器。权重生成器的目标是根据句子中词的分布特征生成词的重要程度。作者选用一元模型(Unigram)作为统计特征，增强对词替换扰动的鲁棒性。由于高频词通常不包含有用信息，为了降低高频词权重，增大低频词权重，作者度量了通用的词-词表重要程度：

$$s(x_i) = \frac{\varepsilon}{\varepsilon + P(x_i)} \tag{24}$$

其中$\varepsilon = 10^{-3}$，x_i是句子x的第i个词，$P(x_i)$是词x_i在整个元训练集上的一元模型似然。

同时，在支持集中相对具有辨识度的词，对于问题集可能也相对具有辨识度。因此，作者度量了特定的词-类别重要程度：

$$t(x_i) = H(P(y|x_i))^{-1} \tag{25}$$

其中，$H(\cdot)$表示熵操作，$P(y|x_i)$通过一个正则线性分类器在支持集上的估计得到。

考虑到这两种统计特征信息互补，且存在一定的噪声。作者通过BiLSTM将两者融合$h_i = \mathbf{BiLSTM}([s(x_i); t(x_i)])$，最终得到词$x_i$的权重($v$是可学习的元参数)：

$$\alpha_i = \frac{e^{v^\top h_i}}{\sum_j e^{v^\top h_j}} \tag{26}$$

在权重生成器的基础上，根据支持集构建岭回归器对问题集样本进行分类。作者首先根据词的权重，构建句子表示：

$$\phi(x) = \sum_i \alpha_i f_\theta(x_i) \tag{27}$$

然后，通过对支持集的拟合构建岭回归器(闭式解避免了梯度的二次迭代)：

$$\mathcal{L}_{RR}(\mathbf{W}) = \|\Phi_S \mathbf{W} - Y_S\|_F^2 + \lambda \|\mathbf{W}\|_F^2 \tag{28}$$

$$\mathbf{W} = \Phi_S^\top (\Phi_S \Phi_S^\top + \lambda I)^{-1} Y_S \tag{29}$$

其中，$\Phi_S \in \mathbb{R}^{NK \times d}$表示整个支持集，$Y_S \in \mathbb{R}^{NK \times N}$表示独热标签，$I$为单位矩阵。

根据得到的岭回归器，对问题集样本进行分类：

$$\hat{Y}_Q = a\Phi_Q \mathbf{W} + b \tag{30}$$

其中，$a \in \mathbb{R}^+, b \in \mathbb{R}$为通过元学习得到的用于校正岭回归器参数的元参数。

最终，通过计算预测值与真实值之间的交叉熵损失训练整个模型。

4 稀缺资源小样本关系分类

当前元学习方法假设模型处理的任务服从同一分布。但在真实场景中，模型所遇到的新任务可能并不满足这一假设。其次，尽管在元测试阶段(Meta-Test)，元学习器只需要少量的监督数据，但在元训练阶段(Meta-Train)，训练元学习器所需要的监督数据依然很庞大，例如，FewRel数据集中每个类别700个样本。在一些领域，比如医疗、金融领域，获取元数据(Meta-Data)是十分困难的。直觉上，如果一些类别的样例很少，同领域的其它类别的样例也不足以构建一个足够大的数据集用以训练元学习器(Geng et al., 2020b)。因此，为了使元学习器能够在这些领域中发挥作用，研究者们从不同的角度提出了不同的解决方法。

4.1 小样本领域适应

Gao et al. (2019b)在FewRel数据集的基础上提出了FewRel2.0数据集。作者以包含大量生物医学文献的PubMed作为数据库，以UMLS作为知识库，利用FewRel1.0数据集的构建方法，构建了一个包含25种关系，每种关系100个样本的生物医学领域的数据集。FewRel2.0沿用了FewRel1.0的训练集，但是以新数据集为测试集，以此探究元学习模型从高资源领域向低资源领域适应的问题。同时，文章提出利用BERT序列分类模型解决此问题，在表现上远远超越了基于对抗的领域适应方法。

Wang et al. (2019a)在预训练语言模型的基础上，结合知识嵌入模型(KE)，将知识图谱中的事实知识融入预训练语言模型，提出了开普勒模型(KEPLER)。作者利用预训练语言模型RoBERTa，将句子中目标实体的文本表示与整个句子编码到统一的语义空间中，在预训练过程中联合优化知识嵌入模型与掩码语言模型。以KEPLER模型作为原型网络的编码器，整个网络在FewRel2.0数据集上取得了最优的表现。

4.2 小-小样本学习

Geng et al. (2020b)通过远程监督和人员筛选的方法，构建了一个新的中文医疗健康领域的小样本关系分类数据集TinyRel-CM，以探索在限制了元数据情况下的小样本学习(Few-few-shot Learning)。数据集包含27种4个实体间的二元关系，每种关系50个样本。作者根据实体类别将其分为6个部分，其中1个作为测试集，其余5个作为训练集，构建了6个任务。为了解决元训练数据不足的问题，作者提出了利用内部支持与跨领域支持的元学习框架MICK。该框架除了对问题集进行分类外，还对支持集进行了分类，以挖掘支持集内部的知识。此外，作者利用跨领域关系分类数据集对小样本任务进行数据增强。

Gao et al. (2020)提出滚雪球网络，一种新的自举方法，利用现有关系的语义知识来挖掘新关系的样本。作者利用关系孪生网络，基于现有关系分类数据集学习样例间的关系相似度度量。在此基础上，给定一个新关系及其少量标记样本，使用关系孪生网络从无标记语料库中累计可靠样本。然后利用这些样本训练关系分类器，提高分类器对新关系的新样本的泛化能力。

Method	FewRel1.0				FewRel2.0			
	5-1	5-5	10-1	10-5	5-1	5-5	10-1	10-5
Distributional Signatures	67.10	83.53	-	-	-	-	-	-
ProtoNet(CNN)	74.52	88.40	62.38	80.45	35.09	49.37	22.98	35.22
LM-ProtoNet	76.60	89.31	65.31	82.10	-	-	-	-
ProtoNet(GPT)	81.40	92.11	72.51	86.03	-	-	-	-
MLMAN	82.98	92.66	73.59	87.29	-	-	-	-
Matching The Blank	**93.86**	**97.06**	**89.20**	**94.27**	-	-	-	-
ProtoNet-ADV(CNN)	70.28	84.63	56.34	74.67	42.21	58.71	28.91	44.35
ProtoNet(BERT)	80.68	89.60	71.48	82.89	40.12	51.50	26.45	36.93
ProtoNet(RoBERTa)	85.78	95.78	77.65	92.26	64.65	82.76	50.80	71.84
ProtoNet(KEPLER)	88.30	95.94	81.10	92.67	66.41	**84.02**	51.85	**73.60**
BERT-PAIR	88.32	93.22	80.63	87.02	**67.41**	78.57	**54.89**	66.85
RoBERTa-PAIR	89.32	93.70	82.49	88.43	66.78	81.84	53.99	70.85
KEPLER-PAIR	90.31	94.28	85.48	90.51	67.23	82.09	54.32	71.01

Table 1: 小样本关系分类算法在常规和跨领域设定下的准确率。N-K表示N-way K-shot小样本设定。Distributional Signatures论文只发布了验证集上的结果。-ADV表示对抗训练。-PAIR表示序列分类模型。部分结果引用自Wang et al. (2019a)。

5 当前技术挑战与未来研究趋势

5.1 当前小样本关系分类的技术挑战

当前小样本关系分类的研究主要集中在同领域任务间的知识迁移，且依然需要庞大的元数据训练元学习器。但这个利用一个领域大量元数据训练出的元学习器很难直接应用到其它领域。尽管，研究者们利用大型预训练语言模型去解决这个问题(Gao et al., 2019b; Wang et al., 2019a)，但是并没有显式地用到目标领域的信息。因此，这些方法实际上是领域泛化的方法。

从领域适应的角度来看，我们将元学习视为以(D^{train}, h^*)为训练样本对的监督式机器学习，其处理的基本单位不再是样本x而是任务T。目前，小样本关系分类都是同构迁移学习，因此源领域与目标领域任务的特征空间相同$\mathcal{T}_S = \mathcal{T}_T$，任务的分布不同$p(T_S) \neq p(T_T)$。但无论是源领域还是目标领域，其最终目的都是学习一个对应于任务T的基学习器h^*，即两个领域的元任务(Meta-Task)相同。因此，小样本领域适应实际上应称为元学习领域适应，其本质是将元学习器从源领域适应到目标领域。但是，如果我们希望利用传统机器学习中的领域适应思想来解决元学习领域适应问题，需要面对两个挑战：

- 如何获取目标领域的任务?
  在传统领域适应中，为了将模型适应到目标领域，需要目标领域的样本(无论有无标签)。对应元领域适应，则需要目标领域的任务。由于元训练集与元测试集类别互斥，因此，目标领域的任务是未知的。如何从目标领域的无标注样本中构建合理的任务，是元学习领域适应的第一个挑战。一种最直观的方法是对目标领域无标签数据进行聚类，核心问题在

于特征的抽取。从Table 1发现，在源领域训练的元学习器，虽然在目标领域数据集上的表现有大幅下降，但也有一定的效果。因此，可以利用源领域的元学习器辅助目标领域聚类。Cong et al. (2020)从对抗训练的角度出发，通过最小熵原理保证目标领域的聚类效果。

- 如何抽取任务特征?
在传统领域适应中，源领域与目标领域的输出空间相同，但是输入的分布不同，一种有效的方法是抽取领域无关的样本特征。尽管，有研究者通过对抗训练，抽取样本层面的领域无关特征来解决元领域适应问题(Gao et al., 2019b)。但在元学习模型处理的基本单位为任务。一个任务并不只包含样本这一个属性。任务中类别之间的相似度，也决定了这个任务的难易程度。因此，如何合理地表达一个任务的特征是元学习领域适应的第二个挑战。同时，当前基于度量的元学习方法本质上是在抽取同领域任务间的通用特征，如果在此基础上同时抽取领域无关的特征，如何保证最终抽取的特征的辨识度能够满足分类需要也有待解决。在保证集合无序性条件下，一种简单的获取任务特征的方法是统计法，如对支持集向量逐元素取均值、求和、求积、求几何平均或取最值(Edwards and Storkey, 2017; Li et al., 2019; Oreshkin et al., 2018)。为了抓取任务中的类别特征及样本数量，Lee et al. (2020)则采用更高阶的统计特征，如方差、偏度和峰度，并对DeepSets(Zaheer et al., 2017)进行了改进。除此之外，根据支持集向量构造无向图，通过图嵌入方法也能获取任务特征。

5.2 未来的研究趋势

- 多模态多领域泛化
无论是从领域适应的角度，还是从小-小样本学习的角度，解决单个领域元数据不足的方法都是迁移其它领域的知识。领域适应方法从单领域对单领域的适应方向解决问题，但需要我们获取目标领域的任务。小-小样本学习从数据增强的角度，直接利用多个领域的小样本关系数据集。但从领域泛化的角度出发，训练一个可以从多领域泛化到多领域的元学习器，就避免了获取大量单个领域任务或样本的麻烦。尽管每个领域的元训练集样本量不大，但是多个领域合成的元训练集在一定程度上也满足了元学习器的训练要求(Guo et al., 2019; Triantafillou et al., 2019)。此外，除了迁移同构领域之间的知识，异构领域可能包含更多的监督信息。利用多模态信息训练元学习器也能缓解单个领域元训练集不足的问题。

- 预训练语言模型压缩
预训练语言模型被证明很适合处理小样本学习任务(Brown et al., 2020)。但是，庞大的参数量以及所需的算力，限制了其在一些线下场景的应用。而且，随着参数量的降低，其在小样本任务上效果也会出现下降。如何在不损失模型效果的情况下，压缩模型的大小，是未来的一个发展方向。

- 更合理的小样本学习设定
目前大部分小样本关系分类模型的本质是元学习在极端小样本设定下的应用(N-way K-shot)。一方面，从定义上来讲，小样本问题并不等同于元学习问题。另一方面，在真实场景中，任务的类别数N与其包含的样本数K并不是固定的(Lee et al., 2020)。近来，有研究者发现最朴素的微调方法，在小样本任务上超越了元学习方法(Chen et al., 2019; Chen et al., 2020)，也有研究者分别从理论与实验的角度证明了学习一个好的表示对小样本任务至关重要(Tian et al., 2020; Du et al., 2020)。因此，元学习方法并不是解决小样本问题的唯一出路。如何确立更接近真实场景的小样本学习设定也需要进一步研究。

6 总结

在这篇文章中，我们系统地梳理了小样本关系分类算法，从度量方法上，将现有方法分为基于原型的方法和基于分布式表示的方法。从是否利用额外信息的角度，将现有方法分为预训练式与非预训练式。基于原型的方法主要从特征抽取器的角度入手，根据小样本分类的特点对特征抽器作特定地设计。基于分布式的方法从句子层面和词的层面建模各自的分布表示。此外，本文介绍了稀缺资源场景下的小样本关系分类任务，指出当前用于这些任务的方法在一些应用场景的局限性。最后，针对这些局限性，展望了小样本关系分类未来的发展方向。

致谢

感谢各位匿名评审给出的意见与建议。本论文受北京市自然科学基金项目(4192057)资助。

参考文献

Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. 2016. Learning to learn by gradient descent by gradient descent. In *Advances in neural information processing systems*, pages 3981–3989.

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Yujia Bao, Menghua Wu, Shiyu Chang, and Regina Barzilay. 2020. Few-shot text classification with distributional signatures. In *International Conference on Learning Representations*.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Wei-Lun Chao, Han-Jia Ye, De-Chuan Zhan, Mark Campbell, and Kilian Q Weinberger. 2020. Revisiting meta-learning as supervised learning. *arXiv preprint arXiv:2002.00573*.

Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Wang, and Jia-Bin Huang. 2019. A closer look at few-shot classification. In *International Conference on Learning Representations*.

Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. 2020. A new meta-baseline for few-shot learning. *arXiv preprint arXiv:2003.04390*.

Xin Cong, Bowen Yu, Tingwen Liu, Shiyao Cui, Hengzhu Tang, and Bin Wang. 2020. Inductive unsupervised domain adaptation for few-shot classification via clustering. *arXiv preprint arXiv:2006.12816*.

Simon S Du, Wei Hu, Sham M Kakade, Jason D Lee, and Qi Lei. 2020. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*.

Markus Eberts. 2019. *Relation Extraction with Attention-based Transfer Learning*. Ph.D. thesis, Hochschule RheinMain, FB Design Informatik Medien, Informatik.

Harrison Edwards and Amos J. Storkey. 2017. Towards a neural statistician. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Miao Fan, Yeqi Bai, Mingming Sun, and Ping Li. 2019. Large margin prototypical network for few-shot relation classification with fine-grained features. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 2353–2356, New York, NY, USA. Association for Computing Machinery.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org.

Tianyu Gao, Xu Han, Zhiyuan Liu, and Maosong Sun. 2019a. Hybrid attention-based prototypical networks for noisy few-shot relation classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6407–6414.

Tianyu Gao, Xu Han, Hao Zhu, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2019b. Fewrel 2.0: Towards more challenging few-shot relation classification. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

Tianyu Gao, Xu Han, Ruobing Xie, Zhiyuan Liu, Fen Lin, Leyu Lin, and Maosong Sun. 2020. Neural snowball for few-shot relation learning. In *Proceedings of AAAI*.

Ruiying Geng, Binhua Li, Yongbin Li, Xiaodan Zhu, Ping Jian, and Jian Sun. 2019. Induction networks for few-shot text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3895–3904.

Ruiying Geng, Binhua Li, Yongbin Li, Jian Sun, and Xiaodan Zhu. 2020a. Dynamic memory induction networks for few-shot text classification. *arXiv preprint arXiv:2005.05727*.

Xiaoqing Geng, Xiwen Chen, and Kenny Q Zhu. 2020b. Mick: A meta-learning framework for few-shot relation classification with little training data. *arXiv preprint arXiv:2004.14164*.

Yunhui Guo, Noel CF Codella, Leonid Karlinsky, John R Smith, Tajana Rosing, and Rogerio Feris. 2019. A new benchmark for evaluation of cross-domain few-shot learning. *arXiv preprint arXiv:1912.07200*.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. Fewrel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Xu Han, Tianyu Gao, Yankai Lin, Hao Peng, Yaoliang Yang, Chaojun Xiao, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2020. More data, more relations, more context and more openness: A review and outlook for relation extraction. *arXiv preprint arXiv:2004.03186*.

Hae Beom Lee, Hayeon Lee, Donghyun Na, Saehoon Kim, Minseop Park, Eunho Yang, and Sung Ju Hwang. 2020. Learning to balance: Bayesian meta-learning for imbalanced and out-of-distribution tasks. In *ICLR*.

Ke Li and Jitendra Malik. 2016. Learning to optimize. *arXiv preprint arXiv:1606.01885*.

Hongyang Li, David Eigen, Samuel Dodge, Matthew Zeiler, and Xiaogang Wang. 2019. Finding task-relevant features for few-shot learning by category traversal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–10.

Li Fei-Fei, R. Fergus, and P. Perona. 2006. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, page 1003–1011, USA. Association for Computational Linguistics.

Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. 2017. A simple neural attentive meta-learner. *arXiv preprint arXiv:1707.03141*.

Tsendsuren Munkhdalai and Hong Yu. 2017. Meta networks. *Proceedings of machine learning research*, 70:2554–2563.

Abiola Obamuyide and Andreas Vlachos. 2019. Model-agnostic meta-learning for relation classification with limited supervision. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5873–5879, Florence, Italy, July. Association for Computational Linguistics.

Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. 2018. Tadam: Task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems*, pages 721–731.

Victor Garcia Satorras and Joan Bruna. 2017. Few-shot learning with graph neural networks. *arXiv preprint arXiv:1711.04043*.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4077–4087. Curran Associates, Inc.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448.

Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. 2020. Rethinking few-shot image classification: a good embedding is all you need? *arXiv preprint arXiv:2003.11539*.

Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, and Hugo Larochelle. 2019. Meta-dataset: A dataset of datasets for learning to learn from few examples.

Joaquin Vanschoren. 2018. Meta-learning: A survey. *arXiv preprint arXiv:1810.03548*.

Oriol Vinyals, Charles Blundell, Timothy P. Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2016. Matching networks for one shot learning. In *NIPS*.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2019a. Kepler: A unified model for knowledge embedding and pre-trained language representation.

Yaqing Wang, Quanming Yao, James Kwok, and Lionel M. Ni. 2019b. Generalizing from a few examples: A survey on few-shot learning. *arXiv preprint arXiv:1904.05046*.

Zihao Wang, Kwunping Lai, Piji Li, Lidong Bing, and Wai Lam. 2019c. Tackling long-tailed relations and uncommon entities in knowledge graph completion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 250–260, Hong Kong, China, November. Association for Computational Linguistics.

Jason Weston, Sumit Chopra, and Antoine Bordes. 2015. Memory networks. *CoRR*, abs/1410.3916.

Yuxiang Xie, Hua Xu, Jiaoe Li, Congcong Yang, and Kai Gao. 2020. Heterogeneous graph neural networks for noisy few-shot relation classification. *Knowledge-Based Systems*, 194:105548.

Zhi-Xiu Ye and Zhen-Hua Ling. 2019. Multi-level matching and aggregation network for few-shot relation classification. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Mo Yu, Xiaoxiao Guo, Jinfeng Yi, Shiyu Chang, Saloni Potdar, Yu Cheng, Gerald Tesauro, Haoyu Wang, and Bowen Zhou. 2018. Diverse few-shot text classification with multiple metrics. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan R Salakhutdinov, and Alexander J Smola. 2017. Deep sets. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 3391–3401. Curran Associates, Inc.

基于阅读理解框架的中文事件论元抽取*

陈敏，吴凡，王中卿，李培峰*，朱巧明
（苏州大学计算机科学与技术学院，苏州，江苏，215006）
{1131651415，944721805}@qq.com，{wangzq，pfli，qmzhu}@suda.edu.cn

摘要

传统的事件论元抽取方法把该任务当作句子中实体提及的多分类或序列标注任务，论元角色的类别在这些方法中只能作为向量表示，而忽略了论元角色的先验信息。实际上，论元角色的语义和论元本身有很大关系。对此，本文提议将其当作机器阅读理解任务，把论元角色表述为自然语言描述的问题，通过在上下文中回答这些问题来抽取论元。该方法更好地利用了论元角色类别的先验信息，在ACE2005中文语料上的实验证明了该方法的有效性。

关键词： 事件论元抽取；阅读理解；先验信息；BERT

Chinese Event Argument Extraction using Reading Comprehension Framework

Min Chen，Fan Wu，Zhongqing Wang，Peifeng Li*，Qiaoming Zhu
（School of Computer Science and Technology, Soochow University, Suzhou, Jiangsu, 215006）
{1131651415，944721805}@qq.com，{wangzq，pfli，qmzhu}@suda.edu.cn

Abstract

Traditional event argument extraction methods formulated this task as a multi-classification or sequence labeling task mentioned by entities in the sentence. In these methods, the category of argument roles can only be described as vectors, while their prior information are ignored. In fact, the semantics of argument role category is closely related with the argument itself. Therefore, this paper proposes to regard argument extraction as machine reading comprehension, with argument role described as natural language question，and the way to extract arguments is to answer these questions based on the context. this method can make better use of the prior information existed in argument role categories and its effectiveness is shown in the experiments of Chinese corpus of ACE 2005.

Keywords: Event argument extraction , Reading comprehension , Prior information , BERT

基金项目：国家自然科学基金(61772354，61836007)；国家自然科学基金青年基金项目(61806137)；江苏高校优势学科建设工程资助项目。

1 引言

作为信息抽取（Information extraction）中的重要子任务，事件（Event）抽取是指从描述事件信息的文本中识别并抽取出包含的事件信息，并以结构化的形式呈现出来。事件抽取任务一般分为2个步骤，触发词（Trigger）抽取和论元（Argument）抽取。触发词抽取是根据上下文识别出触发词并判断其事件类型（Event type）；论元抽取是根据事件类型，抽取出参与事件的论元，并分配论元角色（Argument role）。在ACE2005数据集中，定义了33种事件子类型（8种事件大类）和35种论元角色。例1给出了数据集中包含1个触发词和3个论元角色的事件句。

例1：法官(A1)随即判(E1)被告(A2)7年预防性监禁(A3)。

触发词抽取部分需要识别出触发词E1，其对应的事件类型为宣判（Sentence）。论元抽取部分需要识别出参与宣判事件的论元并分配对应的角色。该事件的论元包括A1、A2和A3，分别对应角色审裁官（Adjudicator）、被告（Defendant）和判决结果（Sentence）。

当前中文事件抽取研究更多的是解决触发词抽取问题 (Feng et al., 2018; Lin et al., 2018; Ding et al., 2019; Xiangyu et al., 2019)，而针对中文论元抽取的工作相对较少。 Zeng et al. (2016)利用CNN和BiLSTM捕获句子和词汇信息，然后把论元抽取视为实体提及的多分类任务。 He and Duan (2019)利用条件随机场（CRF）和多任务学习的框架，把论元抽取视为序列标注任务。尽管这种多分类或序列标注的方式被认为是事件抽取的一个很好地解决办法，但是这样的做法仍然存在问题，论元角色标签本身的语义信息和论元存在着重要关系，现有的研究工作并不能利用论元角色标签本身的先验信息。如例1中，判决结果（Sentence）这类论元角色出现频率较低，而这个类别在多分类或序列标注训练中，仅被视为交叉熵中的一个独热向量，这种不清楚抽取什么往往导致劣质的性能。

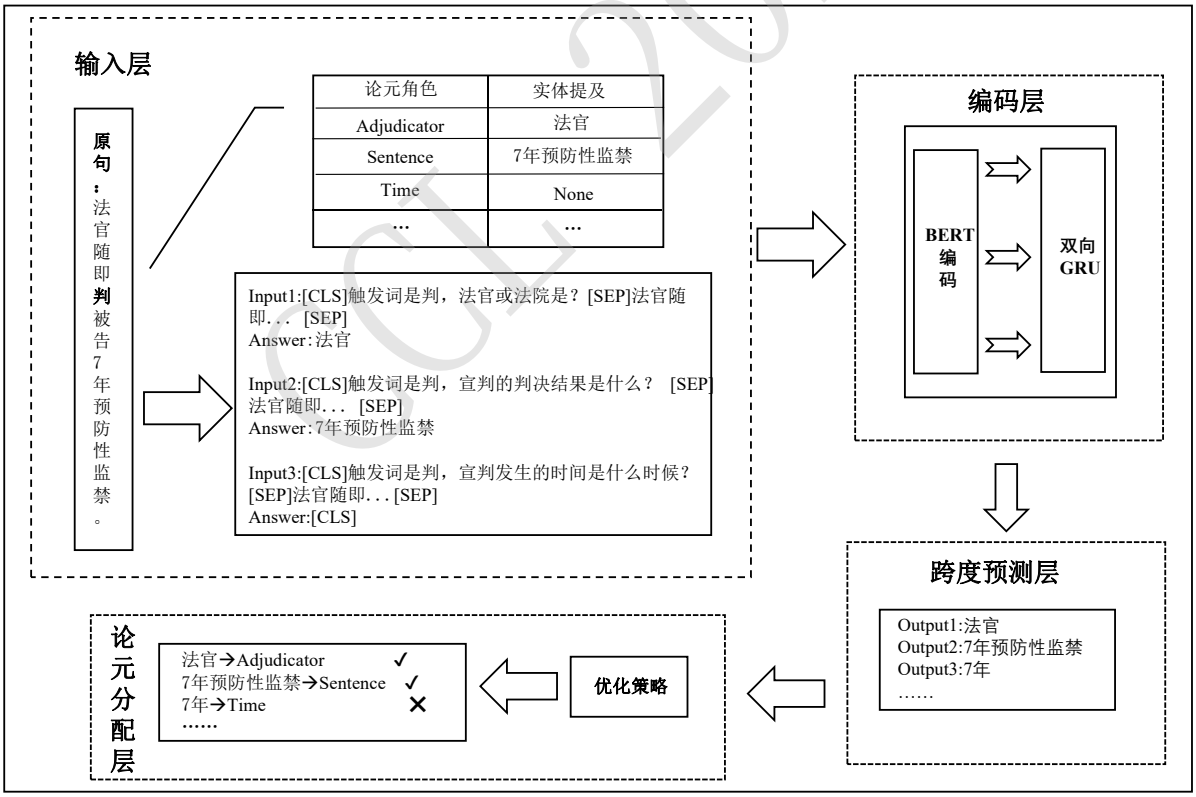


图 1. 论元抽取总体流程

本文的工作主要研究中文事件抽取中的论元抽取。针对论元抽取存在的上述问题，提出了基于BERT阅读理解框架的论元抽取方法，将论元抽取视为完型填空式的机器阅读理解

（Machine reading comprehension）任务。该方法的总体流程如图1所示。如想要抽取的角色为判决结果（Sentence），通过回答问题“触发词是判，宣判的判决结果是什么？”来预测该角色对应的论元“7年预防性监禁”，从而实现对该论元的识别和角色分配。这样的方式可以编码论元角色的先验信息，能够有效抽取出论元角色类别较少的论元。

总的来说，本文的方法利用已知的事件模式信息，将不同事件类型下的论元特征表述为自然语言问题，论元通过在事件句的上下文中回答这些问题来完成抽取。该方法通过BERT预训练模型学习问题和句子的初始隐向量表示，利用双向GRU更好的学习句子的上下文特征，然后对每个字进行二分类确定论元的跨度，采用合理的规则优化论元跨度，最终利用已知的实体提及完成论元角色识别和分配。在ACE2005中文语料上的实验证明，本文提出的基于阅读理解框架的论元抽取方法，优于传统的多分类或序列标注的方法，验证了阅读理解方式对论元抽取任务的有效性。

2 相关工作

事件抽取一直以来都是自然语言处理研究者们关注的重点领域。大多数工作把事件抽取看成两个阶段的问题，包括事件触发词抽取和论元抽取。触发词抽取工作近年来已经取得了很大的发展，论元抽取成为了事件抽取发展的瓶颈。目前论元抽取相关研究大部分是面向英文文本，中文论元抽取的发展较为缓慢。

在英文上，传统的基于特征表示的方法依靠人工设计的特征和模式。 Liao and Grishman (2010)提出跨文档的方法来利用全局信息和其他事件信息。 Hong et al. (2011)充分利用了事件句中实体类型的一致性特征，提出一种跨实体推理方法来提高事件抽取性能。 Li et al. (2013)提出了一种基于结构预测的联合框架，合并全局特征，显式地捕获多个触发词和论元的依赖关系。随着神经网络的流行，研究者们开始利用预训练好的词向量作为初始化表示 Mikolov et al. (2013a; Mikolov et al. (2013b)，进而建模单词的语义信息和语法信息。 Chen et al. (2015)对普通卷积神经网络做出改进，提出一种动态多池化卷积神经网络模型（DMCNN），把事件抽取看做两个阶段的多分类任务，先执行触发词分类，再执行论元分类，很好的解决了一个句子中包含多个事件的问题，但没有利用好触发词和论元之间的语义。 Nguyen et al. (2016)通过循环神经网络（RNN）学习句子表示，联合预测触发词和论元，增加了离散特征。为了捕获触发词和论元之间的依赖关系，引入记忆向量和记忆矩阵来存储在标记过程中的预测信息。 Liu et al. (2018)提出了一种新颖的联合多个事件提取框架（JMEE），通过引入句法短弧来增强信息流动，以解决句子编码的长距离依赖问题，利用基于注意力的图卷积网络来模型化图信息，从而联合抽取多个事件触发词和论元。 Wang et al. (2019b)在DMCNN的基础上，提出了一种分层模块化的论元抽取模型，该模型采用灵活的模块网络（Modular networks），利用了论元角色相关的层次概念，作为有效的归纳偏置，不同论元角色共享相同的高层次的单元模块，有助于更好地抽取出特定的事件论元。

随着深度学习的进一步发展，一些先进的技术也被用于英文事件抽取，包括零样本学习 Huang et al. (2018)、远程监督 Chen et al. (2017)、BERT预训练模型 Devlin et al. (2019)等。

相对于英文论元抽取，中文论元抽取的工作发展较缓，中文需要分词、缺少时态等自身特点给该任务带来一定的挑战。尽管如此，近年来也取得了一些进展。传统方法的工作更多的在挖掘语义和语法特征，很大程度上依赖于手工制作的特征和模式。 Li and Zhou (2012)引入形态结构来表示隐含在触发词内部的组合语义，提出了一个结合了中文词语的形态结构和义原去推测未知触发词的方法，明显提升了事件抽取的召回率。 Chen and Ng (2012)利用局部和全局特征共同抽取触发词和论元。 Zhu et al. (2015)利用事件之间的关系来学习实体扮演特定角色的概率，提出了基于马尔可夫逻辑网络的事件论元推理方法。 He and Duan (2019)将事件抽取看作序列标注任务，并考虑到数据稀疏问题，对不同事件子类进行互增强，提出基于CRF的多任务学习事件抽取联合模型。神经网络发展起来后， Zeng et al. (2016)提出了一种基于LSTM和CNN的卷积双向LSTM神经网络模型，利用BiLSTM和CNN分别编码句子级别信息和局部词汇特征。

随着预训练语言模型的发展，深度学习提高了许多自然语言处理的性能。很多自然语言理解的任务可以转换为机器阅读理解任务 Mccann et al. (2018)，如文本分类、关系抽取、事件抽取、情感分析、文本蕴含、语言推理、语义角色标注等。机器阅读理解任务是从给定问题的段落中提取答案。将NLP任务转为阅读理解任务成为了新的趋势。 Gardner et al. (2019)提出了使

用问答作为特定任务的格式的三种动机，即满足人类信息需求，探查系统对某些上下文的理解以及将学习到的参数从一项任务转移到另一项任务。 Li et al. (2019)将实体关系抽取视为一种多轮问答任务，每种实体和关系生成不同的问答模板，这些实体和关系可以通过回答这些模板化的问题来进行抽取。 Li et al. (2020)提出使用机器阅读理解框架代替序列标注模型，统一处理嵌套与非嵌套命名实体识别问题，在这种情况下，文本中实体的提取被形式化为回答问题，比如“文本中提到了哪个人？”。

3 基于阅读理解框架的论元抽取

受 Li et al. (2020)工作的启发，本文提出了基于阅读理解框架的论元抽取方法。在标准的机器阅读理解设置中，给定一个问题$Q = \{Q_1, Q_2, \cdots, , Q_{N_q}\}$（$N_q$表示问题Q中的字数），上下文$S = \{S_1, S_2, \cdots, S_{N_c}\}$（$N_c$表示句子S中的字数），模型从给出问题的段落中提取答案跨度。该任务可以形式化为两个多分类任务，即预测给定问题的答案跨度的开始位置和结束位置。本文的方法也基于这种设置，该方法的流程和模型如图1和图2所示。

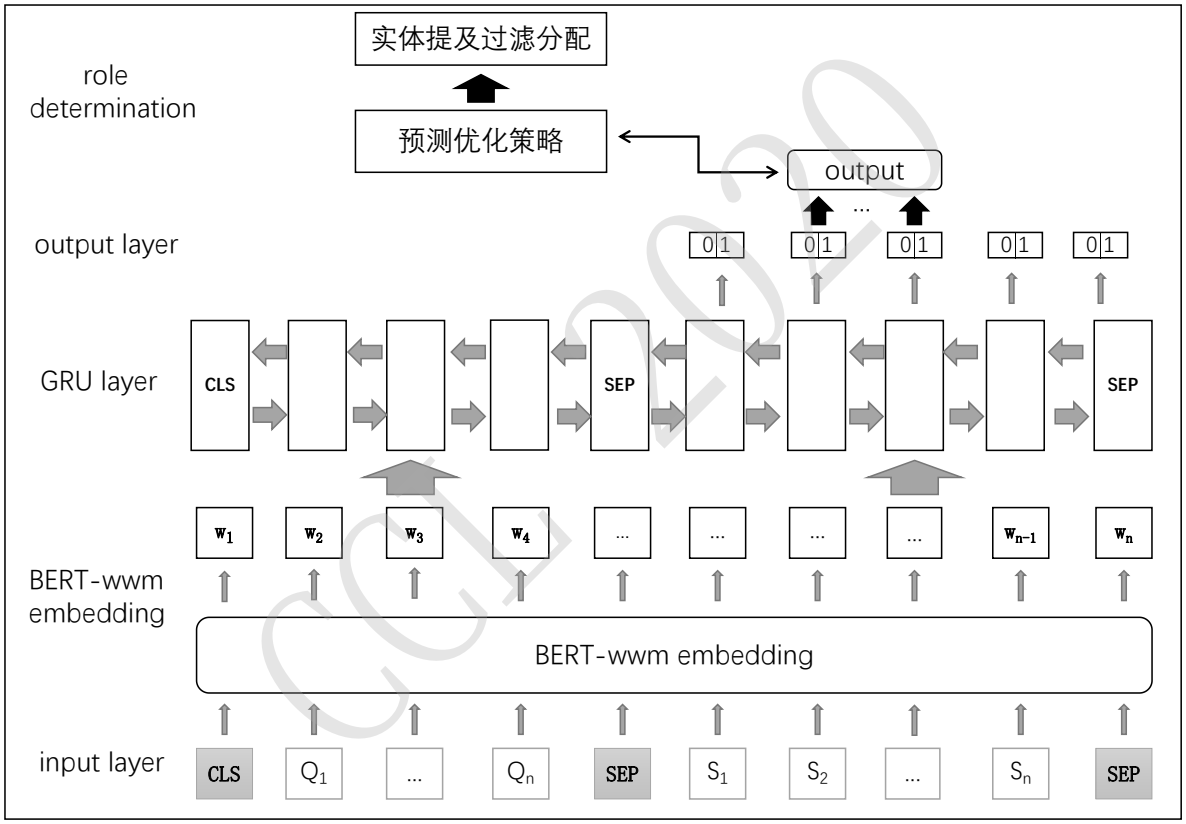


图 2. 基于阅读理解框架的论元抽取模型

主要包含四个部分：1）输入层，2）编码层，3）跨度预测层，4）论元分配层。其中，输入层是按照机器阅读理解的设置，利用本文采用的语料中的事件模式信息生成具有论元表征的问题和原句子作为初始输入表示；编码层通过BERT预训练模型编码字级别特征，利用双向GRU学习序列特征；跨度预测层根据编码层的输出，对每个字进行二分类来确定答案的跨度；论元分配层利用实体提及过滤抽取结果，最后给实体提及分配论元角色。

3.1 模型输入层

BERT模型的输入序列为句子对所对应的embeddings。句子对包含问题和事件句，并有特殊分隔符“[SEP]”分隔。问题由具有论元表征的论元角色标签构成，事件句是触发词抽取结果中包含事件的文本。同BERT的其他下游任务一样，所有的输入序列的第一个token必须为特殊分类嵌入符“[CLS]”，同时输入序列为字向量、位置向量和句子向量之和。模型的具体输入形式如

下：

$$[CLS]...Question...[SEP]...Sentence...[SEP]$$

其中，问题表示的语义信息是很重要的，因为该方法中问题编码了关于论元角色标签的先验知识，并对最终结果有重大影响。本文利用事件模式信息，统计触发词对应事件类型存在的论元角色(这种对应是已知且确定的)，试验了不同问题构成的效果。以3种事件类型为例，事件模式信息如表1所示，不同的问题模板如表2所示，其中Time-*表示与时间相关的论元角色，包括Time-Within、Time-Ending、Time-Starting等。

以受伤事件类型对应的施事者（Agent）角色为例，模板1（伪问题）使用论元角色作为问题，问题设置为“施事者”；模板2（触发词+伪问题）的加入触发词信息，句子中的触发词可以表示触发词信息和触发词位置特征，这也是模型可以学习到的重要特征，问题设置为“触发词是[Trigger]的施事者”（其中[Trigger]表示该事件类型对应的触发词）；模板3（触发词+自然问题）利用ACE2005中文语料库中的注释信息，根据事件类型和论元角色生成了更自然的问题（完整的注释问题参见附录A：注释问题），施事者（Agent）这一角色在受伤类型下扮演的是该事件下造成伤害的人，问题设置为“触发词是[Trigger]，造成伤害的人是谁？”。本文的实验验证了模板3的问题设置是最合理的。

事件类型	论元角色
受伤(Injure)	Agent Victim Instrument Time-* Place
结婚(Marry)	Person Time-* Place
攻击(Attack)	Agent Target Instrument Time-* Place
...	...

表 1. 事件模式表

论元角色	模板1 （伪问题）	模板2 （触发词+伪问题）	模板3 （触发词+自然问题）
Agent	施事者	触发词是[Trigger]的施事者	触发词是[Trigger]， 造成伤害的人是谁?
Victim	受害者	触发词是[Trigger]的受害者	触发词是[Trigger]， 受害者是谁?
Instrument	工具	触发词是[Trigger]的工具	触发词是[Trigger]， 造成伤害的工具或装置是什么?
...	...	...	...

表 2. 不同的问题模板（以Injure事件为例）

3.2 模型编码层

编码层的主体包括BERT和GRU两部分。

BERT在自然语言处理领域具有里程碑的意义。BERT本质上是通过在大量语料的基础上利用自监督学习的方法为每个字或词学习一个好的特征表示。它使用Transformer捕捉语句中的双向关系，使用遮蔽语言模型（MLM）和下一句预测的多任务训练目标。MLM是指在训练时在输入语料上随机遮蔽（mask）掉一些单词，然后通过的上下文预测该单词，这样的预训练方式能更好的表示语义特征。在谷歌发布的BERT版本中，中文是以字为粒度进行切分，没有考虑到传统NLP中的中文分词。本文采用哈尔滨工业大学发布的改进版本（BERT-wwm）(Cui et al., 2019)进行编码，将全词Mask的方法应用在了中文中，即对同属于一个词的汉字mask而不是对单个字的mask。同BERT-base一样，该模型采用12个Transformer Encoder堆叠而成的结构，每一层使用12个独立的注意力机制，包含768个隐层向量。注意力层增加多头注意力机制（Multi-Head Attention），扩展了模型专注于不同位置的能力。多头注意力模块的计算如下所示。

$$Attention(Q,K,V) = softmax(\frac{QK^t}{\sqrt{d_k}})V \tag{1}$$

$$MultiHead(Q,K,V) = [head_1,\cdots,head_h]W^O \tag{2}$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{3}$$

多头注意力机制用来学习每个字与其他字的依赖关系和上下文语义，然后通过前馈神经网络对attention计算后的输入进行变换，最终得到序列的全局信息。对于给定的输入序列$X = \{x_1, x_2, \cdots, x_n\}$，编码层BERT部分的输出最后一层Transformer的隐层向量，表示为$W = \{w_1, w_2, \cdots, w_n\}$。为了更好的学习句子上下文的序列特征，将BERT部分的输出再经过一个双向GRU模型，它可以继承BERT的优点，同时捕获序列语义信息，获取序列的长距离依赖。双向GRU分别从正反两个方向对BERT的输出进行编码，各自得到一个隐层输出，前向GRU层表示为$\overrightarrow{GRU}(W) = [\overrightarrow{h_1}, \overrightarrow{h_2}, \cdots, \overrightarrow{h_n}]$，后向GRU层表示为$\overleftarrow{GRU}(W) = [\overleftarrow{h_1}, \overleftarrow{h_2}, \cdots, \overleftarrow{h_n}]$。最终，编码层的输出为两个方向的GRU隐层向量的拼接结果，表示为$h = [\overrightarrow{h_i}, \overleftarrow{h_i}]$。

3.3 跨度预测层

预测层接收编码层的的隐层向量矩阵，答案跨度的预测主要包括开始位置和结束位置的确定，如果答案为空，把BERT输入层的第一个token“[CLS]”作为正确答案。模型在微调期间需要学习的参数是就是每个token作为答案开始位置（start span）和答案结束位置（end span）的向量，隐层向量经过softmax归一化后进行多个二分类，来获得每一个token分别作为开始位置和结束位置的概率，采用概率最高的区间作为预测结果。具体的计算如下。

$$P_{start} = softmax(E \cdot T_{start}) \tag{4}$$

$$P_{end} = softmax(E \cdot T_{end}) \tag{5}$$

$$I_{start} = argmax(P_{start}) \tag{6}$$

$$I_{end} = argmax(P_{end}) \tag{7}$$

其中$E(\ E \in \mathbb{R}^{n\times d}$，$n$为序列的长度，$d$为编码层的输出维度)是编码层输出的隐层向量矩阵；$T(\ T \in \mathbb{R}^{d\times 2})$即为需要学习的新参数；$P(\ P \in \mathbb{R}^{d\times 2})$为输出概率；$I(I \in [0, n-1])$为输出索引。

实验中采用二类交叉熵作为损失函数，在训练过程中，使用Adam优化器优化模型参数，通过最小化交叉熵损失完成训练调优，二类交叉熵具体计算如下。

$$L = -\frac{1}{N}\sum_{i=1}^{N} y_i log(\hat{y}_i) \tag{8}$$

$$L_{all} = L_{start} + L_{end} \tag{9}$$

其中，N表示序列的长度；y_i表示样本i预测为正的概率；L_{start}和L_{end}分别为开始位置和结束位置的损失。

3.4 论元分配层

传统的论元抽取都是已知实体提及的，本文的方法也一样，利用已知的实体提及完成论元角色分配工作。如果一个问题的预测结果与标准答案相同，则该答案的实体被正确分配角色；如果预测出的实体与答案中不同，则该实体认为错误分配了问题中的角色；如果预测的结果没有与实体匹配，判定该句中没有这个角色。

此外，该部分增加了优化策略，用以解决实体不完全匹配的问题。根据标准结果，匹配特定长度的相同开头或结尾的最长实体作为优化后的抽取结果。例如，在标准结果中的实体提及为“26岁”、“人”，预测结果分别为“26岁的时候”、“全家人”。这样的抽取结果也可以判定为正确的抽取。经过预测优化策略后，再根据实体分配不同的论元角色，最终提高论元抽取的性能。

4 实验

4.1 实验数据与评价方法

本文实验基于ACE2005 中文语料库，包含新闻专线、广播、微博等数据。每条数据包含触发词、实体、论元角色标签等标注信息。本文采用文献 (Zhu et al., 2015; He and Duan, 2019)相同的数据划分方法，从语料库中随机抽取567篇文档作为训练集，66 篇文档作为测试集，并保留训练集中的33 篇文档作为开发集。评判的标准同前人工作一样，一个论元被正确识别当且仅当该论元在文本的位置和类型与标准标注文档中的候选论元的位置和类型完全匹配。采用精确率(P)，召回率(R)，F1值作为本文的评价指标，具体计算如下。

$$P = \frac{TP}{TP + FP} \tag{10}$$

$$R = \frac{TP}{TP + FN} \tag{11}$$

$$F1 = \frac{2 \times P \times R}{P + R} \tag{12}$$

其中，TP为担任角色的实体被正确识别出的个数，FP为角色为None的实体被错误识别的个数，FN为担任角色的实体被错误识别的个数。

4.2 实验参数设置

本文采用哈工大版本的BERT预训练模型（BERT-wwm），其参数字向量维度为768，Transformer层数为12，实验的相关参数设置如表3所示。

参数名	参数值
BERT-wwm基础参数	默认值
GRU隐层维度	500
句子最大长度	400
批处理大小	8
训练轮数	5
学习率	1e-5

表 3. 实验参数设置表

4.3 实验结果

本文工作是针对论元抽取任务，触发词抽取不是重点工作。论元抽取的工作是基于触发词抽取的结果来做，本文的触发词抽取模型利用BERT微调 Xiangyu et al. (2019)的结果，其事件类型分类的精确率(P)为73.9%，召回率(R)为63.8%，F1值为68.5%。

本文主要进行了两组实验对比，一是将本文提出的方法与基准系统进行对比实验，二是设置不同问题策略的对比实验。

4.3.1 与基准系统相比

本文将提出的基于阅读理解框架的方法与现有的论元抽取方法进行了对比。结果如表4所示。

- Rich-C： Chen and Ng (2012)提出的基于特征的模型，该模型针对中文的特殊性开发了一些手工特征，以共同提取事件触发词和论元角色。
- JRNN： Nguyen et al. (2016)提出的一种基于神经网络的模型。它利用双向RNN和手动设计的特征来实现论元抽取。
- C-BiLSTM： Zeng et al. (2016)提出的一种结合LSTM和CNN的卷积双向LSTM神经网络来捕获句级和词汇信息，把论元抽取看成多分类任务的方法。

• MTL-CRF： He and Duan (2019)提出的基于CRF的方法，设计了一个有效挖掘不同事件之间论元相互关系的多任务学习的序列标注模型，联合标注触发词和论元，降低了管道模型带来的级联错误，并没有利用复杂的神经网络，其精确率有明显的提升，但召回率较低。

• DMBERT： Wang et al. (2019a)提出的有效利用预先训练语言模型的方法并使用动态多池化方法来聚合特征。它不同与DMCNN的是利用BERT提取字级别信息和句子信息，获得了较大的性能提升。本文复现了该模型，作为BERT基准。为了公平比较，触发词抽取部分沿用本文的触发词基准结果。

• MRC-EAE：即本文提出的基于BERT并结合双向GRU的阅读理解模型，本文把传统的论元抽取任务建模成SQUAD风格的机器阅读理解任务，使用了BERT编码问题和句子信息，利用了论元角色的先验信息，同时使用GRU学习句子序列特征。

实验系统	论元识别(%)			论元分类(%)		
	P	R	F1	P	R	F1
Rich-C	57.3	43.6	49.5	51.6	39.2	44.6
JRNN	49.6	53.2	51.3	43.1	45.6	44.3
C-BiLSTM	53.0	52.2	52.6	47.3	46.6	46.9
MTL-CRF	70.4	48.6	57.4	**66.6**	44.1	53.1
DMBERT	65.2	51.0	57.3	59.1	47.8	52.8
MRC-EAE(ours)	59.4	55.1	57.2	56.5	**52.3**	**54.4**

表 4. 论元抽取实验结果

从表4中的实验结果可以看出，本文提出的基于阅读理解框架并结合双向GRU的方法优于其他方法。对比多任务学习的序列标注方法MTL-CRF和基于BERT的动态多池化模型DMBERT，本文提出的方法在召回率和F1值上有明显提升，召回率分别提升了8.2%和4.5%，F1值分别提升了1.3%和1.6%。传统的MTL-CRF方法联合抽取触发词和论元，虽然可以降低级联错误，但是这种联合训练的序列标注增加了很多标签，致使类别稀疏，导致召回率较低。同样，在多分类任务DMBERT中，对于有些论元角色较少的类别很难被识别出。而本文提出的方法利用BERT和双向GRU编码，BERT的多头注意力机制和两句输入能充分获取输入文本的语义信息，将问题和句子之间的语义关系充分捕捉，并在句子中获取最终的答案位置。这种阅读理解的方法能够通过问题编码了论元角色的先验信息，这是以往工作中没有利用的重要特征。由于引入论元角色的先验信息，可以有效的识别出角色较少的但是标签有语义区分的类别，如交通工具（Vehicle）、原告（Plaintiff）、卖方（Seller）等，表5给出了5个低频论元在DMBERT和本文方法的结果对比，从结果可以看出，本文提出的方法在这几种少类别的角色标签上有明显的提升效果，更加验证了该方法的有效性。

实验系统	低频论元(测试集类别数目)F1(%)					all
	Adjudicator(19)	Vehicle(12)	Plaintiff(12)	Sentence(10)	Seller(2)	-
DMBERT	45.7	47.1	13.3	41.7	0.0	39.1
ours	52.6	50.0	42.1	59.2	40.0	51.2

表 5. 低频论元角色对比结果

4.3.2 阅读理解方式不同策略的对比

为了验证编码不同论元角色标签的先验信息对模型的影响，本文设置了不同问题模板进行了消融研究，不同的问题模板设置在第二节给出。实验对比结果如表6所示。

模板1的问题设置仅仅代表论元角色的语义，在多事件类型的句子中，模型不能正确抽取对应事件类型的论元；模板2的问题设置方式加入了触发词，可以表示句中需要抽取论元具体的触发词语义和触发词的位置信息，但对于论元的描述不够具体；模板3生成了更自然的问题，这种提问策略加入触发词信息的同时融合事件类型信息和论元角色先验信息。从表中实验结果可以发现，性能最好的问题模板3相比模板1和模板2在F1值上分别提升了3.2%和1.7%。当模板3的问

策略	论元识别(%)			论元分类(%)		
	P	R	F1	P	R	F1
模板1	57.4	51.0	54.0	53.3	47.4	50.2
模板2	59.1	53.0	55.9	54.7	49.0	51.7
模板3	60.2	53.2	56.5	**57.0**	50.3	53.4
模板3$_{\text{-触发词}}$	58.5	51.7	54.9	54.6	48.3	51.2
模板3$_{\text{optimized}}$	58.7	55.7	57.1	55.6	**52.7**	54.1
Final	59.4	55.1	57.2	56.5	52.3	**54.4**

表 6. 不同的策略对比结果

题设置去掉触发词时，性能下降了2.2%，这说明触发词信息的加入可以有效的判断答案的位置和与触发词关系更紧密的论元。此外，在模板3的基础上，对抽取的结果进行优化，在F1值上能提升0.7%；同时利用双向GRU的双向学习序列信息的能力，更好学习输入中问题和句子上下文的关系，在结果优化的基础上F1值提升了0.3%。

4.3.3 错误分析

对实验结果进一步分析发现，本文提出的方法仍然存在不足之处。一方面，本文利用的事件模式信息，存在某些事件句缺失论元角色的情况，即有的问题的答案为空，这种情况下模型往往会被错误预测。例如“法官随即判被告7年预防性监禁。”这一句中并不包含时间相关论元，但是实体提及“7年”会被模型误认为是时间的角色。另一方面，如果一个事件句中某个事件类型存在多个相同的论元角色，受限于本文阅读理解模型的设置，只能识别出其中的一个作为正确答案。例如“而就在吕传升接受记者访问的时候，突然接到了吕秀莲打来的电话，要吕传升暂时封口。”，句中包含2个会面对象（Entity）——“吕传升”和“记者”，模型往往只能学习到“吕传升”这个论元而“记者”被忽略。

5 结论

本文采用的基于阅读理解模型的论元抽取方法，把该任务形式化为回答不同的问题来实现不同论元角色的识别和分配，通过优化问题的质量来提升问题回答的性能。通过反复实验证明，这种完型填空式的抽取方式相比基准模型有了明显的提升，也能适用于事件抽取任务上。然而，本文的工作是基于句子级别的论元抽取，缺失了段落信息的句子往往丢失了很多重要的上下文信息。在下一步的研究工作中，还可以考虑基于篇章层面的阅读理解方式来提升论元抽取的效果。

参考文献

Chen Chen and Vincent Ng. 2012. Joint modeling for Chinese event extraction with rich linguistic features. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 529–544.

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 167–176.

Yubo Chen, Shulin Liu, Xiang Zhang, Kang Liu, and Jun Zhao. 2017. Automatically labeled data generation for large scale event extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 409–419.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-training with whole word masking for Chinese bert. *arXiv preprint arXiv:1906.08101*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Ning Ding, Ziran Li, Zhiyuan Liu, Haitao Zheng, and Zibo Lin. 2019. Event detection with trigger-aware lattice neural network. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 347–356.

Xiaocheng Feng, Bing Qin, and Ting Liu. 2018. A language-independent neural network for event detection. *Science China Information Sciences*, 61(9):092106.

Matt Gardner, Jonathan Berant, Hannaneh Hajishirzi, Alon Talmor, and Sewon Min. 2019. Question answering is a format; when is it useful? *arXiv preprint arXiv:1909.11291*.

Ruifang He and Shaoyang Duan. 2019. Joint Chinese event extraction based multi-task learning. *Journal of Software*, 4:1015–1030.

Yu Hong, Jianfeng Zhang, Bin Ma, Jianmin Yao, Guodong Zhou, and Qiaoming Zhu. 2011. Using cross-entity inference to improve event extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1127–1136.

Lifu Huang, Heng Ji, Kyunghyun Cho, Ido Dagan, Sebastian Riedel, and Clare Voss. 2018. Zero-shot transfer learning for event extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2160–2170.

Peifeng Li and Guodong Zhou. 2012. Employing morphological structures and sememes for Chinese event extraction. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 1619–1634.

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 73–82.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019. Entity-relation extraction as multi-turn question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1340–1350.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2020. A unified mrc framework for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5849–5859.

Shasha Liao and Ralph Grishman. 2010. Using document level cross-event inference to improve event extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 789–797.

Hongyu Lin, Yaojie Lu, Xianpei Han, and Le Sun. 2018. Nugget proposal networks for Chinese event detection. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1565–1574.

Xiao Liu, Zhunchen Luo, and He-Yan Huang. 2018. Jointly multiple events extraction via attention-based graph information aggregation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1247–1256.

Bryan Mccann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv: Computation and Language*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 27th Annual Conference on Neural Information Processing Systems*, pages 3111–3119.

Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 300–309.

Xiaozhi Wang, Xu Han, Zhiyuan Liu, Maosong Sun, and Peng Li. 2019a. Adversarial training for weakly supervised event detection. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 998–1008.

Xiaozhi Wang, Ziqi Wang, Xu Han, Zhiyuan Liu, Juanzi Li, Peng Li, Maosong Sun, Jie Zhou, and Xiang Ren. 2019b. HMEAE: Hierarchical modular event argument extraction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5781–5787.

Xi Xiangyu, Zhang Tong, Ye Wei, Zhang Jinglei, Xie Rui, and Zhang Shikun. 2019. A hybrid character representation for Chinese event detection. In *Proceedings of the 2019 Conference on International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Ying Zeng, Honghui Yang, Yansong Feng, Zheng Wang, and Dongyan Zhao. 2016. A convolution BiLSTM neural network model for Chinese event extraction. In *Proceedings of the Natural Language Understanding and Intelligent Applications*, pages 275–287.

Shaohua Zhu, Peifeng Li, and Qiaoming Zhu. 2015. A Chinese event argument inference approach based on entity semantics and event relevance. In *Proceedings of Workshop on Chinese Lexical Semantics*, pages 577–586.

附录A.注释问题

事件类型	论元角色	注释问题
declare-bankruptcy	Org	宣布破产的组织或公司是什么?
	Time	宣布破产的时间是什么时候?
	Place	宣布破产的地点在哪里?
end-org	Org	结束的组织或公司是什么?
	Time	结束组织或公司的时间是什么时候?
	Place	结束组织或公司的地点在哪里?
merge-org	Org	合并的组织或公司是什么?
	Time	合并组织或公司的时间是什么时候?
	Place	合并组织或公司发生的地点在哪里?
start-org	Org	成立的组织或公司是什么?
	Agent	组织的建立者是?
	Time	成立组织或公司的时间是什么时候?
	Place	成立组织或公司的地点在哪里?
attack	Target	攻击目标（包括意外目标）是什么?
	Attacker	攻击者是谁?
	Time	攻击发生的时间是什么时候?
	Place	攻击发生的地点在哪里?
	Instrument	攻击中使用的工具是什么?
demonstrate	Entity	示威游行的人或组织是?
	Place	游行的地点在哪里?
	Time	游行的时间是什么时候?
meet	Entity	会面的人或组织是?
	Place	会面的地点在哪里?
	Time	会面的时间是什么时候?
phone-write	Entity	打电话或写信交流的人或组织是?
	Place	打电话或写信交流的地点在哪里?
	Time	打电话或写信交流的时间是什么时候?

acquit	Defendant	被无罪释放的是谁?
	Place	无罪释放的地点在哪里?
	Adjudicator	法官或法院?
appeal	Plaintiff	上诉的原告是谁?
	Adjudicator	上诉的被告人是谁?
	Time	上诉发生的时间是什么时候?
	Place	上诉的地点在哪里?
arrest-jail	Person	被逮捕或被监禁的人是谁?
	Agent	实施逮捕的人是谁?
	Place	逮捕发生的地点在哪里?
	Time	逮捕的时间是什么时候?
	Crime	被逮捕人犯的罪行是什么?
charge-indict	Crime	指控或起诉的罪行是什么?
	Defendant	被指控或起诉的人是谁?
	Prosecutor	起诉人或法官是谁?
	Adjudicator	审裁官是谁?
	Time	指控的时间是什么时候?
convict	Defendant	被定罪的人是谁?
	Adjudicator	法官或法院?
	Crime	被判的罪行是什么?
execute	Person	被处决的人是谁?
	Agent	负责执行死刑的代理是?
	Crime	被判的罪行是什么?
extradite	Person	被引渡的人是谁?
	Origin	被引渡人的原所在地是哪里?
	Destination	引渡的目的地是哪里?
	Agent	引渡的代理人是?
fine	Crime	被罚款的罪行是什么?
	Entity	罚款的实体对象是什么?
	Adjudicator	执行罚款的对象是?
	Money	罚款数额是多少?
pardon	Defendant	被赦免的人或公司是?
	Adjudicator	赦免的官员或法官是?
	Time	赦免的时间是什么时候?
	Crime	赦免的罪行是什么?
release-parole	Person	被释放的人是谁?
	Time	释放的时间是什么时候?
	Entity	释放之前的劫持者是谁?
	Crime	被释放的人先前被关押的罪行是什么?
sentence	Defendant	被判刑的人是谁?
	Sentence	宣判的判决结果是什么?
	Crime	判决的罪行是什么?
	Adjudicator	法官或法院?
	Time	判决发生的时间是什么时候?
sue	Plaintiff	起诉代理,原告是谁?
	Defendant	被指控或起诉的人是谁?
	Adjudicator	法官或法院?

	Time	诉讼发生的时间是什么时候?
	Prosecutor	检察官是谁?
trial-hearing	Defendant	审判的被告人是谁?
	Adjudicator	法官或法院?
	Time	审判的时间是什么时候?
	Crime	审判的罪行是什么?
	Place	审判的地点在哪里?
be-born	Person	出生的人是谁?
	Place	出生的地点在哪里?
	Time	出生的时间是什么时候?
die	Victim	受害者是谁?
	Agent	致人死亡的人是谁?
	Time	死亡的时间是什么时候?
	Place	死亡的地点在哪里?
	Instrument	致人死亡的装置是什么?
divorce	Person	离婚的人是谁?
	Place	离婚的地点在哪里?
	Time	离婚的时间是什么时候?
Injure	Agent	进攻者，造成伤害的人是谁?
	Victim	受害者是谁?
	Instrument	造成伤害的装置是什么?
	Place	受伤的地点在哪里?
	Time	受伤的时间是什么时候?
marry	Person	结婚的人是谁?
	Place	结婚的地点是哪里?
	Time	结婚的时间是什么时候?
transport	Artifact	被运送的人或工件是什么?
	Destination	运送的目的地是哪里?
	Time	运送发生的时间是什么时候?
	Origin	运送的起源地是哪里?
	Agent	负责移动事件的代理是?
	Vehicle	使用的交通工具是什么?
elect	Person	被选举的人是谁?
	Time	选举的时间是什么时候?
	Position	选举的职位是什么?
	Entity	投票代理是?
end-position	Person	结束任职的人是谁?
	Time	卸任的时间是什么时候?
	Position	卸任的职位是什么?
	Entity	雇主是谁?
nominate	Person	结束任职的人是谁?
	Position	卸任的职位是什么?
	Agent	雇主是谁?
	Place	卸任的地点在哪里?
start-position	Person	被提名的人是谁?
	Position	被提名担任的职位是什么?
	Time	提名的时间是什么时候?

	Entity	任职实体是什么?
	Place	提名的地点在哪里?
transfer-money	Time	钱财转移的时间是什么时候?
	Beneficiary	在钱财转移中获益的是谁?
	Place	钱财转移发生的地点在哪里?
	Recipient	钱财交易的接受者是谁?
	Giver	捐赠人或公司是什么?
	Money	给予/捐赠/贷款的金额是多少?
transfer-ownership	Artifact	进行交易的物品或组织是什么?
	Seller	销售的代理是?
	Beneficiary	在交易中获益的人或组织是谁?
	Time	交易发生的时间是什么时候?
	Buyer	采购的代理是?

基于BERT的端到端中文篇章事件抽取

张洪宽　宋晖(✉)　王舒怡　徐波
（东华大学　计算机科学与技术学院，中国　上海　201620）
{2181729,2181754}@mail.dhu.edu.cn, {songhui,xubo}@dhu.edu.cn

摘要

篇章级事件抽取研究从整篇文档中检测事件，识别出事件包含的元素并赋予每个元素特定的角色。本文针对限定领域的中文文档提出了基于BERT的端到端模型，在模型的元素和角色识别中依次引入前序层输出的事件类型以及实体嵌入表示，增强文本的事件、元素和角色关联表示，提高篇章中各事件所属元素的识别精度。在此基础上利用标题信息和事件五元组的嵌入式表示，实现主从事件的划分及元素融合。实验证明本文的方法与现有工作相比具有明显的提升。

关键词： 篇章级别事件抽取；端到端；增强嵌入表示；主从事件

A BERT-based End-to-End Model for Chinese Document-level Event Extraction

Hongkuan Zhang, Hui Song(✉), Shuyi Wang, Bo Xu
(School of Computer Science and Techology, Donghua University, Shanghai,201620,China)
{2181729,2181754}@mail.dhu.edu.cn, {songhui,xubo}@dhu.edu.cn

Abstract

Document-level event extraction aims at discovering event mentions and extracting events which contain event arguments and their roles from texts. This paper proposes an end-to-end model for closed-domain based on BERT. We introduce the embedding of event type and entity nodes to the subsequent layer for event argument and role identification, which represents the relation between event, arguments and roles and improves the accuracy of classifying multi-event arguments. With the title, the quintuple of event, we calculate the master slave structure between multiple events with the embedding presentation. Experimental results show that our model outperforms the state of the art.

Keywords: Document-level Event Extraction , End-to-end , Enhancing embedding , Master slave event

1 引言

基金项目：国家自然科学基金青年项目（61906035），上海市青年科技英才扬帆计划项目（19YF1402300）

近年来互联网快速发展，网络媒体每天产生大量的新闻、公告等非结构化信息。信息抽取技术研究如何从海量的信息中快速有效地捕获有价值的信息，以帮助人们针对特定信息做分析、决策。事件抽取是信息抽取的分支，旨在从非结构化的自然语言文本中抽取出用户感兴趣的事件信息并以结构化的形式展示 (Ahn, 2006)。事件抽取在很多领域有着广泛的应用，如构建事件知识图谱、信息检索、自动问答以及辅助其他自然语言处理任务等。

事件抽取分为开放域和限定域事件抽取 (Wei and Wang, 2019)。开放域事件抽取研究通常没有领域范围限制，事件类型及事件的框架结构未知，主要利用无监督方法从文本中发现事件 (Piskorski et al., 2011; Ribeiro et al., 2017; Yu and Wu, 2018)。限定域事件抽取往往针对特定领域（如医疗、金融、司法等）的数据进行建模，识别用户感兴趣的信息。与开放域相比，限定域事件抽取有清晰的事件类型定义及对应的事件框架，能够获得具有实用价值的信息，近年来成为研究和应用的热点。

事件抽取从文本粒度上也可分为句子级别和篇章级别。句子级别事件抽取研究从句子中识别所关注的内容，首先采用深度神经网络，如动态多池化卷积神经网络 (Chen al et., 2015)、结合LSTM和CNN的卷积双向LSTM神经网络 (Zeng al et., 2016)等方法提取特征，然后识别事件元素和角色。事件及其元素识别通常采用管道式方法，分为两个子任务实现，忽略了他们之间的联系，容易导致错误传播问题。针对管道式模型存在的问题，Li (2013)和Nguyen (2016)采用联合模型捕获实体与事件之间的语义关系，同时识别事件和实体，提高了事件抽取的准确率。

篇章级的事件抽取旨在发现从整篇文档识别事件并提取出相应的事件元素，抽取困难之处在于文档中可能存在多个事件（如图1所示），事件元素分布在不同句子中，多个事件之间存在元素重叠（如图1所示）。

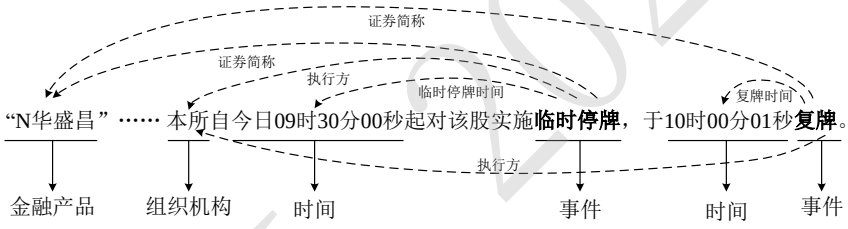


图1.文档中存在“临时停牌”与“复牌”两个事件，且共享事件元素“N华盛昌（角色为证券简称）”和“本所（角色为执行方）”。

目前篇章级的事件抽取大多采用管道式方法，即分别对事件和事件元素进行识别，然后判断元素的角色。如Yang (2018)基于句子抽取结果及文本特征发现核心事件，采用元素补齐策略得到篇章级别的事件信息。ZHONG (2019)在句子级抽取结果的基础上利用整数线性规划进行全局推理，融合共指事件的元素，实现篇章级事件抽取。由于管道模型本身的局限性，导致这些方法整体效果不理想。Yang (2016)提出一个端到端学习模型，通过采用联合因子图模型来联合学习每个事件内部的结构化信息、篇章内不同事件间的关系和实体信息，但是该工作使用了大量人工特征，不利于领域泛化。

Zheng (2019)采用端到端模型首先识别句子级别的事件元素，再利用二分类策略判定事件类型，然后将元素角色识别转化为有向无环图的生成过程。该文能够较好地处理单事件单实例和单事件多实例的样本，但在元素分布较分散且存在元素重合的多事件多实例样本上性能略差。针对以上问题，本文在已有篇章级的抽取研究工作基础上，提出了基于BERT (Devlin al et., 2019)的端到端事件抽取模型DLEMC（Document-Level End-to-end Model in Chinese），该模型无需将文章按句子分割分别处理，尽可能保留完整的文本，减少信息损失；在事件元素识别中引入了事件类型嵌入增强文本特征，并在元素角色分类中利用注意力机制引入事件类型和实体嵌入表示，以便准确识别事件元素及其在不同事件中的角色。

DLEMC模型分为4层：输入编码层、事件检测层、事件元素识别层以及元素角色识别层。输入编码层接收输入的文本信息并输出对应的嵌入表示；事件检测层采用多个分类器对同一个文本特征向量进行多标签事件分类；事件元素识别层通过引入事件特征学习不同事件中元素的语义信息，进行事件元素的识别；元素角色识别层利用注意力机制来提高模型对确定事件中每

个元素特征的关注度，判断其在对应事件中扮演的角色。最后基于事件的嵌入表示计算余弦相似度，进行主从事件划分及融合共指元素，得到篇章级结构化事件信息。实验证明DLEMC的性能与基准模型相比有明显的提升。

本文的贡献总结如下：

(1)本文依据金融领域上市公司公告组织了一个篇章级别事件抽取语料集，并针对该语料定义了事件及事件表示框架。

(2)本文构建了篇章级别的事件抽取模型，该模型采用端到端方式进行联合学习，同时对事件检测、事件元素识别及元素角色分类进行训练，通过实验证明了该模型的有效性。

(3)本文在事件元素识别中引入事件类型特征，以提高不同类型事件下的元素识别能力。为了更准确识别多事件文档中元素角色，我们将事件检测层输出的事件类型及事件元素的嵌入表示作为注意力引入模型的角色分类层，以提高不同事件类型下元素角色识别的准确率。

2 事件定义及表示

随着金融科技的发展，在金融领域每天都有海量的数据产生，金融事件抽取研究能够帮助人们进行金融风险监控、辅助投资决策、大数据分析等。本文研究的事件抽取需要提取事件类别及参与者，目前该领域的中文事件抽取研究缺乏数据支持，前人的相关研究大多没有公开数据集，研究的事件类型比较集中且类别较少(Yang (2018)4类，Zheng (2019)5类，去除重复后共5类。)，为扩充金融领域事件研究的数据，本文组织构建了一定规模的金融领域中文篇章事件抽取数据集，并依据自动内容抽取(Automatic Context Extraction，ACE)[0]定义的事件抽取任务，说明如下：

事件(event)：在某个时间点或时间段，一个或多个机构的金融产品的状态主动地或被动地发生了变化。

实体(entity)：语义类别中的一类或一组对象，本文讨论的实体包括命名实体、金融产品、时间和数值。

事件元素(event argument)：在事件中具有特定作用的实体。

元素角色(argument role)：事件元素在事件中承担的角色。

针对本文研究的金融公告信息，事件定义为$event = def(T, O, F, D, N)$ 其中T为事件类型，O、F、D、N为事件中的4类角色，分别表示组织机构、金融产品、时间、数值，每一类下有若干小类，共计22类事件角色。

(1)事件类型，事件所属的类别，如“临时停牌”、“复牌”、“上市交易”等。

(2)组织机构，参与事件的一类实体，如“东沣科技集团股份有限公司”。

(3)金融产品，金融领域中的相关产品，如“证券名称”，“证券简称”。

(4)时间，指事件发生的具体时间点或者事件持续发生且产生作用的时间间隔，如“10时00分01秒”。

(5)数值，衡量事件中某一属性具体量的多少，如“票面利率4%”、“标准交易单位10张”等。

其中事件类型、组织机构、金融产品和时间的实例在事件文本中一定会出现，数值实例不一定会出现。

3 事件抽取模型DLEMC

篇章级事件抽取研究识别文档中存在的事件和相关元素，并判断元素在事件中扮演的角色。给定文档集$doc = \{s_0, s_1, ..., s_{N_s}\}$，每篇文档包含标题句$s_0$和内容句$\{s_1, ..., s_{N_s}\}$，$N_s$为句子数量。模型DLEMC首先对文档标题和内容分别进行事件检测，得到文档包含的事件类型$\{t_0, t_1, t_2, \dots\}$,其中t_0为标题中的事件，其他为内容中的事件，然后识别出文档内容中每类事件的相关元素$\{e_1, e_2, \dots\}$及其对应的角色$\{role_1, role_2, \dots\}$。

DLEMC模型由4部分组成（如图2所示），包括输入编码层、事件检测层、事件元素识别层和元素角色识别层。

输入编码层，基于BERT对输入的句子进行编码，得到句子对应的向量以及句子中每个$token$的向量。

[0]http://projects.ldc.upenn.edu/ace

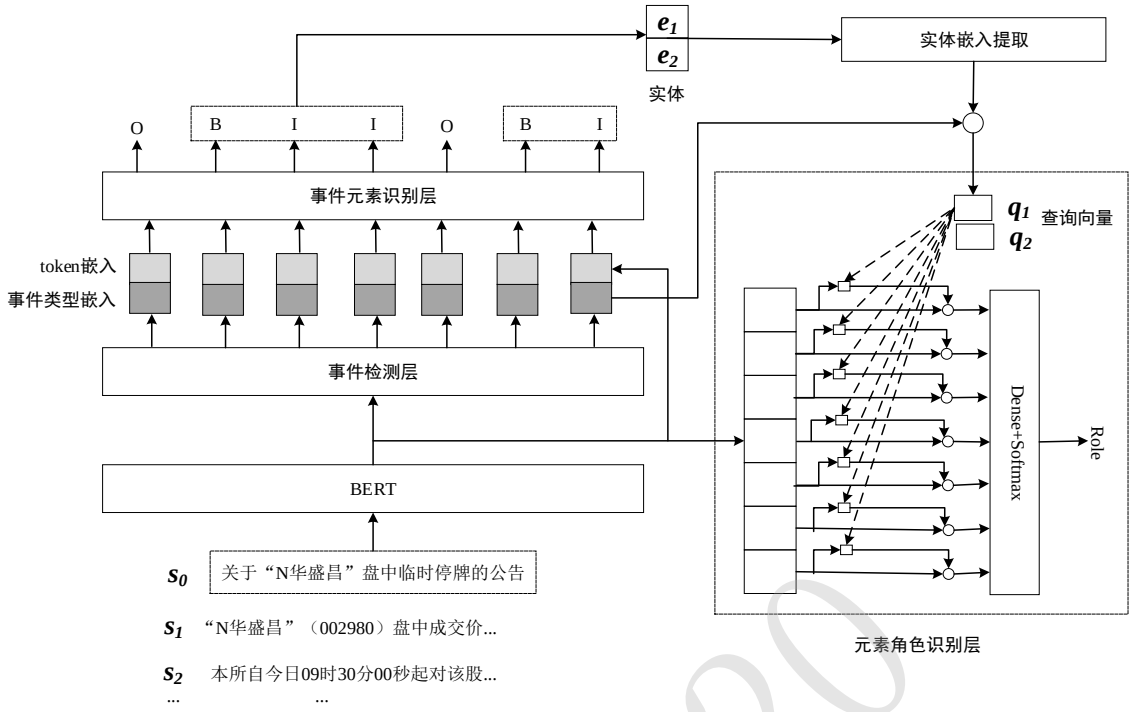


图2.篇章级事件抽取模型DLEMC

事件检测层，将编码层输出的句向量作为输入，预测该句中包含的事件，一个句子中可能存在多个事件。

事件元素识别层，识别句子中参与事件的实体。将句子的$token$向量与事件类型对应的向量进行拼接作为输入，预测每个$token$对应的BIO标签，从而识别出事件元素对应的实体。

元素角色识别层，对上一步识别出的确定事件类型下的实体进行角色分类。将事件类型t和实体e对应的嵌入表示求平均之后作为注意力的查询向量，重新计算$token$的向量表示，再对每个事件元素的角色进行识别。

模型训练时分别计算事件检测层、事件元素识别层以及元素角色识别层的损失，并将三者求和作为模型最终的优化目标。

3.1 输入编码层

本文采用预训练语言模型BERT对文档进行编码，考虑BERT模型的有效位置编码序列长度以及实际训练的模型规模，我们设置最大序列长度为max_length。对于给定的文档，将标题作为独立的句子；对于文档内容，若文本序列长度大于max_length，则依据中文标点符号将其切分成多个句子，反之则将整篇文档作为一个句子。

经过以上处理，本文将一篇文档表示为一系列句子集合$doc = \{s_0, s_1, \ldots, s_{N_s}\}$，$N_s$为句子总数，$s_j$为文档中第$j$个句子，$s_0$为文档标题。每个句子由一系列$token$组成$\{tok_{1,j}, \ldots, tok_{N_w,j}\}$，其中$tok_{i,j}$为第$j$个句子中第$i$个$token$，$N_w$为第$j$个句子的序列长度。每个句子经BERT编码后得到的$token$向量序列为$H_{tok} = \{h_{1,j}, h_{2,j}, \ldots, h_{N_w,j}\}$，其中$h_{i,j}$为第$j$个句子中第$i$个$token$对应的向量，维度为$d$，句子向量序列为$\{h_0, h_1, \ldots, h_{N_s}\}$，$h_0$为标题句向量表示，$h_j$为文档中第$j$个句子的向量表示，维度为$d$。

3.2 事件检测层

事件检测的目的是检测句子中包含的事件，本文的数据集样本中可能存在多个事件，受 (Liu al et., 2019)启发，我们将事件检测建模为多标签分类任务。事件检测样本的标注形式如表1所示。

其中s为句子，t_1、t_2为不同的事件类型，标签为1表示句子s中包含事件t_1，标签为0则表示句子s不包含对应的事件。

句子	事件类型	标签
s	t_1	1
s	t_2	0
s	t_3	1

表1.事件检测数据标注实例

对于给定文档的句子向量表示$\{h_0, h_1, ..., h_{N_s}\}$，我们依次将文档中的句子向量作为全连接层的输入，如式(1)所示。

$$H_{ed} = W_{ed} h_j + b_{ed} \tag{1}$$

其中W_{ed}为参数矩阵，b_{ed}为偏置，h_j为第j个句子s_j的隐层向量表示。对所有事件类型使用$Sigmoid$分类器进行分类，式(2)给出了对某类事件预测的计算方法。

$$y = \frac{1}{1 + e^{-H_{ed}}} \tag{2}$$

此层的预测错误使用交叉熵作为损失函数，如式(3)所示。

$$L_{ed} = -\frac{1}{N} \sum_{j=1}^{N} \sum_{m=1}^{M} y_{j,m} \log y^*_{j,m} + (1 - y_{j,m}) \log(1 - y^*_{j,m}) \tag{3}$$

其中$y^*_{j,m}$为第j个样本预测为第m类事件的预测值，$y_{j,m}$为第j个样本为第m类事件的真实值，N为样本总数，M为预定义事件类别总数，这里设定一个阈值为α，若$y^*_{j,m}$的值大于等于α，则认为该样本包含$y^*_{j,m}$对应的事件，否则认为该样本不包含$y^*_{j,m}$对应的事件。

3.3 事件元素识别层

本文将事件元素识别建模为序列标注任务，使用BIO标签模式（Begin：字段开头，Inside：字段内部，Outside：其他字段）为每个$token$赋予一个实体标签。

对于给定文档中的句子s_j，首先通过事件检测层预测得到对应的文档标题事件和文档内容事件，然后依次识别文档内容中每个事件的相关元素。为提高不同事件类型下实体的语义表示，本文在实体识别部分引入事件特征。具体做法如下：为每种事件类型定义d维（与$token$向量维度相同）的向量，通过查表的方式得到事件类型对应的向量t_{vec}，将句子中的每个$token$向量与事件类型向量t_{vec}进行拼接作为最终的特征向量，式(4)所示为计算某$token$的特征向量。

$$h_{vec,i} = [h_{i,j}; t_{vec}] \tag{4}$$

其中，$h_{i,j}$为第j个句子中第i个$token$的向量表示，$h_{vec,i}$为句子中第i个$token$最终的特征向量，“;”表示拼接，将特征向量$H_{vec} = \{h_{vec,1}, h_{vec,2}, \ldots, h_{vec,N_w}\}$作为全连接层的输入，使用$softmax$作为分类器预测每个token对应的标签，如式(5)所示。

$$P = softmax(W_{ner} H_{vec} + b_{ner}) \tag{5}$$

W_{ner}为参数矩阵，b_{ner}为偏置。使用交叉熵计算该部分的损失，如式(6)所示。

$$L_{ner} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} y_{i,k} \log P_{i,k} \tag{6}$$

N为样本总数，K为标签类别总数，第i个样本预测为第K个标签的概率为$P_{i,k}$，第i个样本真实的标签为$y_{i,k}$。

3.4 元素角色分类层

元素角色识别的目标是为确定事件类型下的实体赋予预定义的角色，本文将角色识别建模为多分类任务。为更好地区分实体扮演的角色，将利用注意力机制来增强文本的特征表示，依次判断每个实体在事件中扮演的角色。

实体往往包含多个$token$，对于给定句子s_j中识别出的实体集$E = \{e_1, e_2, \dots\}$，其中每个实体包含该句中的第i至第k个$token$,$[tok_{i,j}, \dots, tok_{k,j}]$，本文取实体中的所有字符向量的均值作为该实体的嵌入表示c，维度为d，采用这种均值向量可以有效避免模型过拟合问题 (Ji al et., 2018)。计算方式如式(7)所示，$h_{i,j}$为第j个句子中的第i个$token$。

$$c = \frac{1}{k}\sum_{i=1}^{k} h_{i,j} \tag{7}$$

然后，我们将E_{vec}中的第j个实体的向量表示c_j与包含该实体的事件类型对应的向量表示t_{vec}进行相加再求平均得到维度为d的,注意力机制查询向量q，如式(8)所示。

$$q = \frac{t_{vec} + c_j}{2} \tag{8}$$

最终得到查询向量集合$Q = q_1, q_2, \dots$，使用q与句子s_j的$token$向量表示H_{tok}计算得到每个$token$的注意力值a_k，计算方法如式(9)所示。

$$a_k = \frac{\exp(h_{k,j} q_{l,j}^T)}{\sum_{i=1}^{N_s} \exp(h_{i,j} q_{l,j}^T)} \tag{9}$$

其中，$q_{l,j}^T$为句子s_j第l查询向量，$h_{k,j}$表示句子s_j中第k个$token$的对应的，a_k为第k个$token$的注意力值，$a = [a_1, a_2, \dots, a_{N_s}]$为所有$token$的注意力值向量，由$a$与样本中每个$token$的向量$H_{tok}$相乘得到融合实体信息和事件类型语义信息的特征向量$V$，如公式(10)所示，$*$表示点乘。

$$V = a * H_{tok} \tag{10}$$

将V作为全连接层的输入，使用$softmax$分类器进行分类，如公式(11)所示。

$$y^* = softmax(W_{rt} V + b_{rt}) \tag{11}$$

其中M为类别总数，$y_{j,u}^m$为第j个样本中第u个事件元素属于第j类预定义角色类型的真实值，$y_{j,u}^{*(m)}$为模型预测值, N为样本总数。此部分的损失函数如(12)所示:

$$L_{rt} = -\frac{1}{N}\sum_{j=1}^{N}\sum_{u=1}\sum_{m=1}^{M} y_{j,u}^m \log y_{j,u}^{*(m)} \tag{12}$$

3.5 模型训练

我们将事件检测、事件元素识别以及元素角色分类同时进行训练，模型的训练目标是综合3部分的损失达到最小，训练时分别计算事件检测层的二分类交叉熵损失L_{ed}，以及事件元素识别层与元素角色分类层的多分类交叉熵损失L_{ner}与L_{rt}，我们将3个损失求和作为模型最终的优化目标：$L_{final} = L_{ed} + L_{ner} + L_{rt}$

模型训练时采用Adam (Kingma al et., 2015)作为优化器，通过验证集选择最好的模型进行预测。

3.6 主从事件识别

文档中包含多个事件时，将根据标题事件进行主从事件划分，然后对同指事件元素进行融合，从而得到篇章级事件抽取结果。文档标题往往能概括一篇文档的主要内容，故本文将文档标题中的事件作为主事件，其他事件作为从事件。

对于文档中的多个事件$Events = \{e_0, e_1, e_2, \dots\}$，其中$e_0$为文档标题预测出的事件，其他的均为文档内容预测出的事件。事件类型以及事件元素使用从DLEMC模型中获得的嵌入式表示，基于余弦相似度计算两个事件的相似程度，如式(13)所示。

$$obj = sim(e_0, e_i) \tag{13}$$

obj为相似度得分，用于衡量e_0与e_i两个事件的共指程度，我们取最高得分对应的那组事件作为文档的主事件，其他事件为从事件。

主从事件融合的目的是对同一个文档里多个事件之间的共指事件元素进行合并，从而得到规范的篇章级事件信息。本文通过计算不同事件中事件元素的语义相似度来衡量它们的共指程度，具体规则为：语义相似度超过设定阈值γ的事件元素作为共指元素，否则为非共指元素。两个事件中不同元素的相似度基于余弦相似度计算，如公式(14)所示。

$$score_{list} = sim(t_{1,i}, t_{2,j}) \qquad i, j \in \{1, 2, ...\} \quad t_{1,i}, t_{2,j} \in (T, O, F, D, N) \tag{14}$$

式(14)中$t_{1,i}$与$t_{2,j}$分别表示事件e_1的第i个元素与事件e_2的第j个元素，$score_{list}$为计算得到的语义相似度的得分集合，若该集合中某一项值大于阈值γ，就将该项对应的两个事件元素合并。

4 实验

4.1 数据集

本文将从互联网上搜集的上市公司公告作为实验数据集。共有文档总数23067，其中5056个文档中包含多个事件，占比21.9%，将总文档按照8：1：1划分成训练集、验证集和测试集，数据集中标注的实体类别包括NUM（Number，数值）、ORG（Organization，组织机构）、FIN（Finance，金融产品）、TIM（Time，日期、时间）。事件分为11类：上市交易、停牌、临时停牌、复牌、摘牌、名称变更、支付利息、债券转让、暂停上市、终止上市、到期兑付。各类样本数如表2所示。

事件类型	训练集	验证集	测试集	总数	多事件率(%)
上市交易	5575	902	897	7554	0
临时停牌	3061	266	284	3611	94.0
停牌	1671	189	236	2096	6.3
复牌	3122	260	321	3073	95.2
摘牌	1252	143	135	1530	100
债券转让	1082	154	166	1042	0
终止上市	589	128	102	819	0
到期兑付	1252	1143	135	1530	100
暂停上市	934	136	90	1160	0
名称变更	530	87	102	719	0
支付利息	822	91	167	1080	0
文档数量	18455	2306	2306	23067	21.9

表2.数据集样本统计

为了验证远程监督标注事件的质量，我们从每类事件中随机选取20个样本进行人工标注，作为真实值，再用远程监督方法标注它们作为预测值，依据4.2中的评价指标进行验证，验证结果如表3所示。表3说明远程监督方法标注的语料具有较高的精确率，以及不错的召回率和F1值。本文的实验中均采用远程监督方法生成训练集、验证集和测试集。

Precision	Recall	F1	多事件率(%)
97.5	90.0	93.3	26.3

表3.远程监督事件标注质量

4.2 模型评价

本文采用精确率(Precision，P)、召回率(Recall，R)和F1(F1-measure，F1)值作为评价指标，采用微平均计算F1值，一个事件类型与某一事件元素及其角色为一个统计项。在事件类型预测正确的前提下，若事件元素及其对应的角色均与标注相同则视为正确，否则视为预测错误，若事件类型预测错误则将所有的元素与角色均视为预测错误，具体计算方式如下。

$$P = \frac{\text{识别事件类型与元素和标注相同的数量}}{\text{识别出事件类型与元素总数量}}$$

$$R = \frac{\text{识别出事件类型与元素和标注相同的数量}}{\text{标注的事件类型与元素总数量}}$$

$$F_1 = \frac{(2*P*R)}{(P+R)}$$

4.3 参数设置

本文的实验基于BERT_base模型来初始化词向量，维度为786，dropout的比率为0.4，批次大小为16，模型学习率为3e-5，训练10个epoch，最大文本序列长度max_length为200，事件类型向量随机初始化生成，维度为768。

4.4 结果分析

我们实现了Yang (2018)提出的DCFEE模型，Zheng (2019)提出的Doc2EDAG模型，以及基于BERT的管道模型BERT-P，在BERT-P中事件检测部分与本文提出的方法DLEMC相同，但在实体识别任务中未增加事件特征，在角色分类任务中未利用事件和实体特征注意力，DLEMC-P为DLEMC的管道模式。我们将DCFEE、BERT-P和Doc2EDAG作为本文的baseline，在包含全部11类事件的测试集上进行各项测试。

1）为验证本文的模型DLEMC在事件类型检测上的有效性，我们在测试集上对模型进行了评价，实验结果如表4所示。

Model	P(%)	R(%)	F1(%)
DCFEE	80.3	77.2	78.7
BERT-P/DLEMC-P	84.1	83.2	83.6
Doc2EDAG	**86.1**	84.3	85.1
DLEMC	85.8	**84.7**	**85.3**

表4.事件检测评价结果

如表4所示，利用序列标注识别事件触发词的DCFEE模型效果比较差。通过分类模型检测事件的BERT-P、DLEMC-P、DLEMC、Doc2EDAG模型在各项指标上均优于DCFEE，其中端到端联合学习模型DLEMC、Doc2EDAG在各项指标上均优于管道式模型。本文提出的DLEMC模型在准确率上略低于采用二分类策略的Doc2EDAG模型，但召回率和F1均优于Doc2EDAG，其中F1提高了0.2%，实验表明多标签分类模型在多事件检测中有较好的表现。

2）为验证DLEMC模型在篇章事件元素识别和角色分类时的有效性，我们在测试集上对模型进行了评价。

由表5实验结果可以看出，基于预训练语言模型BERT词向量表征的事件抽取模型BERT-P明显优于使用word2vec (Mikolov et al., 2013)的DCFEE模型。本文提出的DLEMC模型则在P、R、F1等3个指标上都优于BERT-P，其中F1提升了4.1%。

DLEMC-P保留增强的文本嵌入表示，但采用管道式方法完成事件抽取的子任务，实验表明其在准确率上有提升，但是召回率上大幅下降，但F1值仍然比直接基于管道的BERT-P提升

Model	P(%)	R(%)	F1(%)
DCFEE	64.3	58.2	61.1
BERT-P	74.5	58.7	65.7
DLEMC-P	**77.1**	59.6	67.2
Doc2EDAG	76.1	64.0	69.5
DLEMC	76.2	**64.3**	**69.8**

表5.篇章级别事件抽取评价结果

了1.5%。端到端模型Doc2EDAG和DLEMC在准确率上略低于DLEMC-P，但召回率和F1较管道式方法均有大幅提升。得益于DLEMC在实体识别部分加入事件类型特征，在角色分类部分加入事件类型与实体注意力特征，本文的DLEMC在各项指标上均优于Doc2EDAG。

3）为验证DLEMC在处理多事件时的有效性，我们将数据集划分为单事件(Single-event)与多事件(Multi-event)两个子集，并分别用这两个子集对模型进行评价，实验结果如表6所示。

Model	Single-event(%)	Multi-event(%)	Avg(%)
DCFEE	63.7	41.5	52.6
BERT-P	76.5	60.2	68.3
DLEMC-P	79.6	62.7	71.2
Doc2EDAG	81.2	63.5	72.3
DLEMC	**81.4**	**63.8**	**72.6**

表6.单一事件与多事件评测F1值与平均值(Avg)

表6中的实验结果表明DCFEE中的基于核心事件进行元素补全策略存在局限性，不能较好的处理多事件样本。而BERT-P以及DLEMC-P基于优秀的词嵌入方法，提高了特征表达能力，在单事件和多事件样本上性能均有较大改善，但由于管道模式不可避免地将前序任务中的错误信息传递至后序任务，模型的整体性能低于端到端的模型Doc2EDAG和DLEMC。实验表明，对于被分割的样本，DLEMC有效增强了文本中的事件特征表示，提高了在同一类型事件的多实例和不同类型事件的多实例情况下的性能，其在单事件和多事件评价上均优于Doc2EDAG。

4）为得到完整的结构化篇章事件信息，我们在DLEMC事件抽取结果的基础上对包含多个事件的文档进行了主从事件划分和主从事件元素融合，实验结果如表7、表8所示。

P(%)	R(%)	F1(%)
83.4	82.1	82.7

表7.主从事件划分评价结果

由表7可看出，主从事件划分的效果是可接受的，由于主从事件划分依赖事件检测的结果，使得该部分仍具有较大提升空间。

P(%)	R(%)	F1(%)
70.4	67.1	68.7

表8.主从事件元素融合评价结果

表8给出了对正确识别且角色判定正确的事件元素进行同指事件元素融合实验的评价结果。事件元素融合的性能受到句子级事件抽取结果的影响，造成最终的性能指标偏低。

5 相关工作

目前事件抽取方法可以分为两类：基于模式匹配方法和基于统计学习方法。模式匹配方法在特定领域有较高的准确率，但是通常需要编写大量的人工模板，且普适性较差 (Yangarber and Grishman, 1997; Surdeanu and Harabagiu, 2002)。统计学习方法可以分为两类:传统的基于特征工程的机器学习方法以及基于深度学习的方法。传统特征工程主要依赖自然语言处理工具获取有效的特征（如句法、词汇、词性等），然后利用传统的分类模型（如最大熵、支持向量机）进行分类 (Ahn, 2006; Jungermann and Morik, 2008; Liao and Grishman, 2010)。基于深度学习的方法依靠神经网络自动提取特征，在事件抽取中取得了不错的效果。如WU (2019)使用一种混合神经网络模型，进行实体和事件的联合学习。Chen (2015)使用一种动态多池化卷积神经网络来捕获多个特征，提升了事件抽取的性能。Zeng (2016)使用一种卷积双向LSTM神经网络，分别从词级别和字级别进行触发词和实体的识别。

事件抽取任务依据是否具有预定义的事件框架（事件类型及每类事件对应的角色）可以分为开放域和限定域事件抽取 (Wei and Wang, 2019)，开放域事件抽取目标在于识别自然语言文本中的事件，一般没有领域限制，不需要预定义事件框架。限定域事件抽取会预先定义好要抽取的事件类型，如“袭击”事件、“审判”事件等。同时也会定义每类事件参与者的角色，如“审判”事件中包含“被审判人”、“审判时间”、“地点”等角色。

从文本粒度来看，目前事件抽取的相关研究主要集中在句子级别，即识别句子中的事件并提取相应的事件元素。句子级事件抽取主要有两种建模方式，管道模式和联合模式。管道式方法通常将事件识别和元素提取分为两个独立的任务，忽略了事件与元素之间的联系，导致效果不够理想。联合模型一般同时识别句子中的事件并提取相关元素，利用深度神经网络捕获事件与元素之间的语义联系，模型训练时能够互相影响并优化，性能一般要优于管道式模型。

现实中的文本信息往往是以篇章形式出现的，针对的篇章事件抽取能够获得更完整、规范的信息。篇章级别的事件抽取研究方法通常首先对给定文档中的句子进行处理，然后再对句子级别的事件信息进行合并，从而得到篇章级别的事件抽取结果。目前篇章级的事件抽取大多采用管道式模型，如ZHONG (2019)采用触发词和实体联合标注的方法同时抽取句子级别的触发词和实体，然后使用多层感知机对实体进行角色分类，并利用整数线性规划进行同指事件的融合，实现篇章级别的事件抽取。Yang (2016)通过采用联合因子图模型来联合学习每个事件内部的结构化信息、篇章内不同事件间的关系和实体信息，提高了篇章事件抽取的效果，但他们的工作依赖大量的人工特征。Yang (2018)基于句子抽取结果以及文本特征发现主事件描述，并利用上下文元素补齐策略得到篇章事件结构化信息。

总体来讲，目前篇章级别的事件抽取研究较少，且集中在特定领域，通常依赖大量人工规则，难以进行领域拓展。而句子级别的事件抽取日趋成熟，应用领域更广，但得到的结果无法提供较好的篇章级事件信息。

6 总结与展望

本文针对金融领域篇章级别事件抽取任务定义了事件表示框架，在该框架下提出基于深度学习的端到端模型抽取事件信息，模型采用3层，分别实现多标签分类的事件检测、基于融合事件类型特征的事件元素识别以及基于注意力机制的元素角色分类。对获取的多个事件，利用余弦相似度进行主从事件划分以及多事件的元素融合，得到篇章级事件结构化信息。我们构建了金融领域事件抽取语料对本文方法进行验证，实验证明本文方法明显优于基准方法。

然而，由于事件元素之间存在较强的相似性，如“上市交易”事件中的“股份总数160,000,000股，其中40,000,000股自上市之日起开始上市交易”，模型可能会将“股份总数”与“上市股数”错误分类。如何提高相似元素的特征表示及其分类效果，从而提升篇章级事件抽取的整体性能，是本文未来的改进方向。

参考文献

David Ahn. 2006. *The stages of event extraction*, In The Workshop on Annotating and Reasoning about Time and Events. pages 1–8.

Xiang,Wei,and Bang.Wang. 2019. *A Survey of Event Extraction From Text*, IEEE Access 7:173111-173137.

J.Piskorski,H.Tanev,M.Atkinson,E.V.D.Goot,andV.Zavarella. 2011. *Online news event extraction for global crisis surveillance*, Transactions on computational collective intelligence, vol. 6910, no. 1, pp. 182–212.

S.Ribeiro,O.Ferret,andX.Tannier. 2017. *Unsupervised event clustering and aggregation from news wire and web articles*, in Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism, pp. 62–67.

S. Yu and B. Wu. 2018. *Exploiting structured news information to improve event detection via dual-level clustering,* in IEEE Third International Conference on Data Science in Cyberspace, pp. 873–880.

Chen, Y., Xu, L., Liu, K., Zeng, D., Zhao, J. 2015. *Event extraction via dynamic multipooling convolutional neural networks.* In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, vol. 1, pp. 167–176.

Zeng, Y., Yang, H., Feng, Y., Wang, Z., Zhao, D. 2016. *A convolution BiLSTM neural network model for Chinese event extraction.* In Natural Language Understanding and Intelligent Applications (pp. 275-287). Springer, Cham.

Qi Li, Heng Ji, and Liang Huang. 2013. *Joint event extraction via structured prediction with global features.* In ACL.

Nguyen, Thien Huu and Cho, Kyunghyun and Grishman, Ralph. 2016. *Joint Event Extraction via Recurrent Neural Networks.* Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.

Hang Yang,Yubo Chen,Kang Liu,Yang Xiao,and Jun Zhao. 2018. *DCFEE: A Document-level Chinese Financial Event Extraction System based on Automatically Labeled Training Data*, In Proceedings of ACL 2018, System Demonstrations.

Zheng Shun,Cao Wei,Xu Wei,Bian Jiang. 2019. *Doc2EDAG: An End-to-End Document-level Framework for Chinese Financial Event Extraction*, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP).

ZHONG Weifeng, YANG Hang, CHEN Yubo, LIU Kang, ZHAO Jun. 2019. *Document-level Event Extraction Based on Joint Labeling and Global Reasoning.*,33(9): 88-95,106.

Devlin,J., Chang,M., Lee,K.,Toutanova,K. 2019. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* ArXiv, abs/1810.04805.

Liu, Shulin & Li, Yang & Zhang, Feng & Yang, Tao & Zhou, Xinpeng. 2019. *Event Detection without Triggers.*735-744. 10.18653/v1/N19-1080.

Ji Xin, Yankai Lin, Zhiyuan Liu, Maosong Sun. 2018. *Improving Neural Fine-Grained Entity Typing with Knowledge Attention.* The 32th AAAI Conference on Artificial Intelligence.

Kingma, Diederik P. and Ba, Jimmy. 2015. *Adam: A Method for Stochastic Optimization.* In ICLR.

Mikolov T,Sutskever I,Chen K,et al. 2013. *Distributed representations of words and phrases and their compositionality.* Advances in Neural Information Processing Systems, 26, 3111-3119.

Yangarber R, Grishman R. 1997. *Customization of information extraction systems.* [C]//Proceedings of International Workshop on Lexically Driven Information Extraction：1-11.

Surdeanu M,Harabagiu S M. 2002. *Infrastructure for open-domain information extraction.* [C]// Proceedings of the 2nd International Conference on Human Language Technology Research. Morgan Kaufmann Publishers Inc：325-330.

Felix Jungermann, Katharina Morik. 2008. *Enhanced Services for Targeted Information Retrieval by Event Extraction and Data Mining.* [M]// Natural Language and Information Systems. Springer Berlin Heidelberg.

Liao S, Grishman R. 2010. *Using Document Level Cross-Event Inference to Improve Event Extraction.* [C]// ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden. DBLP.

WU Wentao, LI Peifeng, ZHU Qiaoming. 2019. *Joint Extraction of Entities and Events by a Hybrid Neural Network*. 33(8): 77-83.

Yang Bishan and Mitchell Tom M. 2016. *Joint extraction of events and entities within a document context*. // Proceedings of NAACL- HLT 2016. San Diego，California，USA: Association for Computational Linguistics: 289-299

面向微博文本的融合字词信息的轻量级命名实体识别

陈淳 **李明扬** **孔芳***
苏州大学计算机科学与技术学院
江苏省苏州市干将东路333号苏州大学158信箱
{20195227037, 20175227067}@stu.suda.edu.cn, kongfang@suda.edu.cn

摘要

中文社交媒体命名实体识别由于其领域特殊性，一直广受关注。非正式且无结构的微博文本存在以下两个问题：一是词语边界模糊；二是语料规模有限。针对问题一，本文将同维度的字词进行融合，获得丰富的文本序列表征；针对问题二，提出了基于Star-Transformer框架的命名实体识别模型，借助星型拓扑结构更好地捕获动态特征；同时利用高速网络优化Star-Transformer中的信息桥接，提升模型的鲁棒性。本文提出的轻量级命名实体识别模型取得了目前Weibo语料上最好的效果。

关键词： 命名实体识别；中文社交媒体；星型-Transformer；高速网络

Lightweight Named Entity Recognition for Weibo Based on Word and Character

Chun Chen **Mingyang Li** **Fang Kong**
School of Computer Science and Technology, Soochow University
{20195227037, 20175227067}@stu.suda.edu.cn, kongfang@suda.edu.cn

Abstract

Chinese social media named entity recognition has been widely concerned due to its domain specificity. Informal and unstructured Weibo text has two issues to be addressed. First is the ambiguous word boundary; Second is the limited scale of corpus. To deal with the first problem, this paper places character and word embedding on the same dimension to obtain rich sequence representation. Aiming at the second problem, a named entity recognition model based on Star-Transformer framework is proposed to capture dynamic feature preferably, with the help of star topology structure. Besides, Highway Networks is also used to optimize the information connection in Star-Transformer, improving the robustness of the model. The lightweight named entity recognition model proposed in this paper achieves the best performance on Weibo corpus.

Keywords: Named Entity Recognition , Chinese Social Meida , Star-Transformer , Highway Networks

1 引言

*通讯作者：kongfang@suda.edu.cn

命名实体识别(Named Entity Recognition, NER)旨在识别出非结构化文本序列中具有特殊含义的实体，并为这些实体分配相应的类别，比如人名、地名、组织机构名等等。由于命名实体识别在对话生成 (Reddy et al., 2019)、关系抽取 (Zelenko et al., 2003)、共指消解 (Clark and Manning, 2016)等任务中起着基础支撑作用，因此命名实体识别在自然语言处理(Natural Language Processing, NLP)领域得到了广泛的研究。

社交媒体领域的中文命名实体识别一直是亟待发展的热点任务之一，由于其领域特殊性，社交媒体的中文命名实体识别主要有3个难点：1) 相较于英文，汉语没有显式的词语边界，专有词汇也没有拼写变化等提示信息；2) 社交媒体领域多是不规范的短文本，新词和错词频繁出现，网络用语以及表情等噪声干扰较多；3)社交媒体领域的语料相较于规范的新闻类语料规模较小。

作为一个典型的序列标注问题，命名实体识别的神经网络模型通常包含三个组件：单词嵌入层、上下文编码器层以及解码器层，现有的命名实体识别模型一般通过不断优化这三个组件来寻求突破。循环神经网络（Recurrent Neural Network, RNN） (Zaremba et al., 2014)这一类编码器，尤其是双向长短期记忆网络(Bi-directional Long Short-Term Memory, BiLSTM) (Hochreiter and Schmidhuber, 1997)以序列作为输入，在上下文信息方面有着强大的学习能力，但是不能捕捉到较长距离的上下文依赖关系。Vaswani et al. (2017)提出的Transformer模型采用完全连接的自注意力结构来对远程上下文进行建模，能够较好地弥补RNN模型的缺点，而且Transformer具有更好的并行计算能力。但Yan et al. (2019)的实验证实Transformer不能很好地适用于NER任务中，原因是Transformer内部结构复杂并且采用全连接的注意力机制，这也导致性能的提升需要依赖大量的训练数据。而对于社交媒体领域的命名实体识别而言，难点之一就是语料规模过小，因此传统的Transformer无法取得预期的性能。因此，如何将Transformer较好地融入到NER任务中成为“当务之急”。

已有的中文社交媒体命名实体识别研究都是使用字粒度的Weibo语料 (Peng and Dredze, 2015)，本文同样的在Weibo语料上进行实验。此外，考虑到词粒度的语料包含更丰富且情境化的序列信息，我们根据Weibo语料中每个字的位置特征对语料进行了重新标注，整理出按词粒度划分的新Weibo语料。进一步的，本文将字粒度和词粒度置于同等维度作为编码层的输入，获得了较好的文本序列表示，有效缓解了中文边界模糊的问题。

本文的主要贡献如下：1)本文重新标注并整理出词粒度的中文Weibo语料，已开源供大家使用[0]；2)本文首次将轻量级Star-Transformer模型应用到中文社交媒体NER任务中，并且利用Highway Networks机制使Star-Transformer更高效地适配NER任务，取得了可观的性能。实验结果表明，本文提出的基于字词粒度的Star-Transformer with Highway Networks(STHN)模型可以大幅提升Transformer在社交媒体命名实体识别上的性能，并且在Weibo语料上取得目前最好的性能。

2 相关研究

RNN一类的神经网络模型由于其顺序特征而被应用在NLP任务中，而其中应用最为广泛的BiLSTM模型已经成为主流编码器。Huang et al. (2015)等首先引入BiLSTM和CRF模型来解决序列标注问题，从那时起，BiLSTM模型被广泛应用于NER领域，Chiu and Nichols (2016)、Dong et al. (2016)、Lample et al. (2016)以及Ma and Hovy (2016)的研究都是基于此模型。

在字词编码方面，已有研究均是以字粒度为主，词粒度为辅，没有将二者放在同等维度上考虑。具有代表性的相关研究有：Zhang and Yang (2018)引入Lattice结构将所有与词典匹配的潜在单词信息整合到字符序列中，获得较好的向量表示；Gong et al. (2020)以字粒度作为输入，将词语边界信息融入到BiLSTM和CRF中，以此弥补字粒度输入信息不足的缺点；Gui et al. (2019)利用CNN对不同窗口大小的潜在单词进行编码；Peng et al. (2019)优化了Lattice结构的潜在词向量表示，获得了较快的运算速度以及更好的命名实体识别性能。

尽管BiLSTM模型在NER领域获得了不小的成就，但是它必须逐一计算token的表示，这极大地阻碍了GPU的并行性利用，而且BiLSTM无法捕捉到较长距离的上下文依赖关系。2017年以来，Transformer (Vaswani et al., 2017)逐渐在NLP各个任务中占据主导地位，例如机器翻译 (Vaswani et al., 2017)、语言建模 (Radford et al., 2018)以及预训练模型 (Devlin et al.,

[0]词粒度划分的weiboNER语料：https://github.com/cchen-nlp/weiboNER

2019)等等。然而Transformer在NER任务中效果不佳，Yan et al. (2019)提出了TENER模型，引入了方向感知、距离感知和无比例关注，同时定制了命名实体识别专属的Transformer编码层，使得Transformer在NER任务上获得较好性能。Li et al. (2020)的FLAT模型将Lattice结构转换为平面结构，并且设计合适的Transformer位置编码，进一步提升了NER性能。

本文结合字与词粒度各自的优势，将二者放在同等维度作为下层编码器的输入，同时引入Star-Transformer代替传统模型中的BiLSTM模型，加入Highway Networks机制进行结点信息的自我桥接，通过Star-Transformer特有的注意力连接和门控机制的动态调整服务于社交媒体领域的命名实体识别。

3 基于Star-Transformer 的命名实体识别框架

将命名实体识别看作是序列标注问题之后，实体采用BMES规则标注，实体的开头标注为B(Beginning)，实体内部单元标注为M(Median)，实体的结尾标注为E(End)，其他的词标注为O(Other)。

图 1给出了本文提出的基于字词粒度的Star-Transformer with Highway Networks模型完整框架，从图中可以看出STHN模型可以分为两个部分：第一部分是字词粒度的嵌入式表示，模型以同等维度的字向量和词向量作为输入，其中字向量需要经过Self-Attention做初步处理；第二部分是STHN模型。下面逐个介绍STHN模型中各个组成部分。

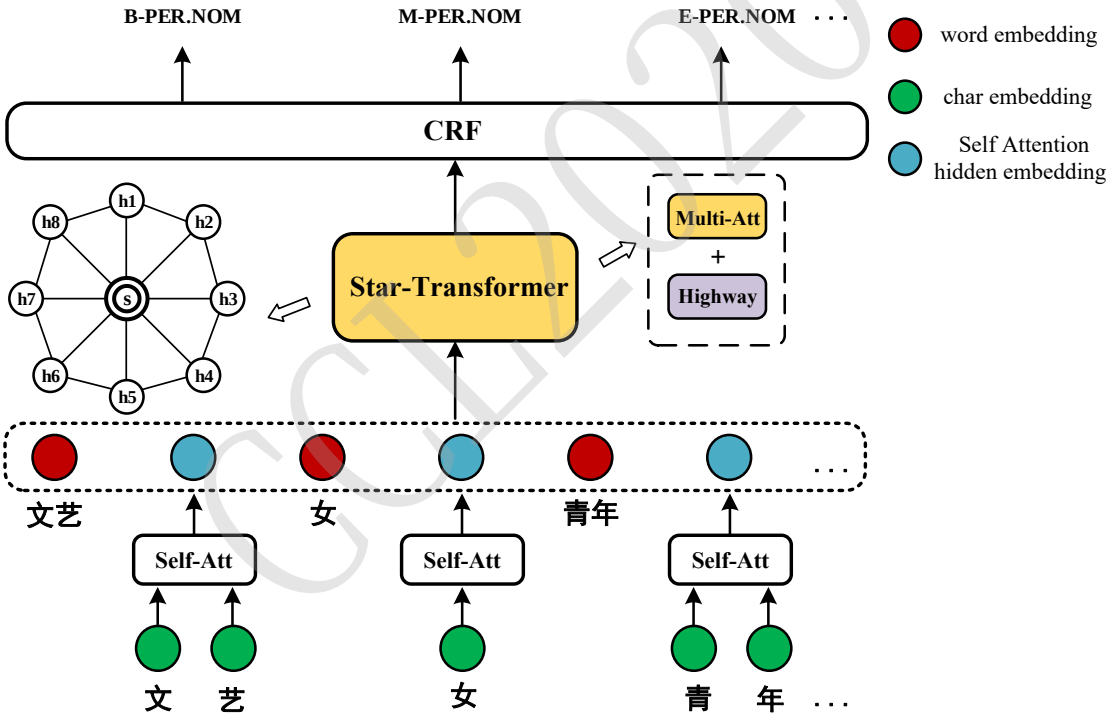


图 1. STHN模型图

3.1 字、词粒度的模型输入

在编码阶段，原始数据通过查找字或词向量表转化为字或词向量序列。对于文本中的字与词的向量表示，我们使用2018年预训练的词向量 (Li et al., 2018)，该词向量使用Word2vec 中的Skip-Gram模型训练，维度为300。该词向量包括百度百科、中文维基百科、人民日报、微博、知乎等多领域的字词特征，具体的模型训练设置如表 1所示。

字词向量表查找的过程是让原始文本中每一个字符或者单词在表上查找相对应的字词向量，如果某个字符或单词在表中不存在，则被赋予一个随机值。

考虑到现有研究都是以字粒度作为编码层的输入，并且He and Wang (2008)、Liu et al. (2010)和Li et al. (2014)的工作验证了基于字粒度要优于词粒度，但是基于字粒度的嵌入式表示

Window Size	Dynamic Window	Sub-sampling
5	Yes	1e-5
Low-Frequency Word	**Iteration**	**Negative Sampling**
10	5	5

表 1. SGNS模型训练参数设置

存在识别结果的标签不连续的情况，而基于词粒度的嵌入式表示具有显式的词汇边界，可以有效缓解社交媒体语料中中的边界模糊问题。本文将字粒度与词粒度放在同等维度上作为输入，其中字粒度需要先经过Self-Attention做初步特征提取，这部分与Yan et al. (2019)是相同的。

本文模型中最终编码层的输入是词粒度的嵌入式表示与经过特征提取的字粒度嵌入式表示的结合，如公式(1)∼(2)所示:

$$char^{'} = SelfAtt(char) \tag{1}$$

$$h_i = [word_i; char_i^{'}] \tag{2}$$

3.2 Star-Transformer模型

传统Transformer的注意力连接为全连接结构，如图 2(a)所示，而命名实体识别任务旨在识别出特定含义的实体，并且社交媒体语料中实体密度较稀疏，并不需要时刻关注句子序列中所有的结点，即传统Transformer的全连接结构在命名实体识别任务里存在信息冗余的现象，这些多余的信息不仅会降低运算速度，甚至会对命名实体识别任务起到反作用。因此传统Transformer对于社交媒体的实体识别任务来说并不合适。为了降低模型的复杂性，Guo et al. (2019)提出用星型拓扑结构代替全连通结构来简化架构，如图 2(b)所示。其中每两个相邻结点通过一个共享中继结点进行连接，因此，模型复杂性从二次降低到线性，同时保留捕获局部成分和长期依赖关系的能力。本节将详细介绍Star-Transformer的相关内容。

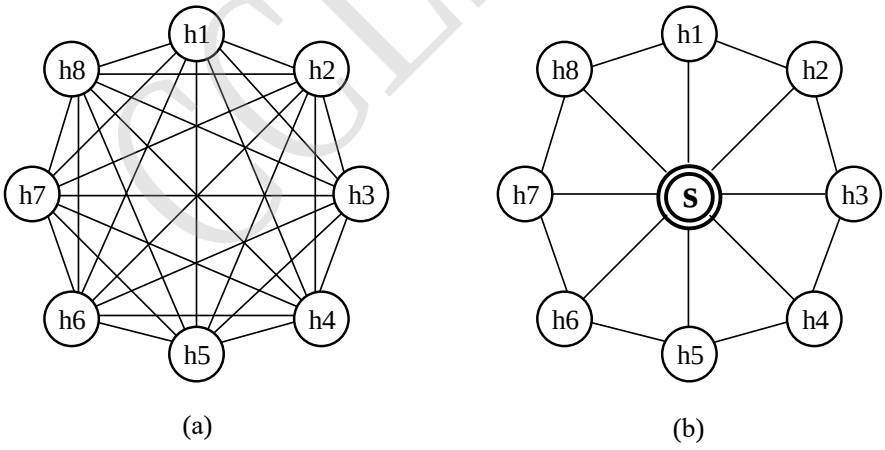


图 2. 传统Transformer(a)与Star-Transformer(b)结点连接方式图

3.2.1 Multi-Head Attention

Transformer (Vaswani et al., 2017)首先使用h个注意力头对一个输入序列分别进行单独的自我注意，然后对每个注意力头进行连接和线性变换操作，称为多头注意力机制（Multi–Head Attention）。一般来说，多头注意力机制可以用查询（query）到一系列键（key）值（value）对的映射来描述。

首先介绍放缩点积注意力Scaled Dot-product Attention，其本质上是使用了点积进行相似度计算。给定一个向量序列X，我们可以使用一个查询向量Q软选择相关信息，如公

式(3)∼(4)所示：

$$Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})) \cdot V \tag{3}$$

$$K = XW^K, V = XW^V \tag{4}$$

其中，W^K，W^V是对应向量的学习参数。然后我们可以将多头注意力机制定义成公式(5)∼(6)：

$$MultiAtt = (z_1 \oplus z_2 \oplus \cdots \oplus z_h) \cdot W^o \tag{5}$$

$$z_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{6}$$

其中，$\oplus$表示向量连接操作，W^o, W_i^Q, W_i^K, W_i^V是对应向量的学习参数。

3.2.2 Star-Transformer Encoder

Star-Transformer (Guo et al., 2019)的星型拓扑结构如图 2(b)所示，由一个中继结点s和n个卫星结点组成。第i个卫星结点h_i的状态表示文本序列中第i个token的特征。中继结点s充当虚拟中心，在所有卫星节点之间收集和散布信息。

Star-Transformer提出了基于time step的循环更新方式：每个token由输入向量初始化，中继结点初始为所有token的平均值，每个token依次通过多头注意力机制更新。在更新卫星结点的时候，每个卫星结点h_i的状态根据其相邻的结点更新，包括上一轮的上一个结点的隐态h_{i-1}^{t-1}；上一轮该结点的隐态h_i^{t-1}；上一轮下一个结点的隐态h_{i+1}^{t-1}；本结点的向量表示e^i；上一轮的中继结点状态s^{t-1}，具体过程如公式(7)∼(9)所示：

$$C_i^t = [h_{i-1}^{t-1}; h_i^{t-1}; h_{i+1}^{t-1}; e^i; s^{t-1}] \tag{7}$$

$$h_i^t = MultiAtt(h_i^{t-1}, C_i^t, C_i^t) \tag{8}$$

其中，C表示第i个卫星结点的上下文信息，在更新完信息后，使用层归一化操作处理卫星节点信息：

$$h_i^t = LayerNorm(ReLU(h_i^t)) \tag{9}$$

在更新中继结点s时，中继结点s将汇总所有卫星结点h_i的信息以及之前的状态，如公式(10)∼(11)所示：

$$s^t = MultiAtt(s^{t-1}, [s^{t-1}; H^t], [s^{t-1}; H^t]) \tag{10}$$

$$s^t = LayerNorm(ReLU(s^t)) \tag{11}$$

通过交替更新卫星结点h_i和中继结点s的信息，Star-Transformer在减少了注意力连接的前提下，依旧可以捕获句子序列中的局部特征和长期依赖关系，能够较好地融入到命名实体识别任务中。

3.3 Highway Networks

高速网络（Highway Networks）(Srivastava et al., 2015)是一种能够在信息传递之间进行平滑切换的神经网络，它能够有效解决网络深度加深，梯度信息回流受阻，造成网络训练困难的问题。Dauphin et al. (2017)、Gehring et al. (2017)以及Wu et al. (2019)验证了LSTM类型的门控单元在序列学习任务中是有效的。而Chai et al. (2020)证明了高速网络类型的门控机制有助于增强Transformer组件。

考虑到Star-Transformer用星型拓扑结构代替全连通结构，减少了相对较多的计算，我们对Star-Transformer中的每个卫星结点h_i利用高速网络进行信息的自我桥接，使得每一层Star-Transformer都能够充分利用上一层的卫星节点信息，这样的自我桥接可以看作是特征的动态调整，图 3给出了我们在Star-Transformer内部加入的高速网络结构。

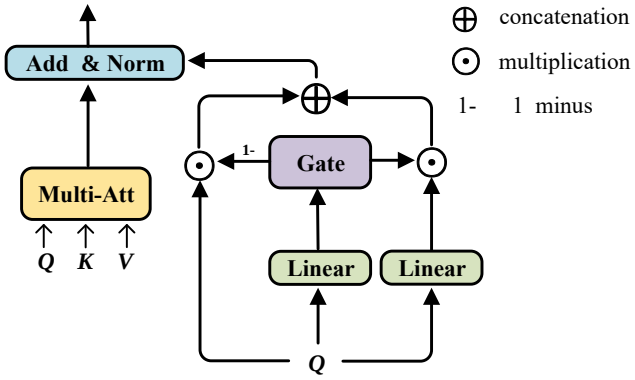


图 3. 高速网络结构图

我们在Star-Transformer计算完多头注意力之后，进入层归一化之前，加入一个新的输入分支$HW(h_i)$，这个输入分支就是上文所说的卫星结点h_i的自我桥接，如公式(12)∼(14)所示：

$$gated = \sigma(w_1 h_i + b_1) \tag{12}$$

$$f(h_i) = w_2 h_i + b_2 \tag{13}$$

$$HW(h_i) = [(1 - gated) \cdot h_i + gated \cdot f(h_i)] \tag{14}$$

其中，w_1, w_2表示门控机制的权重参数，b_1, b_2表示门控机制的偏差参数，σ为激活函数。

接着我们将使用高速网络增强的表征来丰富原有的多头注意力结果，如公式(15)所示：

$$H_i = LayerNorm(HW(h_i) + MultiAtt(h_i, C_i, C_i)) \tag{15}$$

最后，经过高速网络进行自我桥接之后的卫星节点信息和多头注意力计算结果相加并经过层归一化，得到新的卫星节点H_i。

4 实验设置和结果分析

本文使用命名实体识别社交媒体领域的Weibo数据集，通过不同的设置对前文所述的模型进行实验，并对实验结果进行讨论与分析，最终采用准确率$Precision$、召回率$Recall$和综合指标$Micro - F_1$值 (DBL, 1992)对标注结果进行评价。

4.1 实验数据集

本文采用了Peng and Dredze (2015)公开的Weibo NER语料，该语料是按照字粒度划分的，我们根据语料中的位置特征，整理出了对应的按照词粒度划分的Weibo NER语料，且将标注方式从BIO标注转换成了BMESO标注。

我们在整理语料过程中还引入了词性标注（part-of-speech tagging）特征，希望能够通过对语料中不同词性的区别来优化命名实体识别的结果。语料中采用Stanford Parser的词性标注器进行标注，使用的模型是chinese-distsim.tagger (de Marneffe et al., 2014)。我们对比了将一整个句子进行标注的方式以及对单个词标注的方式，最终采用更加准确的融合句法信息的标注方式。为了模型对比的公平性，本文的实验部分没有使用词性标注等外部信息。

更新后的Weibo NER语料的字粒度结构不变，包含训练集、开发集和测试集共1890句。表 2详细地给出了该语料原本的字粒度结构以及我们整理之后词粒度结构，从中我们可以清晰地看到Weibo语料规模较小，带标记字符的数量表明了待识别的实体数目也相对较少。

Weibo NER语料标注的实体类型包括PER、ORG、LOC和GPE，且每个类型分别有特定实体（named entity, NE）和指代实体（nominal mention, NM），表 3给出了Weibo NER语料中各个类别的分布情况。特定实体即为传统领域中需要识别出来的实体，比如人名的特定实体

Type	Train	Dev	Test
Sentence	1350	270	270
Character	73378	14509	14842
Word	45678	9026	9143
Char with label	4951	971	1078
Label percent	6.71%	6.69%	7.26%

表 2. Weibo NER数据集结构

有詹天佑、钱钟书以及舒淇等，而指代实体是将名词性的指代词作为实体，例如人名的指代实体有阿姨、妹纸以及皇上等。社交媒体类语料中常常会有特定实体和指代实体混合出现的情况，这是其与规范的新闻类语料差别最大的地方，这种特殊的结构无疑增加了社交媒体领域命名实体识别的难度。

Type	Train	Dev	Test	All
GPE.NAM	205	26	47	278
GPE.NOM	8	1	2	11
LOC.NAM	56	6	19	81
LOC.NOM	51	6	9	66
ORG.NAM	183	47	39	269
ORG.NOM	42	5	17	64
PER.NAM	574	90	111	775
PER.NOM	766	208	170	1144

表 3. Weibo NER数据集Label分布

4.2 实验参数设置

本文实验采用Pytorch 0.4.1框架，并用NVIDIA的1080GPU进行加速。使用的预训练词向量参数在表 1中已经给出，模型的查询表使用预训练得到的向量进行初始化，其他参数均采用均匀分布的随机函数初始化。

表 4给出了模型的参数值，我们使用Adam（Adaptive moment estimation）来优化所有可训练的参数；为了保证字词二者的同一性，使用的字词嵌入式表示维度都是300；神经网络的隐藏层维度均设为300；多头注意力机制的头数head为5(维度300可被head整除)；Star-Transformer层数为6层；整个模型的学习率learning rate设置为0.0005，学习率减少步长lr_decay设置为0.05，所有神经网络的dropout设置为0.5，L2正则化参数设置为1e-8。

Parameter	Value	Parameter	Value
char emb size	300	learning rate	0.0005
word emb size	300	lr_decay	0.05
hidden dim	300	dropout	0.5
Multi head	5	batch size	10
star layer	6	regularization	1e-8

表 4. 超参数设置

4.3 实验结果及分析

表 5给出了本文的模型在社交媒体Weibo语料上的实验结果对比，其中STAR和STHN分别表示本文提出的基于Star-Transformer的模型以及利用高速网络优化的Star-Transformer的模型。

	Level	Models	NE(%)	NM(%)	Overall(%)
Peng and Dredze (2015)	char	CRF	51.96	61.05	56.05
Peng and Dredze (2016)	char	LSTM	55.28	62.97	58.99
He and Sun (2017a)	char	LSTM	50.60	59.32	54.82
He and Sun (2017b)	char	LSTM	54.50	62.17	58.23
Zhang and Yang (2018)	char+Lattice	LSTM	53.04	62.25	58.79
Gui et al. (2019)	char+Lattice	CNN	**57.14**	**66.67**	59.92
Peng et al. (2019)	char+Lattice	LSTM	56.99	61.41	**61.24**
Yan et al. (2019)	char	Transformer	–	–	58.39
Li et al. (2020)	char+Lattice	Transformer	–	–	**63.42**
Our work	char	LSTM	53.16	60.70	55.76
	char	Transformer	46.90	53.45	48.96
	char	STAR	51.28	62.02	55.08
	char	STHN	52.32	64.53	56.63
	char+word	LSTM	58.82	69.32	64.88
	char+word	Transformer	53.02	64.11	59.79
	char+word	STAR	57.87	**72.04**	66.58
	char+word	STHN	**61.58**	69.45	**68.15**

表 5. 中文社交媒体命名实体识别实验结果对比(F_1)

STHN模型模型在特定实体NE的识别上性能达到了61.58%，比已有最好的模型结果高出了约4.44%；对于Weibo语料中特有的指代实体NM的识别，Star-Transformer模型获得了72.04%的F_1值；整体上本文提出的STHN模型取得了目前最好的综合性能68.15%，比之前最好的FLAT模型高出4.73%。

社交媒体类Weibo语料没有规范的文本内容，这使得词与词的边界更加模糊，比任何领域都迫切地需要词粒度信息的输入。就传统的LSTM模型而言，我们基于字与词粒度的实验已经有了不小的突破，综合性能为64.88%，比同样以LSTM作为主模型的Lexicon结构 (Peng et al., 2019)高了约3.64%。类似的，在融入了词粒度信息后，相同的模型在NE、NM以及整体上都能获得明显的提升。

表 6给出了本文实验的详细结果，在两种粒度上，STAR模型的三个指标都是明显高于Transformer模型的，这进一步验证了Star-Transformer在命名实体识别任务上的有效性。除此以外，基于字词粒度的STAR模型在召回率R值上作用显著，比相同条件下的LSTM模型高了约6.17%，而引入高速网络的STHN模型又比STAR模型高出了约1.24%。

Level	Models	P(%)	R(%)	F1(%)
Char	LSTM	60.86	51.45	55.76
	Transformer	57.70	42.51	48.96
	STAR	58.95	51.69	55.08
	STHN	60.00	**53.62**	**56.63**
Char + Word	LSTM	**75.66**	56.79	64.88
	Transformer	65.40	55.06	59.79
	STAR	70.64	62.96	66.58
	STHN	72.63	**64.20**	**68.15**

表 6. 详细实验结果对比

但是Star-Transformer的P值明显低于LSTM模型，虽然Star-Transformer已经是轻量级的Transformer，但本质上还是一个多连接计算注意力的模型，正是这样的机制使得Star-Transformer充分理解了句子的结构，识别出了更多的实体、提升了实体类型判别的准确度，但

同时存在过度识别的现象，从而导致P值的降低。

从解决上述问题的角度考虑，我们利用高速网络对Star-Transformer的每个卫星节点进行信息的自我桥接，实验结果显示STHN模型相较于STAR模型在P值上提升了近2%，拉近了与LSTM模型的距离。与此同时，STHN模型进一步提升了R值和F_1值。由此可见，高速网络的门控机制可以有效缓解Transformer的过度识别问题，同时带来命名实体识别性能的提升。

4.4 NE、NM结果对比分析

表 7给出了三个模型分别在特定实体(NE)和指代实体(NM)上面的实验结果。对于特定实体NE来说，整体趋势和表 6实验结果一致，LSTM仅在P值上占优势；Star-Transformer的引入带来了R值的大幅度提升，约为4.27%；我们最终的STHN模型进一步优化了Star-Transformer，在R和F_1值上都达到了最高值，虽然P值没有超越LSTM，但是已经尽可能将差距最小化。

表 7NM相关的数据体现了Star-Transformer对指代实体NM的识别性能，STAR模型在P、R和F_1三个指标上都比STHN模型的结果高。此外，STAR模型的R值比LSTM模型高出了约6.18%，可见相较于特定实体，Star-Transformer更适合用在指代实体NM的识别上。这也进一步验证了我们将Star-Transformer应用到社交媒体领域命名实体识别的有效性。

Models	NE			NM		
	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
LSTM	**71.92**	49.76	58.82	**77.22**	62.89	69.32
STAR	62.30	54.03	57.87	75.28	**69.07**	**72.04**
STHN	69.23	**55.45**	**61.58**	70.37	68.56	69.45

表 7. NE和NM详细结果分析

4.5 STHN效用分析

我们进一步统计并分析了Weibo语料上的实验结果，发现结合了Highway Networks的Star-Transformer模型能够识别出更多的Single类实体，表 8展示了相关数据。STHN模型比LSTM多识别出了67个Single实体，在数据量较小的weibo语料中占了约20%。这样超强的学习能力在带来性能提升的同时，也存在着过度识别的问题——将一些本不是实体的词识别为实体，从而导致P值的降低。

	总数	LSTM	STAR	STHN
Single	327	258	299	**325**
Else	175	98	**142**	89

表 8. 识别结果分析

表 9列举了几个在实验结果中出现的典型案例，很多在LSTM模型中被预测错误的实体，STHN模型能够准确地将其预测出来。Star-Transformer的连接机制能够捕获丰富的序列上下文信息，使得模型更好地理解句子结构，从而能够正确识别出更多的实体，这也是使R值大幅提升的关键。

Sentence	1.平洲玉器街... 2.中国女足打好基础再说吧...			
Word	平洲	玉器	街	中国女足
LSTM	B-LOC.NAM	M-LOC.NAM	E-LOC.NAM	S-PER.NAM
STHN	**S-LOC.NAM**	**B-LOC.NAM**	**E-LOC.NAM**	**S-ORG.NAM**

表 9. 识别案例分析

5 结论

本文根据公开的Weibo字粒度语料划分出了Weibo词粒度语料，并且提出了字词融合的方法，将字粒度与词粒度放在同等维度上加以考虑作为下层神经网络的输入，获得词语边界明确的句子表征。此外，我们分析了传统Transformer在命名实体识别任务上的劣势，并首次将Star-Transformer应用到社交媒体领域的命名实体识别任务中。由于Star-Transformer独特的星型拓扑结构，以及Highway Networks的动态特征调整，我们的STHN模型能够较好地理解句子序列的上下文信息，正确识别出更多的实体，在社交媒体领域的Weibo语料上取得了目前最好的效果。

社交媒体领域的Weibo语料规模较小，未来可以考虑将STHN模型应用到更多领域中，比如新闻领域，利用规模较大的语料深入研究Transformer在命名实体识别任务中的应用。

参考文献

Yekun Chai, Jin Shuo, and Xinwen Hou. 2020. Highway transformer: Self-gating enhanced self-attentive networks. *CoRR*, abs/2004.08178.

Jason P. C. Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional lstm-cnns. *Trans. Assoc. Comput. Linguistics*, 4:357–370.

Kevin Clark and Christopher D. Manning. 2016. Improving coreference resolution by learning entity-level distributed representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Yann N. Dauphin, Angela Fan, Michael Auli, and David Grangier. 2017. Language modeling with gated convolutional networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 933–941. PMLR.

1992. *Proceedings of the 4th Conference on Message Understanding, MUC 1992, McLean, Virginia, USA, June 16-18, 1992*. ACL.

Marie-Catherine de Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D. Manning. 2014. Universal stanford dependencies: A cross-linguistic typology. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014*, pages 4585–4592. European Language Resources Association (ELRA).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Chuanhai Dong, Jiajun Zhang, Chengqing Zong, Masanori Hattori, and Hui Di. 2016. Character-based LSTM-CRF with radical-level features for chinese named entity recognition. In Chin-Yew Lin, Nianwen Xue, Dongyan Zhao, Xuanjing Huang, and Yansong Feng, editors, *Natural Language Understanding and Intelligent Applications - 5th CCF Conference on Natural Language Processing and Chinese Computing, NLPCC 2016, and 24th International Conference on Computer Processing of Oriental Languages, ICCPOL 2016, Kunming, China, December 2-6, 2016, Proceedings*, volume 10102 of *Lecture Notes in Computer Science*, pages 239–250. Springer.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional sequence to sequence learning. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1243–1252. PMLR.

Zhaoheng Gong, Ping Chen, and Jiang Zhou. 2020. Integrating boundary assembling into a DNN framework for named entity recognition in chinese social media text. *CoRR*, abs/2002.11910.

Tao Gui, Ruotian Ma, Qi Zhang, Lujun Zhao, Yu-Gang Jiang, and Xuanjing Huang. 2019. Cnn-based chinese NER with lexicon rethinking. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 4982–4988. ijcai.org.

Qipeng Guo, Xipeng Qiu, Pengfei Liu, Yunfan Shao, Xiangyang Xue, and Zheng Zhang. 2019. Star-transformer. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1315–1325. Association for Computational Linguistics.

Hangfeng He and Xu Sun. 2017a. F-score driven max margin neural network for named entity recognition in chinese social media. In Mirella Lapata, Phil Blunsom, and Alexander Koller, editors, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers*, pages 713–718. Association for Computational Linguistics.

Hangfeng He and Xu Sun. 2017b. A unified model for cross-domain and semi-supervised named entity recognition in chinese social media. In Satinder P. Singh and Shaul Markovitch, editors, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3216–3222. AAAI Press.

Jingzhou He and Houfeng Wang. 2008. Chinese named entity recognition and word segmentation based on character. In *Third International Joint Conference on Natural Language Processing, IJCNLP 2008, Hyderabad, India, January 7-12, 2008*, pages 128–132. The Association for Computer Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *CoRR*, abs/1603.01360.

Haibo Li, Masato Hagiwara, Qi Li, and Heng Ji. 2014. Comparison of the impact of word segmentation on name tagging for chinese and japanese. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014*, pages 2532–2536. European Language Resources Association (ELRA).

Shen Li, Zhe Zhao, Renfen Hu, Wensi Li, Tao Liu, and Xiaoyong Du. 2018. Analogical reasoning on chinese morphological and semantic relations. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, pages 138–143. Association for Computational Linguistics.

Xiaonan Li, Hang Yan, Xipeng Qiu, and Xuanjing Huang. 2020. FLAT: chinese NER using flat-lattice transformer. *CoRR*, abs/2004.11795.

Zhangxun Liu, Conghui Zhu, and Tiejun Zhao. 2010. Chinese named entity recognition with a sequence labeling approach: Based on characters, or based on words? In De-Shuang Huang, Xiang Zhang, Carlos A. Reyes García, and Lei Zhang, editors, *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence, 6th International Conference on Intelligent Computing, ICIC 2010, Changsha, China, August 18-21, 2010. Proceedings*, volume 6216 of *Lecture Notes in Computer Science*, pages 634–640. Springer.

Xuezhe Ma and Eduard H. Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Nanyun Peng and Mark Dredze. 2015. Named entity recognition for chinese social media with jointly trained embeddings. In Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 548–554. The Association for Computational Linguistics.

Nanyun Peng and Mark Dredze. 2016. Improving named entity recognition for chinese social media with word segmentation representation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. The Association for Computer Linguistics.

Minlong Peng, Ruotian Ma, Qi Zhang, and Xuanjing Huang. 2019. Simplify the usage of lexicon in chinese NER. *CoRR*, abs/1908.05969.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Revanth Reddy, Danish Contractor, Dinesh Raghu, and Sachindra Joshi. 2019. Multi-level memory for task oriented dialogs. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 3744–3754. Association for Computational Linguistics.

Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. Highway networks. *CoRR*, abs/1505.00387.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Felix Wu, Angela Fan, Alexei Baevski, Yann N. Dauphin, and Michael Auli. 2019. Pay less attention with lightweight and dynamic convolutions. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Hang Yan, Bocao Deng, Xiaonan Li, and Xipeng Qiu. 2019. TENER: adapting transformer encoder for named entity recognition. *CoRR*, abs/1911.04474.

Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2014. Recurrent neural network regularization. *CoRR*, abs/1409.2329.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2003. Kernel methods for relation extraction. *Journal of machine learning research*, 3(Feb):1083–1106.

Yue Zhang and Jie Yang. 2018. Chinese NER using lattice LSTM. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1554–1564. Association for Computational Linguistics.

引入源端信息的机器译文自动评价方法研究

罗琪　　李茂西*
江西师范大学 计算机信息工程学院 / 江西 南昌 330022
Email:{luoqi，mosesli}@jxnu.edu.cn

摘要

机器译文自动评价是机器翻译中的一个重要任务。针对目前译文自动评价中完全忽略源语言句子信息，仅利用人工参考译文度量翻译质量的不足，该文提出了引入源语言句子信息的机器译文自动评价方法：从机器译文与其源语言句子组成的二元组中提取描述翻译质量的质量向量，并将其与基于语境词向量的译文自动评价方法利用深度神经网络进行融合。在WMT'19译文自动评价任务数据集上的实验结果表明，所提出的方法能够有效增强机器译文自动评价与人工评价的相关性。深入的实验分析进一步揭示了源语言句子信息在译文自动评价中发挥着重要的作用。

关键词： 机器翻译 ； 译文自动评价 ； 质量向量 ； 语境词向量 ； 自然语言推断

Research on Incorporating the Source Information to Automatic Evaluation of Machine Translation

Qi Luo　　**Maoxi Li***
School of Computer Information Engineering, Jiangxi Normal University
Nanchang, 330022, China
Email:{luoqi，mosesli}@jxnu.edu.cn

Abstract

Automatic evaluation of machine translation is one of the most critical tasks in machine translation. However, the source sentence information is completely ignored and only the reference is used to measure the translation quality in previous work. For this shortcoming, the paper presents a novel automatic evaluation metric incorporating the source information: extracting the quality embeddings that describes the translation quality from a tuple consist of the machine translations and their corresponding source sentences, and incorporating it into the automatic evaluation method based on contextual embeddings by using a deep neural network. The experimental results on the dataset of WMT'19 Metrics task show that the proposed method can effectively enhance the correlation between the results of the automatic evaluation metrics and that of the human judgments. Deep analysis further reveals that the information of the source sentences plays an important role in automatic evaluation of machine translation.

Keywords: machine translation , automatic evaluation of machine translation , quality embeddings , contextual embeddings , natural language inference

基金项目：国家自然科学基金(61662031, 61462044)

0 引言

机器译文自动评价是机器翻译的重要组成部分。它不仅能一定程度上度量翻译系统的整体性能，还能在翻译系统开发时指导其特征权值的优化。因此，研究机器译文自动评价对机器翻译的发展和应用有着重要的意义。

近年来，许多机器译文自动评价方法被相继提出，它们将机器翻译系统的输出译文与人工参考译文进行对比来定量刻画译文的质量。根据对比时涉及的语言知识层次，它们分为基于词语匹配的方法，如BLEU(Papineni et al., 2002)和NIST(Doddington, 2002)等；基于浅层句法结构匹配的方法，如POSBLEU(Popović and Ney, 2009)和POSF(Popović and Ney, 2009)等；基于深层语义信息匹配的方法，如引入复述的指标Meteor Universal(Banerjee and Lavie, 2005)和TERp(Snover et al., 2008)等、引入语义角色标注的指标MEANT(Lo, 2017)等等。随着深度学习的发展和其在自然语言处理中的广泛应用，一些研究者利用词语深度表示和神经网络结构对比翻译系统输出译文和人工参考译文进行译文自动评价，如基于静态词向量word2vec(Mikolov et al., 2013)的方法(Chen and Guo, 2015)、基于动态词向量BERT(Devlin et al., 2018)的方法(Mathur et al., 2019)、和基于神经网络结构的方法ReVal(Gupta et al., 2015)和RUSE(Shimanaka et al., 2018)等等。

然而，这些方法评价机器译文的主要思路还是遵循BLEU(Papineni et al., 2002)的基本观点："机器译文越接近于人工参考译文，其译文质量越高"。从这个观点出发，译文自动评价即是计算机器译文和人工参考译文的相似度。因此，译文自动评价过程中完全忽略了源语言句子，即在没有对源语言句子充分利用的基础上进行译文的自动评价。所以，找到结合源语言句子进行译文自动评价的切入点，势必能提高译文自动评价与人工评价的相关性。因此，我们尝试引入从源语言句子和其机器译文中提取的质量向量（Quality Embedding, QE），并将其与基于语境词向量的译文自动评价方法(Mathur et al., 2019)进行深度融合来增强译文自动评价，提高译文自动评价与人工评价的相关性。

1 相关工作

在基于深度神经网络的机器译文自动评价中，Lo(2017)，和Chen(2015)等人提出利用词语的分布式表示，静态预训练的词向量word2vec(Mikolov et al., 2013)，来提高机器译文和人工参考译文对比时同义词、近义词和复述等匹配的准确率。Guzmán(2019)等人提出了一种基于词向量和神经网络的机器译文自动评价方法，其目标是在给定人工参考译文的情况下，从一对机器译文中选择最佳译文，使用神经网络可以方便地融合由词向量捕获的丰富语法和语义表示。Gupta(2015)等人用基于树结构的长短时记忆网络(Tai et al., 2015)（Long Short-Term Memory network, LSTM）对机器译文和人工参考译文进行编码，根据两者之间元素差异和夹角计算机器译文的质量得分。Shimanaka(2018) 等人使用双向LSTM（Bidirectional LSTM, Bi-LSTM）对机器译文和人工参考译文进行编码，并利用多层感知机回归模型计算机器译文的质量得分。Mathur(2019)等人基于BERT(Devlin et al., 2018)语境词向量使用Bi-LSTM网络结构学习机器译文和人工参考译文的句子表示，并将自然语言推理中启发式方法(Mou et al., 2015)和增强序列推理模型(Chen et al., 2016)（Enhanced Sequential Inference Model, ESIM）引入到机器译文自动评价中，该方法在WMT'19译文自动评价任务（Metrics Task）上取得了优异的成绩，因此，本文将在Mathur(2019)等人的工作基础上，将利用源语言句子提取的质量向量融入译文自动评价中，进一步增强译文自动评价的性能。

2 背景知识

2.1 基于语境词向量的译文自动评价

自然语言推断关注假设结论（hypothesis）是否可以从前提语句（premise）中推断获取，它与译文自动评价任务非常类似。译文的质量越好，机器译文被人工参考译文表示（推断）的程度越高，同时人工参考译文被机器译文表示（推断）的程度也越高；反之亦然。在自然语言推断的框架下，Mathur(2019)等人使用语境词向量分别表示机器译文和人工参考译文，并根据两个表示的交互程度来度量机器译文的质量。使用自然语言推断中启发式方法(Mou et al., 2015)以及ESIM方法(Chen et al., 2016)，Mathur等人分别提出了$(\text{Bi-LSTM+attention})_{\text{BERT}}$译文自动评价方法和$(\text{ESIM})_{\text{BERT}}$译文自动评价方法。

2.1.1 (Bi-LSTM+attention)$_{\text{BERT}}$译文自动评价方法

将长度为l_r的人工参考译文r和长度为l_t的机器译文t分别利用BERT语境词向量进行表示，使用Bi-LSTM网络对其进行编码得到人工参考译文和机器译文包含上下文含义的新的向量表示$h_{r_{1:x}}(x=1\dots l_r)$、$h_{t_{1:y}}(y=1\dots l_t)$，通过向量点积求得人工参考译文和机器译文的相似度矩阵A，A中元素$a_{i,j}=h_{r_i}^T h_{t_j}$，利用相似度矩阵$A$，结合$h_r$和$h_t$，计算人工参考译文和机器译文的相互表示：

$$\tilde{h}_t=\sum_{i=1}^{l_r}\frac{\exp(a_{i,j})}{\sum_j\exp(a_{i,j})}\cdot h_r \tag{1}$$

$$\tilde{h}_r=\sum_{j=1}^{l_t}\frac{\exp(a_{i,j})}{\sum_i\exp(a_{i,j})}\cdot h_t \tag{2}$$

其中符号$\tilde{h}_t$表示h_r中每个词与h_t的相关程度，$\tilde{h}_r$表示h_t中每个词与h_r的相关程度。

为了避免向量$\tilde{h}_t$和$\tilde{h}_r$简单求和容易导致结果对序列长度敏感的问题(Chen et al., 2016)，对向量$\tilde{h}_t$和$\tilde{h}_r$分别进行最大池化和平均池化，将池化结果分别拼接得到向量v_t和v_r，并且启发式方法(Mou et al., 2015)被用作对局部推理进行增强得到增强后的表示向量m:

$$m=[v_t\oplus v_r\oplus(v_t\odot v_r)\oplus(v_t-v_r)] \tag{3}$$

其中符号“$\oplus$”表示向量拼接操作；符号“$\odot$”表示两个向量逐元素相乘操作。最后向量m被作为前馈神经网络的输入用于预测机器译文被人工参考译文表示的程度，即译文质量的得分。

2.1.2 (ESIM)$_{\text{BERT}}$译文自动评价方法

ESIM方法利用式(4)和(5)计算机器译文被人工参考译文表示的增强向量m_t和人工参考译文被机器译文表示的增强向量m_r。为降低模型参数复杂性，利用一个前馈神经网络层将m_t和m_r转换至模型的维度。Bi-LSTM网络被用作对降维后的信息进行编码，以便得到其局部信息的上下文表示向量。将编码后的向量进行平均池化和最大池化，并将池化后的结果$v_{r,avg}$、$v_{r,max}$和$v_{t,avg}$、$v_{t,max}$进行拼接，形成固定长度向量p，即：

$$m_t=[h_t\oplus\tilde{h}_t\oplus(h_t\odot\tilde{h}_t)\oplus(h_t-\tilde{h}_t)] \tag{4}$$

$$m_r=[h_r\oplus\tilde{h}_r\oplus(h_r\odot\tilde{h}_r)\oplus(h_r-\tilde{h}_r)] \tag{5}$$

$$p=[v_{r,avg}\oplus v_{r,max}\oplus v_{t,avg}\oplus v_{t,max}] \tag{6}$$

最后向量p被作为前馈神经网络的输入用于预测机器译文质量的得分。

2.2 译文质量向量提取方法

译文质量向量是译文质量估计中描述翻译质量的向量，它从源语言句子和其相应的译文中抽取，完全不需要借助人工参考译文进行计算。目前主流的质量向量提取方法包括基于循环神经网络（Recurrent Neural Network, RNN）的编码器-解码器模型(Bahdanau et al., 2014)的方法(Kim et al., 2017; Li et al., 2018)和基于Transformer模型(Vaswani et al., 2017)的方法(Fan et al., 2019; Wang et al., 2019)。它们将源语言句子和其机器译文使用强制学习的方式输入已训练好的神经机器翻译模型，截取在使用前馈神经网络进行$softmax$分类前一层网络的输出向量，作为机器译文当前位置词语的质量向量。

给定源语言句子，为了获取机器译文中每个词语的质量向量，基于联合神经网络的模型（Unified Neural Network for Quality Estimation, UNQE）(Li et al., 2018)被用作提取质量向量。联合神经网络模型使用译文质量估计任务数据集联合训练基于RNN的编码器-解码器模型和基于RNN的预测器，可以提取更优的质量向量，并且该模型在WMT18句子级别质量估计任务中取得了优异的成绩(Specia et al., 2018)，证实了其效果。

3 结合质量向量的机器译文自动评价

为了把源语言句子信息引入译文自动评价中，我们以质量向量作为切入点，将给定源语言句子情况下机器译文质量的表示和给定人工参考译文情况下机器译文的增强表示进行融合。模型结构如图1所示，其中符号src、mt和ref分别表示源语言句子、机器译文和人工参考译文。图左边描述通过UNQE方法(Li et al., 2018)从源语言句子和其机器译文中提取出描述翻译质量的词语级质量向量，并将其利用Bi-LSTM网络处理成句子级别的质量向量；图右边描述通过$(\text{Bi-LSTM+attention})_{\text{BERT}}$或$(\text{ESIM})_{\text{BERT}}$方法(Mathur et al., 2019)将机器译文和人工参考译文抽象为交互表示的增强向量，图上表示将质量向量与交互表示的增强向量进行拼接，将拼接后的向量输入前馈神经网络以预测机器译文质量得分。

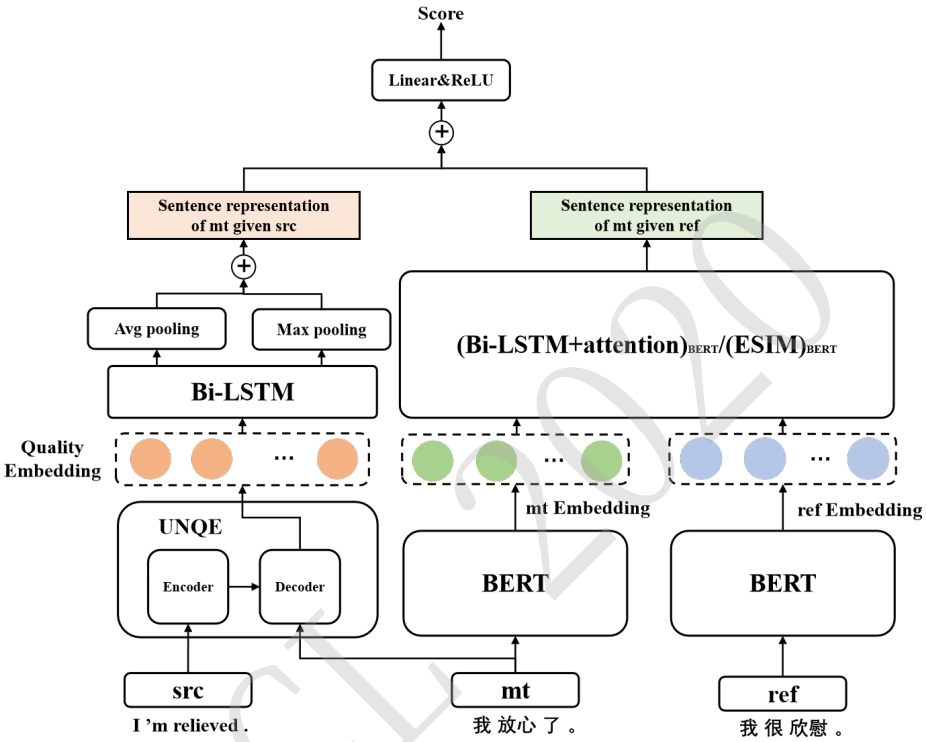


图 1. 引入译文质量向量增强机器译文自动评价的模型架构

3.1 $(\text{Bi-LSTM+attention})_{\text{BERT+QE}}$译文自动评价方法

由于从源语言句子和机器译文中抽取的质量向量是词语级的，即机器译文中每个词（token）使用一个实数向量描述其翻译质量，而机器译文和人工参考译文的交互表示增强向量是句子级，为了在同一层次将二者进行融合，需要将质量向量进一步抽象成句子级别表示。Bi-LSTM网络被用来对词语级质量向量$e_{qe_{1:k}}(k=1\ldots l_t)$进行编码，得到$e_{qe_{1:k}}$的包含上下文信息的向量$h_{qe,k}(k=1\ldots l_t)$，通过对$h_{qe}$进行最大池化和平均池化处理，将池化后的结果拼接即得到了句子的质量向量表示v_{qe}:

$$h_{qe,k} = \text{Bi-LSTM}\left(e_{qe}, k\right), \forall k \in [1, \ldots, l_t] \tag{7}$$

$$v_{qe,max} = \max_{k=1}^{l_t} h_{qe,k}, \quad v_{qe,avg} = \frac{1}{l_t}\sum_{k=1}^{l_t} h_{qe,k} \tag{8}$$

$$v_{qe} = [v_{qe,avg} \oplus v_{qe,max}] \tag{9}$$

其中符号$v_{qe,avg}$表示对h_{qe}进行平均池化后的结果，$v_{qe,max}$表示对h_{qe}进行最大池化后的结果，k表示句子中的词序号。

在机器译文和人工参考译文的交互表示增强向量方面，Bi-LSTM网络被用来对人工参考译文和机器译文的语境词向量编码，利用式(1)-(2)求得人工参考译文和机器译文的相互表示，随后利用式(8)的池化操作和式(9)的拼接操作求得了人工参考译文句子表示v_r和机器译文句子表示v_t。

为了将源端信息有效地引入机器译文自动评价模型中，我们将v_r和v_t进行局部信息增强组合，同时将增强后的信息与式(9)处理后的句子级别质量向量v_{qe}拼接起来形成新的固定长度向量$\tilde{m}$：

$$\tilde{m} = [v_t \oplus v_r \oplus (v_t \odot v_r) \oplus (v_t - v_r) \oplus v_{qe}] \tag{10}$$

其中符号v_t是$\tilde{h}_t$的平均池化后向量$v_{t,avg}$和最大池化后向量$v_{t,max}$拼接后形成的向量；v_r是$\tilde{h}_r$的平均池化后向量$v_{r,avg}$和最大池化后向量$v_{r,max}$拼接后形成的向量。最后将向量$\tilde{m}$作为前馈神经网络的输入，使用其预测译文的质量得分：

$$y_{\text{score}} = w^T \,\text{ReLU}\left(W^T \tilde{m} + b\right) + b' \tag{11}$$

其中参数w，W，b，b'均为前馈神经网络的权值。

为了训练模型的所有参数，译文自动评价得分y_{score}与人工评价得分h的均方误差被用来对模型进行优化，优化目标正式描述为：

$$loss = \frac{1}{M}\sum_{i=1}^{M}(y_{\text{score}}^{(i)} - h^{(i)})^2 \tag{12}$$

其中$y_{score}^{(i)}$为自动评价方法对待评价机器译文的打分，$h^{(i)}$为人工评价结果，M为训练集包含的样本数量。

3.2 $(\text{ESIM})_{\text{BERT+QE}}$译文自动评价方法

为了控制译文自动评价模型的复杂性，将对式(4)和(5)得到的机器译文和人工参考译文的局部信息表示m_t、m_r使用一个映射F转换至模型的维度后，经过Bi-LSTM进行编码，以得到其局部信息的上下文表示向量$\tilde{m}_t$和$\tilde{m}_r$，如式(13)-(14)所示。为了引入源端信息增强机器译文自动评价，我们将$\tilde{m}_t$和$\tilde{m}_r$平均池化和最大池化后的向量与机器译文质量估计的$v_{qe,avg}$和$v_{qe,max}$向量拼接得到新的信息组合向量$\tilde{p}$。将拼接后的信息表示向量作为前馈神经网络的输入以预测机器译文的质量分数：

$$\tilde{m}_{t,i} = \text{Bi-LSTM}\left(F\left(m_{t,i}\right), i\right), \forall i \in [1, \ldots, l_t] \tag{13}$$

$$\tilde{m}_{r,j} = \text{Bi-LSTM}\left(F\left(m_{r,j}\right), j\right), \forall j \in [1, \ldots, l_r] \tag{14}$$

$$\tilde{p} = [\tilde{v}_{r,avg} \oplus \tilde{v}_{r,max} \oplus \tilde{v}_{t,avg} \oplus \tilde{v}_{t,max} \oplus v_{qe,avg} \oplus v_{qe,max}] \tag{15}$$

$$y_{\text{score}} = w^T \,\text{ReLU}\left(W^T \tilde{p} + b\right) + b' \tag{16}$$

其中符号i、j均表示词序号，F表示激活函数为$ReLU$的单层前馈神经网络层；式(15)中的$\tilde{v}_{t,avg}$和$\tilde{v}_{t,max}$向量分别是$\tilde{m}_t$平均池化和最大池化的向量，$\tilde{v}_{r,avg}$和$\tilde{v}_{r,max}$分别是$\tilde{m}_r$平均池化和最大池化的向量；式(16)中的w，W，b，b'均为该前馈神经网络模型的参数。同样，模型的优化目标也在训练集上最小化译文自动评价得分y_{score}与人工评价得分h的均方差，同式(12)所示。

获取了机器译文句子级别分值后，我们对整个测试集（或文档集）中机器译文的句子级别得分取平均值作为翻译系统的系统级别（或文档级别）得分。

4 实验

4.1 实验设置

为了验证引入源端信息的机器译文自动评价方法的效果，我们在WMT'19 Metrics Task(Ma et al., 2019)的德英任务、中英任务和英中任务上进行实验。为了比较不同译文自动评价方法的性能，我们遵循WMT评测官方的做法利用皮尔森相关系数与肯德尔相关系数分别计算自动评价结果和人工评价结果的系统级别相关性和句子级别相关性，皮尔森相关系数或肯德尔相关系数越大，相关性越好。

UNQE提取的中英、英中任务上的质量向量维度为700，德英任务上质量向量维度为500。模型中Bi-LSTM隐藏层状态维度均固定为300，Dropout设置为0.2，使用Adam优化器优化训练，初始学习率为0.0004，训练批次大小为32，使用"bert-base-uncased"提取英文句子语境词向量，使用"bert-base-chinese"提取中文句子语境词向量。

在实验中，我们不仅将本文提出的方法与BLEU(Papineni et al., 2002)、chrF(Popović, 2015)以及BEER(Stanojević and Sima'an, 2014)等经典的方法进行了比较，而且与Mathur(2019)等人提出的自动评价方法、与不使用人工参考译文的译文质量估计方法UNQE(Li et al., 2018)进行了对比。需要说明的是Mathur等人是混合所有相同目标语言（比如德英和中英）译文自动评价训练集语料进行模型训练，而我们引入了源端信息，考虑实际译文打分需求且避免受不同源语言差异性的负面影响，我们针对每个语言对利用其训练集数据单独训练模型。德英语言对使用的是WMT'15-17 Metrics task(Bojar et al., 2015; Bojar et al., 2016; Ondrej et al., 2017)德英语言对的句子级别任务数据集。对于中英和英中语言对而言，单独训练可用训练集语料规模太小，因此加入了CWMT'18翻译质量评估在中英和英中语言对上的语料。德英方向按照9：1比例划分训练集和开发集，中英和英中方向完全使用CWMT'18翻译质量评估数据的训练集和开发集，具体数据统计如表1所示。测试集为WMT'19 Metrics Task的数据集，具体数据统计如表2所示。

	de-en	zh-en	en-zh
训练集	1458	8785	12865
开发集	162	1064	1040

表 1. 德英、中英和英中训练集、开发集数据统计

		de-en	zh-en	en-zh
WMT'19	systems	16	15	12
	sentences	2000	2000	1997
	sum	32000	30000	23964

表 2. WMT'19 Metrics task德英、中英和英中任务的测试集数据统计

4.2 实验结果

表3和表4分别给出了在WMT'19 Metrics task上引入源语言句子信息的译文自动评价方法与对比的译文自动评价方法与人工评价的句子级别和系统级别的相关性。

表3的数据表明引入源语言句子信息的方法"(Bi-LSTM+attention)$_{\text{BERT+QE}}$"和"(ESIM)$_{\text{BERT+QE}}$"在德英、中英和英中三个语言对上，与人工评价的句子级别相关性均值分别高于使用语境词向量的方法"(Bi-LSTM+attention)$_{\text{BERT}}$"和"(ESIM)$_{\text{BERT}}$"。"(Bi-LSTM+attention)$_{\text{BERT+QE}}$"相对于"(Bi-LSTM+attention)$_{\text{BERT}}$"在德英、中英、英中三个任务上分别提升了4.6%、3.2%和3.8%，"(ESIM)$_{\text{BERT+QE}}$"相对于"(ESIM)$_{\text{BERT}}$"方法分别提升了7.5%、2.8%和6.3%。其中"(Bi-LSTM+attention)$_{\text{BERT+QE}}$"方法在三个语言对任务中句子级别相关系数均是最高。这说明引入源端信息能增强机器译文自动评价与人工评价的句子级别相关性。

	de-en	zh-en	en-zh	avg.
UNQE	0.011	0.243	0.258	0.171
sentBLEU	0.056	0.323	0.270	0.216
BEER	0.128	0.371	0.232	0.244
chrF	0.122	0.371	0.301	0.265
(ESIM)$_{\text{BERT}}$	0.134	0.362	0.336	0.277
(Bi-LSTM+attention)$_{\text{BERT}}$	0.153	0.375	0.345	0.291
(ESIM)$_{\text{BERT+QE}}$	0.144	0.372	0.357	0.291
(Bi-LSTM+attention)$_{\text{BERT+QE}}$	**0.160**	**0.387**	**0.358**	**0.302**

表 3. WMT'19 Metrics Task的德英、中英和英中任务上自动评价与人工评价的句子级别相关性

	de-en	zh-en	en-zh	avg.
UNQE	0.264	0.688	0.916	0.623
BLEU	0.849	0.899	0.901	0.883
BEER	0.906	0.942	0.803	0.884
chrF	0.917	0.956	0.880	0.918
(ESIM)$_{\text{BERT}}$	0.896	0.951	0.967	0.938
(Bi-LSTM+attention)$_{\text{BERT}}$	0.910	0.956	0.965	0.944
(ESIM)$_{\text{BERT+QE}}$	0.896	0.958	**0.970**	0.941
(Bi-LSTM+attention)$_{\text{BERT+QE}}$	**0.917**	**0.972**	0.965	**0.951**

表 4. WMT'19 Metrics Task的德英、英中和中英任务上自动评价与人工评价的系统级别相关性

表4的数据表明本文所提方法"(Bi-LSTM+attention)$_{\text{BERT+QE}}$"和"(ESIM)$_{\text{BERT+QE}}$"在德英、中英和英中三个语言对评测任务上，与人工评价的系统级别相关系数的均值分别高于"(Bi-LSTM+attention)$_{\text{BERT}}$"和"(ESIM)$_{\text{BERT}}$"。"(Bi-LSTM+attention)$_{\text{BERT+QE}}$"相对于"(Bi-LSTM+attention)$_{\text{BERT}}$"方法在德英、中英任务上提升了0.8%和1.7%，在英中任务上保持一致，"(ESIM)$_{\text{BERT+QE}}$"相对于"(ESIM)$_{\text{BERT}}$"方法在中英、英中任务上分别提升了0.7%和0.3%，在德英上保持一致。这说明引入源端信息能增强机器译文自动评价与人工评价的系统级别相关性。

令人惊奇的是仅使用源端信息，完全不使用人工参考译文的UNQE方法也与人工评价结果有较好的相关性。尽管其在平均相关性上劣于所有使用人工参考译文的方法，但是它与sentBLEU方法在平均句子级别相关性和平均系统级别相关性上差距并不大，在英中的句子级别相关性(0.258)上甚至稍高于BEER方法(0.232)，在英中的系统级别相关性(0.916)上高于BLEU(0.901)、BEER(0.803)、chrF(0.880)等方法。这说明了源端信息对译文自动评价非常有帮助，从一个侧面佐证了正确地将质量向量引入译文自动评价必将提高译文自动评价的性能。

4.3 实验分析

为了进一步分析融合源端信息的译文自动评价方法的特点，我们在开发集上分别抽取了中英和英中翻译自动评价的实例进行分析。表5给出了对两个译文进行打分的实例，其中HTER是指将机器译文mt转换成人工后编辑的参考译文ref需要的最少编辑次数与译文长度的比值，它可以看作是译文人工打分的结果。自动评价方法对机器译文的打分越接近人工打分(1-HTER)，表明该自动评价方法对译文的评价越准确。

在第一个实例中，源语言句子中"**对城市交通来说**"在机器译文中缺乏对应翻译，存在漏译的情况，但(Bi-LSTM+attention)$_{\text{BERT}}$和(ESIM)$_{\text{BERT}}$却给了很高的分值，而本文的方法打分均更接近人工HTER分值。说明(Bi-LSTM+attention)$_{\text{BERT+QE}}$和(ESIM)$_{\text{BERT+QE}}$方法结合了源语言句子信息对译文进行评价，能更准确地描述译文的完整度特征，因此，相比于仅

结合人工参考译文信息打分的(Bi-LSTM+attention)$_{\text{BERT}}$和(ESIM)$_{\text{BERT}}$方法，引入源端信息的方法其评价更准确。在第二个实例中，机器译文中存在多译、过度翻译的情况，源语言句子中"*Tokyo*, *Japan*"被过度翻译成"**东京**"和"**日本**"两个地方。对于这种情况，本文方法依然比(Bi-LSTM+attention)$_{\text{BERT}}$和(ESIM)$_{\text{BERT}}$更接近人工打分结果HTER。这定性的说明了结合源端信息的机器译文自动评价方法能更充分利用源语言句子的信息对译文质量进行评价。

src：如此规模的城市发展**对城市交通来说**既是挑战，也是机遇。	
mt：This scale of urban development urban traffic is both a challenge and an opportunity.	
ref：This scale of urban development urban traffic is both a challenge and an opportunity to urban transportation.	
人工打分(1-HTER)：0.833	
(Bi-LSTM+attention)$_{\text{BERT}}$得分：0.883	(ESIM)$_{\text{BERT}}$得分：0.862
(Bi-LSTM+attention)$_{\text{BERT+QE}}$得分：0.833	(ESIM)$_{\text{BERT+QE}}$得分：0.845
src: The African Development Conference was dominated by Japan, and the previous five meetings were held in Tokyo, Japan or Yokohama, so this meeting will be the first move to Africa.	
mt: 非洲发展会议由日本主导，前五次会议分别在东京、**日本**或横滨举行，因此这次会议将是第一次到非洲的会议。	
ref：非洲开发会议由日本主导，此前的五次会议均是在日本东京或者横滨举行，因此，本次会议也将是首次移师非洲。	
人工打分(1-HTER)：0.836	
(Bi-LSTM+attention)$_{\text{BERT}}$得分：0.705	(ESIM)$_{\text{BERT}}$得分：0.904
(Bi-LSTM+attention)$_{\text{BERT+QE}}$得分：0.888	(ESIM)$_{\text{BERT+QE}}$得分：0.879

表 5. 不同自动评价方法对机器译文打分实例

5 结论

本文提出引入源端信息的机器译文自动评价方法。与传统的BLEU、BEER、chrF等评价指标相比，引入源端信息的机器译文自动评价方法，融合了源语言句子、人工参考译文、机器译文三者的信息，能更全面更有效地描述译文质量。在未来的工作中，我们将尝试在更大的语料库、更多的语言对上进行实验，以及引入更先进的模型和方法来挖掘源端信息，以提高机器译文自动评价方法的性能。

参考文献

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. In *Proceedings of the ICLR*, pages 1–15.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, et al. 2015. Findings of the 2015 workshop on statistical machine translation. In *Proceedings of the WMT*, pages 1–46.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, et al. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the WMT*, pages 131–198.

Boxing Chen and Hongyu Guo. 2015. Representation based translation evaluation metrics. In *Proceedings of the ACL and IJCNLP*, pages 150–155.

Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2016. Enhanced lstm for natural language inference. In *Proceedings of the ACL*, page 1657–1668.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the NAACL*, page 4171–4186.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurence statistics. In *Proceedings of the HLT*, pages 138–145.

Kai Fan, Jiayi Wang, Bo Li, Fengming Zhou, Boxing Chen, and Luo Si. 2019. "bilingual expert" can find translation errors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6367–6374.

Rohit Gupta, Constantin Orasan, and Josef van Genabith. 2015. Reval: A simple and effective machine translation evaluation metric based on recurrent neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1066–1072.

Francisco Guzmán, Shafiq Joty, Lluís Màrquez, and Preslav Nakov. 2019. Pairwise neural machine translation evaluation. In *Proceedings of the ACL and IJCNLP*, pages 805–814.

Hyun Kim, Hun-Young Jung, Hongseok Kwon, Jong-Hyeok Lee, and Seung-Hoon Na. 2017. Predictor-estimator: Neural quality estimation based on target word prediction for machine translation. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 17(1):1–22.

Maoxi Li, Qingyu Xiang, Zhiming Chen, and Mingwen Wang. 2018. A unified neural network for quality estimation of machine translation. *IEICE TRANSACTIONS on Information and Systems*, 101(9):2417–2421.

Chi-kiu Lo. 2017. Meant 2.0: Accurate semantic mt evaluation for any output language. In *Proceedings of the second conference on machine translation*, pages 589–597.

Qingsong Ma, Johnny Wei, Ondřej Bojar, and Yvette Graham. 2019. Results of the wmt19 metrics shared task: Segment-level and strong mt systems pose big challenges. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 62–90.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2019. Putting evaluation in context: Contextual embeddings improve machine translation evaluation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2799–2808.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Lili Mou, Rui Men, Ge Li, Yan Xu, Lu Zhang, Rui Yan, and Zhi Jin. 2015. Natural language inference by tree-based convolution and heuristic matching. In *Proceedings of the ACL*, pages 130–136.

Bojar Ondrej, Rajen Chatterjee, Federmann Christian, Graham Yvette, Haddow Barry, Huck Matthias, Koehn Philipp, Liu Qun, Logacheva Varvara, Monz Christof, et al. 2017. Findings of the 2017 conference on machine translation (wmt17). In *Proceedings of the WMT*, pages 169–214.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the ACL*, pages 311–318.

Maja Popović and Hermann Ney. 2009. Syntax-oriented evaluation measures for machine translation output. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 29–32.

Maja Popović. 2015. chrf: character n-gram f-score for automatic mt evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395.

Hiroki Shimanaka, Tomoyuki Kajiwara, and Mamoru Komachi. 2018. Ruse: Regressor using sentence embeddings for automatic machine translation evaluation. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 751–758.

Matthew Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. 2008. Terp system description. In *MetricsMATR workshop at AMTA*, pages 104–108.

Lucia Specia, Frédéric Blain, Varvara Logacheva, Ramón Astudillo, and André FT Martins. 2018. Findings of the wmt 2018 shared task on quality estimation. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 689–709.

Miloš Stanojević and Khalil Sima'an. 2014. Beer: Better evaluation as ranking. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 414–419.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the ACL and IJCNLP*, page 1556–1566.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Ziyang Wang, Hui Liu, Hexuan Chen, Kai Feng, Zeyang Wang, Bei Li, Chen Xu, Tong Xiao, and Jingbo Zhu. 2019. Niutrans submission for ccmt19 quality estimation task. In *China Conference on Machine Translation*, pages 82–92.

“细粒度英汉机器翻译错误分析语料库”的构建与思考

裘白莲[12]，王明文[1]，李茂西[1]，陈聪[1]，徐凡[1]
(1. 江西师范大学　计算机信息工程学院，江西　南昌　330022;
2. 华东交通大学　外国语学院，江西　南昌　330013)

摘要

机器翻译错误分析旨在找出机器译文中存在的错误，包括错误类型、错误分布等，它在机器翻译研究和应用中起着重要作用。该文将人工译后编辑与错误分析结合起来，对译后编辑操作进行错误标注，采用自动标注和人工标注相结合的方法，构建了一个细粒度英汉机器翻译错误分析语料库，其中每一个标注样本包括源语言句子、机器译文、人工参考译文、译后编辑译文、词错误率和错误类型标注；标注的错误类型包括增词、漏词、错词、词序错误、未译和命名实体翻译错误等。标注的一致性检验表明了标注的有效性；对标注语料的统计分析结果能有效地指导机器翻译系统的开发和人工译员的后编辑。

关键词： 机器翻译 ； 错误分析 ； 错误标注 ； 译后编辑

Construction of Fine-Grained Error Analysis Corpus of English-Chinese Machine Translation and Its Implications

Qiu Bailian[12] , Wang Mingwen[1] , Li Maoxi[1] , Chen Cong[1] , Xu Fan[1]
(1. School of Computer and Information Engineering, Jiangxi Normal University, Nanchang, Jiangxi 330022, China;
2. School of Foreign Languages, East China Jiaotong University, Nanchang, Jiangxi 330013, China)

Abstract

Machine translation error analysis, aimed at finding out problems in machine translation output, including error classes and error distribution etc., plays an important role in the research and application of machine translation. In this paper, post-editing is combined with error analysis with error labels annotated based on post-editing operations. Automatic error annotation and manual annotation are used to build a Fine-grained Error Analysis Corpus of English-Chinese Machine Translation (ErrAC), in which every annotated sample includes a source sentence, MT output, reference, post-edit, WER and error annotation. The annotated error classes include addition, omission, lexical error, word order error, untranslated word, named entity translation error etc. Annotator agreement analysis shows the effectiveness of the annotation. The statistics and analysis based on the annotated corpus can provide effective guidance for the development of machine translation system and post-editing practice.

Keywords: machine translation , error analysis , error annotation , post-editing

收稿日期：20– ; 定稿日期：20–
基金项目：国家自然科学基金(61876074，61662031，61772246); 教育部人文社科基金（16YJA740028）

1 引言

机器翻译质量评价是机器翻译研究的重要内容。机器翻译质量评价主要有人工评价和自动评价两种方式。由于人工评价成本较高，周期较长，不易获得，目前机器翻译质量评价大多采用自动评价指标，如BLEU(Papineni et al., 2002)，METEOR(Banerjee and Lavie, 2005)和TER(Snover et al., 2006)等，这些自动评价指标依据参考译文对机器译文给出整体得分，能够反映机器翻译质量整体情况，但是无法反映机器译文具体存在哪些问题，需要在哪些方面进行改进。为获取存在问题的具体信息，就需要进行机器翻译错误分析。错误分析可以找出机器译文中具体存在的问题，有助于了解机器翻译系统的不足，找准改进的方向，还可以为机器翻译质量估计、错误预测、自动译后编辑提供参考。近十几年来，错误分析在国外机器翻译研究领域受到重视，出现很多相关的研究，例如：(Koponen, 2010)使用错误分析评价机器翻译质量，(Bojar, 2011)分析了英捷机器翻译的错误类型，(Klubička et al., 2017)通过错误分析对NMT和PBMT进行细粒度的人工评价。但在国内相关研究还较少，仅有一些针对机器译文错误进行的语言学分析。例如，(罗季美and 李梅, 2012)将机器译文错误分为词汇错译、句法错译、符号错译三大类并展开分析，(罗季美, 2014)从短语和从句层面分析了机器翻译的句法错误。这些研究仅使用独立的人工译文与机器译文做对比展开分析，而且针对的是传统的机器翻译系统如RBMT，其错误分析的结果已经不能反应当前机器翻译的水平。(孙逸群, 2019)对5篇海洋类论文摘要机辅翻译中的错误进行了剖析。其错误分析侧重实例分析和改错，而且语料规模小，不具代表性。据我们了解，目前还没有专门针对英汉机器翻译错误分析可公开获得的语料库。值得注意的是，随着神经机器翻译的发展，机器翻译质量极大提高，但是英汉翻译方向神经机器翻译质量究竟如何，还存在哪些具体问题，针对这些问题还鲜有专门的错误分析，本文尝试针对这些问题展开研究与探讨。

错误分析和译后编辑是高度相关的工作，错误分析是找出机器译文的错误，译后编辑是改正机器译文的错误。错误分析和译后编辑都可以用来评价机器翻译的质量，但以往的研究大多把错误分析和译后编辑单独使用或单独作为研究对象，较少有把两者结合起来的研究。我们将译后编辑和错误分析结合起来，先对机器译文进行译后编辑，然后以译后编辑译文(PE译文)作为参照，对机器译文进行错误标注。在此基础上，构建了一个细粒度英汉机器翻译错误分析语料库(Fine-grained **Err**or **A**nalysis **C**orpus of English-Chinese Machine Translation, 简称为ErrAC)。PE译文比参考译文更适合作为错误标注参照的原因在于，翻译本来就存在一文多译的现象，同一个源语言句子可以有多种不同的正确译文，而在机器译文的基础上进行译后编辑，力求PE译文是最接近机器译文的正确译文，其编辑距离最短。因此，以PE译文来衡量机器翻译的质量相对而言更客观，更能准确地找出机器翻译真正存在的问题。(Snover et al., 2006)研究结果表明，使用人工译后编辑译文得到的HTER值，比最接近机器译文的参考译文的TER，更能准确地衡量机器翻译的质量，而且，HTER与人工评价的相关性比BLEU与人工评价的相关性更高。下面给出了WMT19新闻机器翻译测试集上的两个实例，它们表明以人工参考译文和PE译文作为错误分析参照的区别。

例1.

源语言句子： It would be extremely ill advised to venture out into the desert on foot with the threat of tropical rainfall.
机器译文： 在 热带 降雨 的 威胁 下 ， 徒步 冒险 进入 沙漠 是 极 不 明智 的 。
PE译文： 在 热带 降雨 的 威胁 下 ， 徒步 冒险 进入 沙漠 是 极 不 明智 的 。
参考译文：由于 热带 降雨 的 威胁 ， 沙漠 冒险 活动 将 十分 危险 。
PE译文WER 0.00 参考译文WER 76.92

例2.

源语言句子： Do you think he's telling the truth to the country?
机器译文： 你 认为 他 对 国家 说 的 是 真话 吗 ？
PE译文： 你 认为 他 对 国人 说 的 是 真话 吗 ？
参考译文：你 觉得 他 对 国人 所 说 的 是 事实 吗 ？
PE译文WER 8.33 参考译文WER 35.71

从例1可见，机器译文是正确的译文，达到了翻译的忠实、通顺的要求，但是与参考译文有很大的差别。如果按照参考译文来标注错误，那么会得出这一机器译文质量低劣的结果，其WER值(Word Error Rate，词错误率)高达76.92，这样显然无法准确、有效地衡量机器译文质量。例2中，译后编辑实际上只需要一次替换的编辑操作，即修改一处错词，就可以达到忠实、通顺的要求，PE译文WER为8.33，但是机器译文与参考译文的差别较大，WER为35.71，把机器译文修改成参考译文需要三次替换操作和一次插入操作。由此可见，使用PE译文作为参照对机器译文进行错误标注，比直接使用参考译文更客观，更能有效地反映机器译文的质量，更能准确地反映机器翻译系统的问题。

本文工作的意义体现为以下四个方面：1)获得对神经机器翻译质量更客观、更准确的评价；2)为机器翻译系统开发、译后编辑工作提供参考；3)可以为机器翻译质量估计、错误预测、自动译后编辑提供数据和参考；4)可用于错误类型与自动评价指标、译后编辑工作量之间相关性的研究。

下文结构如下：第2节介绍错误分析和译后编辑相关研究和相关语料库建设情况；第3节介绍语料来源和语料库构建过程；第4节对错误标注结果进行统计与分析；第5节总结全文。

2 相关工作

错误分析可以以人工和自动两种方式进行。(Vilar et al., 2006)建立了人工错误分析的框架，定义了错误类型，根据错误分类对机器译文进行错误标注。(Popović et al., 2006)提出基于屈折变化和句法信息的自动错误分析框架，自动获得错误的细节信息。机器翻译错误分析主要有以下几种应用。第一，用于评价某一机器翻译系统的质量(Vilar et al., 2006; Lommel et al., 2014)，或比较几种不同的机器翻译系统，通常是比较SMT和NMT等不同系统(Bentivogli et al., 2016; Toral and Sánchez-Cartagena, 2017; Bentivogli et al., 2018)；第二，考察不同错误类型对机器翻译质量的影响(Popovic et al., 2013; Federico et al., 2014)；第三，用于译后编辑的相关研究，考察不同错误类型对译后编辑工作量不同方面的影响(Krings, 2001; Zaretskaya et al., 2016)。但是，这些研究是在机器译文上进行错误分析，或者以参考译文为参照进行错误分析，不是以PE译文为参考，这会导致错误分析与实际情况存在偏差。

随着译后编辑在翻译行业越来越普遍，逐渐出现了一些可公开获得的译后编辑语料(Potet et al., 2012)。WMT从2012年开始质量估计子任务，从2015年开始自动译后编辑子任务，这两个子任务都提供了译后编辑译文语料。部分语料还有错误标注，包括基本的编辑距离操作如替换、删除、插入和移位，或者“好”、“差”二元标签，其语言对涉及英德、英俄等。CWMT从2018年开始翻译质量估计任务，提供英汉语言对机器翻译译后编辑译文，部分语料有“好”、“差”二元标签，部分语料有每个句子的HTER值。这些语料对机器翻译质量估计、错误预测、自动译后编辑、译后编辑人员培训都非常有用。

同时，还出现了一些做了错误标注的译后编辑语料库。例如，TRACE语料库包含法英、英法译后编辑译文，其中有基本编辑距离错误类型的标注(Wisniewski et al., 2013)。(Koponen, 2012)使用英西机器翻译语料，提供译后编辑译文，对语料进行错误标注，研究错误类型与估计的译后编辑工作量、实际编辑操作之间的关系，但是其语料不能公开获得。

Terra语料库(Fishel et al., 2012)是可以公开获得的人工错误标注语料库，用于自动错误分类工具Addicter(Zeman et al., 2011)和Hjerson(Popović, 2011)的评估。这个语料库由不同研究小组独立标注，标注策略各不相同，有的小组不使用参考译文，有的小组使用参考译文。这样会导致错误标注一致性不高，因为标注策略不同，标注的结果会有较大差异。而且，这项工作中人工错误分类和自动错误分类是完全独立进行的。TARAXü语料库(Avramidis et al., 2014)能够公开获得，该语料库包含译后编辑译文和机器翻译错误标注数据，但是这两项工作是完全独立进行的，而且不是在同一个数据集上进行。PE2rr语料库(Popović and Arcan, 2016)在译后编辑译文的基础上进行错误标注，更准确地反映了机器译文的错误情况，可以公开获得，但是该语料库只包含英语、塞尔维亚语、德语、西班牙语之间的语料。这些语言均属于印欧语系，语言之间的差别相对较小，其错误分析的数据可能无法一般化。英语和汉语分属于不同的语系，差别较大，有的印欧语系语言之间机器翻译常见的错误如屈折错误在英汉语言方向上没有，而有的错误类型则可能比较突出，那么英汉机器翻译与其它相同语系语言之间机器翻译的错误情况、错误分布有没有差异，有什么差异，这就需要专门进行英汉机器翻译错误分析，而目前英

新闻类别	句子数	源语言句子词数	机器译文词数	编辑词数(%)
政治	700	15425	18622	2365(12.5)
经济	112	2473	3019	473(15.4)
社会	494	9293	10958	1532 (13.7)
体育	305	6269	8154	2144(25.7)
科教	174	3791	4670	651(13.6)
文艺	212	4783	6270	1215(18.9)
总数	1997	42034	51693	8380(16.2)

表 1: 源语言句子词数、机器译文词数及编辑词数

汉语言对机器翻译质量评价和错误分析还缺少类似的语料库。

3 语料库构建

本节介绍语料来源和语料库构建过程。语料库构建过程分为两个阶段，译后编辑和错误标注。首先由专业人士进行译后编辑，然后采用自动错误标注加人工标注的方式进行错误标注。

3.1 语料来源

我们的语料来源为WMT2019新闻机器翻译测试集英中翻译方向，该测试集包括源语言句子、机器译文和人工翻译的参考译文。我们将测试集按照新闻内容分为六类：政治、经济、社会、体育、科教和文艺。我们使用的机器译文是KSAI组(金山AI)提交的机器译文，该小组在英中机器翻译任务中人工打分排名第一。KSAI提交的机器译文是基于各种神经机器翻译模型，以Transformer作为基线系统，使用了几种数据过滤和回译作为数据清洁和数据增强的方法。最终模型是经过多模型集成、重排序、后处理的系统组合(Barrault et al., 2019)。语料库的统计信息见表1，句子数为1997个，源语言句子词数为42034，机器译文词数为51693，编辑词数为8380。编辑词数百分比是按照编辑词数与机器译文词数加漏词数量的百分比来计算的。

3.2 译后编辑

进行本次译后编辑工作的译后编辑人员为2名翻译专业教师，均精通英汉两种语言，具有丰富的翻译和译审经验。为保证译后编辑质量，在进行译后编辑之前，译后编辑人员经过多次讨论和修改，知晓此次译后编辑的目标和原则。译后编辑的目标是修改机器译文的错误，使译文达到忠实源语言句子、语句通顺的要求，质量适中即可。本次译后编辑采取轻度译后编辑的原则，即只进行最少量的必要的编辑操作以达到译文质量可接受的效果，不考虑风格、文采问题，也不考虑译后编辑人员在用词习惯、语法结构等方面的个人喜好问题。针对本次译后编辑制定五条具体指南如下：(1)力求译文意思正确、语句通顺；(2)确保没有信息增加或遗漏；(3)尽可能多地使用机器译文；(4)除非影响语义，否则不修改句子结构；(5)单纯的风格问题无需修改。

新闻类别	译后编辑操作(以WER作为编辑距离)			
	无 0	低 0-25%	中 25-50%	高 >50%
政治	29.7	51.9	12.1	6.3
经济	17.0	53.6	20.5	8.9
社会	33.8	44.1	16.0	6.1
体育	14.4	37.1	27.5	21.0
科教	23.6	57.5	16.1	2.9
文艺	19.3	47.2	20.8	12.7
总数	26.0	47.8	17.2	9.0

表 2: 译后编辑工作量等级分布

表1表明了语料库句子数量、源语言句子词数、机器译文词数，以及机器译文经过译后编辑的编辑词数。整体编辑词数百分比为16.2%，需要进行编辑修改的比例不是很大，这表明在新

闻翻译对质量要求适中的应用场景中，在领域语料比较丰富的基础上，神经机器翻译质量达到很大程度可接受的水平，机器译文在很大程度上可用。在各种新闻类别中，编辑词数百分比最大的是体育新闻，达到25.7%，是出现错误最多、需要译后编辑量最大的新闻类别。而编辑词数百分比最小的是政治新闻，为12.5%，是需要译后编辑量最小的新闻类别。可见，不同新闻类别之间机器翻译质量的差别较大，其可能的原因在于相关领域训练语料规模的大小。

我们以WER表示机器译文每个句子的编辑距离，按照编辑距离的大小将所需的译后编辑工作量(体现为所进行的实际编辑操作)分为四个等级，结果见表2。从表2中可以看出，有26%的句子已经可接受，无需任何编辑操作，47.8%的句子只需要少量编辑操作即可达到质量适中的要求，17.2%需要中等编辑工作量，只有9%的句子需要进行大量修改。ErrAC语料库中也给出了每个句子的WER值。在不同新闻类别中，体育类需要的译后编辑工作量相对较高，不需要编辑操作的比例为14.4%，低于其它所有类别，而需要进行大量译后编辑操作的比例达到21%。

3.3 错误标注

错误标注工作分两个阶段进行。首先，以PE译文为参考，使用Hjerson自动错误标注工具进行错误标注；然后，将自动错误标注的结果一一进行人工核对和修改，并细化和扩展错误类型。

进行错误标注之前先对中文语料进行预处理，采用清华THULAC分词工具进行分词。Hjerson工具以机器译文和PE译文作为输入，以词为单位进行错误标注，输出错误标注结果。Hjerson工具可以识别和标注五种类型的错误，即增词、漏词、错词、词序错误和屈折错误(动词时态/人称/情态/格/性/数)。Hjerson工具主要是针对英语、德语等印欧语系语言开发的，其中的屈折错误常出现于印欧语系语言之间的翻译中，而英汉机器翻译的目的语为汉语，汉语不是屈折语言，没有屈折错误，因此Hjerson实际上标注出来的错误有四种，即增词、漏词、错词和词序错误，在ErrAC语料库中分别以ext、miss、lex和reord表示。除漏词错误，所有其他错误均针对机器译文做标注。其中，在机器译文中出现了而在PE译文中没有出现的词标注为增词。在PE译文中出现了而在机器译文中没有出现的词标注为漏词。漏词错误需要针对PE译文做标注，因为漏词是机器译文中没有的词，无法在机器译文的标注中体现，在PE译文上做标注，才能体现漏词错误及漏词的位置。

自动标注之后进行人工标注，标注者为本文作者之一，知晓标注规则和方法。在人工标注阶段，除核对和修改自动错误标注，还对错误类型进行了细化和扩展。细化针对增词错误类型，细化的标注有两种，一是数词加量词，二是人称代词加结构助词。由于英汉语言习惯的差别，这两种增词错误是英汉翻译中经常出现的问题，在机器翻译中更为明显。英文中的冠词a或an，在机器翻译中常被译为一个、一种、一名等，而很多情况下按照汉语的习惯用法这些是应该省略的，如例3所示。数词和量词的增词分别标注为ext-num和ext-cla，其出现次数分别为81次和83次，占增词总数的4.76%和4.87%。

例3.

源语言句子： Thomas Bjorn, the European captain, knows from experience that a sizeable lead heading into the last-day singles in the Ryder Cup can easily turn into an uncomfortable ride.
机器译文：欧洲 队长 托马斯 · 比约恩 （Thomas Bjorn） 从 经验 中 知道 ， 在 莱德杯 最后 一 天 的 单打 比赛 中 ， 一 个 相当 大 的 领先 优势 很 容易 演变 成 一 场 不 舒服 的 比赛 。
PE译文：欧洲 队长 托马斯 · 比约恩 （Thomas Bjorn） 根据 经验 知道 ， 在 莱德杯 最后 一 天 的 单打 比赛 中 ， 大比分 的 领先 优势 也 很 容易 变 成 不利 局面 。
机器译文标注：x x x x x x x x x lex x lex x x x x x x x x x x x x x ext-num ext-cla lex lex x x x x x lex x ext-num ext-cla lex lex lex lex x
PE译文标注： x x x x x x x x x lex x x x x x x x x x x x x x x lex x x x miss x x lex x lex lex x

此外，英语中的人称代词we/he/she/they等以及其相应物主代词our/his/her/their等，在机器翻译中基本都按原本译出，但是根据汉语使用习惯，很多时候在译文中都应该省略，否则译文不自然、不通顺，如例4所示。人称代词和结构助词增词分别标注为ext-pro和ext-aux，分

别出现114次和84次，分别占增词总数的6.69%和4.93%。

人工标注阶段扩展的三种错误类型为未译、命名实体翻译错误和标点符号错误。机器译文中出现了一些未经翻译的英文单词，标注为untr。机器译文中还出现了一些命名实体翻译错误或命名实体翻译前后不一致的问题，包括人名、地名、组织结构名称等。未译的大多都是命名实体，但因为错误形式不同，所以做了区分。命名实体翻译错误标注为nen。此外，还有标点符号错误、多余或遗漏的问题，这类问题全部归类为标点符号错误，标注为punc。

例4.

源语言句子： We've transformed the look and feel of our beauty aisles to enhance the environment for our customers.
机器译文： 我们 已经 改变 了 我们 美容 通道 的 外观 和 感觉 ， 为 我们 的 客户 改善 了 环境 。
PE译文： 我们 已经 改变 了 美容 通道 的 外观 和 氛围 ， 为 客户 改善 环境 。
机器译文标注：x x x x ext-pro x x x x x lex x x ext-pro ext-aux x x ext x x
PE译文标注：x x x x x x x x x lex x x x x x x

除了细化和扩展错误类型，在人工标注阶段还进行了多标签错误标注。因为有的词存在多种错误，如错词、未译、命名实体翻译错误也可能出现在错误的位置上，即同时也是词序错误。这种情况自动错误标注工具无法标注，在人工阶段做了补充，针对叠加的词序错误标注了多错误标签，在语料库中表示为+reord。

错误标注完成之后，为检验标注质量，我们进行了标注者一致性分析。我们采用取样的方法，取数据集中前100个句子，分别由A1和A2两位标注者独立进行标注，两位标注者均经过培训，知晓标注规则和方法。错误标注不是简单的打分或排序，它涉及所标注的错误数量、错误类型和标注的位置，标注者一致性不容易计算。我们采用(Stymne and Ahrenberg, 2012)关于错误标注不同标注者一致性的计算方法，该计算方法关注所标注错误的共现情况，即

$$Agreement = \frac{2 * A^{agree}}{A1^{all} + A2^{all}} \quad (1)$$

其中上标all表示每位标注者标注的总数，上标agree表示两位标注者标注错误类型相同的数量。不同标注者一致性详见表3，整体一致性达90.6%。可见，在自动标注工具的基础上进行人工修改，不仅提高了错误标注效率，也有助于提高标注者一致性。

不同标注者一致性			
错词	88.8%	未译	100%
增词	74.9%	命名实体	100%
漏词	97.6%	标点符号	100%
词序	99.5%	总数	90.6%

表 3: 不同标注者一致性

该计算方法关注所标注错误的类型和数量，没有考虑标注错误的位置。在ErrAC语料库中，我们经过观察发现，不同标注者出现标注位置不一致的主要是词序错误，即reord的标注位置会有差异，其他错误类型的标注位置基本上差异不大。各种错误类型中，增词的标注者一致性相对较低，这是因为在英汉翻译中，词与词并不是一一对应的，词一对多、多对一的情况很常见，会造成标注者对于某个词是属于增词还是错词的标注产生差异。例如源语言句子中"holiday homes"，机器译文为"度假 之 家"，PE译文为"度假屋"，标注者A1标注为"lex lex lex"，标注者A2标注为"lex ext lex"。两者对"之"字的错误类型标注不一致，分歧的原因在于标注者A1将"度假 之 家"三个词理解为对应源语言句子"holiday homes"两个词，而标注者A2的理解是"度假"对应源语言句子"holiday"，"家"对应源语言句子"home"，那么"之"就理解为是增词。

采用同样的计算方法，我们还计算了同一标注者一致性。在标注者A1完成第一次标注之后，间隔两个月的时间，随机取数据集中100个句子再次进行标注。经过计算得出，同一标注者一致性为93.6%。

	各种错误类型数量						
新闻类别	增词	错词+词序错误	漏词	词序错误	未译+词序错误	命名实体+词序错误	标点符号
政治	500	1087 +62	326	414	56 +0	38 +1	1
经济	104	226 +16	53	56	47 +2	3 +0	7
社会	287	662 +38	237	241	98 +0	42 +0	9
体育	425	1131 +64	202	303	23 +5	124 +5	21
科教	155	262 +16	101	96	40 +2	12 +1	6
文艺	232	493 +41	165	205	88 +3	65 +1	25
总数	1703	3861 +239	1084	1315	354 +10	284 +8	87

表 4: 错误类型数量

	各种错误类型错误率(%)						
新闻类别	增词	错词+词序错误	漏词	词序错误	未译+词序错误	命名实体+词序错误	标点符号
政治	2.68	5.84 +0.33	1.75	2.22	0.30 +0.00	0.20 +0.01	0.01
经济	3.44	7.49 +0.53	1.76	1.85	1.56 +0.07	0.10 +0.00	0.23
社会	2.62	6.04 +0.35	2.16	2.20	0.89 +0.00	0.38 +0.00	0.08
体育	5.21	13.87 +0.78	2.48	3.72	0.31 +0.03	1.52 +0.06	0.26
科教	3.32	5.61 +0.34	2.16	2.06	0.86 +0.04	0.26 +0.02	0.13
文艺	3.70	7.86 +0.65	2.63	3.27	1.40 +0.05	1.04 +0.02	0.40
总数	3.29	7.47 +0.46	2.10	2.54	0.68 +0.02	0.55 +0.02	0.17

表 5: 错误率(注：错误率为错误数量与文本总词数的百分比)

4 统计与分析

我们对错误标注结果做了统计，每种错误类型的数量和错误率见表4和表5。错误率是错误数量与文本总词数的百分比，这样方便对不同的机器译文进行错误分析时相互比较。从表4可见，数量最多的错误类型是错词，即在机器翻译中选择了错误的词汇进行翻译，错词数量为3861，约占编辑词数的46%。其次是增词，数量为1703，约占编辑词数的20%。词序错误和漏词分别约占16%和13%。

错误分析对机器翻译系统开发具有很好的参考价值，其主要意义在于，有助于了解机器翻译系统存在的具体问题，了解系统的不足和短板，明确改进的方向，为机器翻译系统开发提供参考。我们对神经机器翻译译文进行错误分析，根据所发现的主要问题，对机器翻译系统开发提出建议如下。

第一，针对一词多义问题。通过错误分析可知，错词问题是神经机器翻译的主要问题。机器译文中错词问题大多是因为源语言句子中一词多义，而目前的神经机器翻译技术没有对句子进行真正的理解，无法根据领域和上下文信息来选择正确的义项，导致翻译时选词错误。建议机器翻译系统开发时，一方面通过引入外部的领域知识库或知识图谱，充分利用外部知识，另一方面通过大型单语语料库训练准确的语境词向量进行词义消岐，充分利用上下文信息，来缓解一词多义导致的错词问题。

第二，针对增词错误。在ErrAC语料库中，代词加结构助词、数词加量词这两种类型的增词占增词总数的21.25%。在机器翻译系统开发时，可以考虑对这些词类的翻译设置一定的约束，同时还需要提高训练语料的质量。如果训练语料在这些词类的翻译上处理得比较好，神经机器翻译在这方面也会有更好的表现。

第三，针对术语翻译错误。以体育类新闻为例，体育类新闻中错词的数量多达1131处，占语料库中错词总数(3861)的29.3%，其错误率为13.87%。原因在于，体育类新闻中很多词是专业术语，在译文中也需要对等地翻译成专业术语，而机器翻译往往把这些词按照常用义项译出，没有根据领域来选择合适的义项，导致翻译错误。比如，The attempt sailed high above the box，句中的“box”，机器翻译为“盒子”，而在足球术语中应为“禁区”。建议开发机器翻译系统时，引入相关领域的术语词典资源，并使系统在待译文本输入时可以识别其所属领域，即时调用相关领域术语资源，以缓解术语翻译错误的问题。

第四，针对代词引起的翻译错误问题。机器翻译中由于对代词指代对象不明，导致出现翻译错误的情况很多，有时甚至引起整个句子的意思出现偏差。代词指代不明有多种原因，比如，句中代词可指代的对象有多个，导致代词指代模糊；或者代词的指代对象距离代词很远，

跨越了单个句子。目前神经机器翻译模型大多是句子级别的，无法很好地利用篇章上下文信息解决跨越句子的指代问题。建议开发和改善以段落、篇章为输入单元的翻译模型，开发基于篇章级别的神经机器翻译系统。这样的系统还可以获取句子之间的依赖关系，更连贯地翻译整个篇章文本。

第五，针对缺乏训练语料问题。领域相关语料稀缺会直接影响翻译质量，比如，体育类新闻中命名实体翻译错误多达124处，占语料库中命名实体翻译错误总数(284)的43.7%。原因在于，体育类新闻中人名、球队名、俱乐部名称等出现的频率比其他类新闻更高。在机器译文中，这些命名实体翻译出现译错以及翻译前后不一致的情况很多。这些命名实体不能正确翻译的直接原因是相关领域的训练语料较少。针对这一问题，一方面当然是尽可能增加语料的数量，扩大训练语料的覆盖度，另一方面是提高训练语料的质量。应当避免直接从网上爬取双语语料作为训练语料，而要仔细甄别双语语料的质量，使用高质量的双语语料。获得大量高质量的双语语料对于提高神经机器翻译质量具有决定性作用。此外，针对命名实体翻译的问题，建议在机器翻译系统中加入命名实体翻译检查机制，检查并改正命名实体翻译前后不一致的情况。

从ErrAC语料库的数据中可以总结出一些经验教训供译后编辑人员参考。

第一，关注一词多义引起的错词问题。各种类型错误中，错词数量最多，达到3861次，可见一词多义仍然是机器翻译的一个障碍，目前神经机器翻译系统还无法根据领域和上下文选择正确的词义进行翻译。因此，在译后编辑过程中，需要关注一个词在不同领域、不同上下文中表达的不同意义，关注词义选择的问题，提高译后编辑的准确率和效率。

第二，善于发现和修改词序错误能有效提高译后编辑效率。词序错误占编辑词数的16%。据(Kirchhoff et al., 2012)研究发现，词序错误是机器翻译使用者最不喜欢的错误类型。其原因可能在于词序错误更难发现和修改，特别是长距离词序错误。(Popovic et al., 2014)发现，错词和词序错误所需要的认知努力最大。如果是错词叠加词序错误，需要的译后编辑认知努力更大，需要的译后编辑时间更多。因此，词序错误所需要的译后编辑工作量可能相对较大，在译后编辑过程中需要予以关注。译后编辑人员应该熟悉中英文在词序方面的差异，增强对翻译中词序问题的敏感性。

第三，在ErrAC语料库中，增词错误数量较多，但相对比较容易修改。(Popovic et al., 2014)发现，删除增词的编辑操作所需要的译后编辑认知努力和时间最少。而且，关于增词错误，还可以关注代词加结构助词、数词加量词这样的增词，在本语料库中，这几种类型的增词占增词总数的21.25%。这样有针对性地进行译后编辑，有助于提高译后编辑的速度和效率。

第四，具备全局意识，从篇章整体的角度修改错误。在机器译文中，经常出现命名实体翻译前后不一致的问题，影响篇章的连贯性，导致译文读者理解困难。虽然译后编辑人员在篇章全局的理解和把握上有优势，但有时容易忽略篇章信息，更多关注单个句子的细节。因此，在译后编辑过程中需要对该问题予以注意，修改译名不一致的问题，保证命名实体翻译前后一致，加强译文篇章的连贯性和可读性。

第五，适当关注标点符号，根据中文习惯来修改。在英汉翻译中，受英文句子结构的影响，机器译文常出现中文长句。在译后编辑过程中，需要根据中文习惯合理断句，插入标点符号，尤其是逗号。在ErrAC语料库中，插入标点符号的译后编辑操作达165次，其中大多数是插入逗号。

最后，加强对机器翻译的了解。译后编辑人员除了需要具备扎实的双语能力和翻译能力，还需要对机器翻译有较好的了解。他们需要了解机器翻译系统的不足和问题，熟悉机器译文中常出现的错误，尝试摸索总结其错误模式，并掌握有针对性的纠错方法。只有在译后编辑实践中不断积累经验，才能不断提高译后编辑的质量和效率。译后编辑人员可以充分利用机器翻译提供的便利，同时发挥人工的优势，促进人机融合翻译模式的发展。

5 总结

我们构建了一个可公开获得的细粒度英汉机器翻译错误分析语料库ErrAC，语料库中每一个标注样本包括源语言句子、机器译文、参考译文、PE译文、词错误率，以及基于PE译文所进行的错误标注。错误分析是机器翻译质量评价的重要内容，错误分析语料库可以准确、有效地评价机器翻译质量，获得关于机器译文错误类型、错误分布的数据，有助于了解目前神经机器翻译存在的具体问题，为机器翻译系统开发提供参考，明确其改进的方向。我们将译后编辑

与错误分析结合起来，对所进行的译后编辑操作进行错误标注，这比使用参考译文作为参照进行错误标注，更能准确地反应机器译文的具体问题，更符合人对机器译文错误的认知。错误分析对机器翻译系统的开发和译后编辑工作都有很好的参考作用，还可以为机器翻译质量估计、错误预测、自动译后编辑和译后编辑教学提供数据基础和参考作用。由于人工的限制，目前数据库规模还比较有限，而且只针对神经机器翻译做了错误分析，没有涉及SMT等其他系统的错误分析和相互比较。未来的工作除扩大语料库规模，涵盖更多领域和不同机器翻译系统的语料，还将基于该语料库构建初步的计算模型，用于机器翻译质量估计和自动译后编辑实验。此外，本文未涉及错误类型与自动评价指标、译后编辑工作量之间相关性的考察，未来将继续这方面的研究。

参考文献

Eleftherios Avramidis, Aljoscha Burchardt, Sabine Hunsicker, Maja Popovic, Cindy Tscherwinka, David Vilar, and Hans Uszkoreit. 2014. The taraxü corpus of human-annotated machine translations. In *LREC*, pages 2679–2682.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Loïc Barrault, Ondřej Bojar, Marta R Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, et al. 2019. Findings of the 2019 conference on machine translation (wmt19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61.

Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2016. Neural versus phrase-based machine translation quality: a case study. *arXiv preprint arXiv:1608.04631*.

Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2018. Neural versus phrase-based mt quality: An in-depth analysis on english–german and english–french. *Computer Speech & Language*, 49:52–70.

Ondřej Bojar. 2011. Analyzing error types in english-czech machine translation. *The Prague Bulletin of Mathematical Linguistics*, 95(1):63–76.

Marcello Federico, Matteo Negri, Luisa Bentivogli, and Marco Turchi. 2014. Assessing the impact of translation errors on machine translation quality with mixed-effects models. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1643–1653.

Mark Fishel, Ondřej Bojar, and Maja Popovic. 2012. Terra: a collection of translation error-annotated corpora. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*, pages 7–14.

Katrin Kirchhoff, Daniel Capurro, and Anne Turner. 2012. Evaluating user preferences in machine translation using conjoint analysis. In *EAMT 2012: Proceedings of the 16th Annual Conference of the European Association for Machine Translation, Trento, Italy*, pages 119–126.

Filip Klubička, Antonio Toral, and Víctor M Sánchez-Cartagena. 2017. Fine-grained human evaluation of neural versus phrase-based machine translation. *The Prague Bulletin of Mathematical Linguistics*, 108(1):121–132.

Maarit Koponen. 2010. Assessing machine translation quality with error analysis. In *Electronic proceeding of the KaTu symposium on translation and interpreting studies*.

Maarit Koponen. 2012. Comparing human perceptions of post-editing effort with post-editing operations. In *Proceedings of the seventh workshop on statistical machine translation*, pages 181–190. Association for Computational Linguistics.

Hans P Krings. 2001. *Repairing texts: empirical investigations of machine translation post-editing processes*, volume 5. Kent State University Press.

Arle Lommel, Aljoscha Burchardt, Maja Popovic, Kim Harris, Eleftherios Avramidis, and Hans Uszkoreit. 2014. Using a new analytic measure for the annotation and analysis of mt errors on real data. In *Proceedings of the 17th Annual Conference of the European Association for Machine Translation*, pages 165–172.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Maja Popović and Mihael Arcan. 2016. Pe2rr corpus: manual error annotation of automatically pre-annotated mt post-edits. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 27–32.

Maja Popović, Hermann Ney, Adrià De Gispert, José B Mariño, Deepa Gupta, Marcello Federico, Patrik Lambert, and Rafael Banchs. 2006. Morpho-syntactic information for automatic error analysis of statistical machine translation output. In *Proceedings of the workshop on statistical machine translation*, pages 1–6. Association for Computational Linguistics.

Maja Popovic, Eleftherios Avramidis, Aljoscha Burchardt, Sabine Hunsicker, Sven Schmeier, Cindy Tscherwinka, David Vilar, and Hans Uszkoreit. 2013. Learning from human judgments of machine translation output. In *Proceedings of the XIV Machine Translation Summit*, pages 231–238.

Maja Popovic, Arle Lommel, Aljoscha Burchardt, Eleftherios Avramidis, and Hans Uszkoreit. 2014. Relations between different types of post-editing operations, cognitive effort and temporal effort. In *Proceedings of the 17th annual conference of the european association for machine translation*, pages 191–198. European Association for Machine Translation Dubrovnik, Croatia.

Maja Popović. 2011. Hjerson: An open source tool for automatic error classification of machine translation output. *The Prague Bulletin of Mathematical Linguistics*, 96(1):59–67.

Marion Potet, Emmanuelle Esperança-Rodier, Laurent Besacier, and Hervé Blanchon. 2012. Collection of a large database of french-english smt output corrections. In *LREC*, pages 4043–4048.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of association for machine translation in the Americas*, volume 200, pages 223–231.

Sara Stymne and Lars Ahrenberg. 2012. On the practice of error analysis for machine translation evaluation. In *LREC*, pages 1785–1790.

Antonio Toral and Víctor M Sánchez-Cartagena. 2017. A multifaceted evaluation of neural versus phrase-based machine translation for 9 language directions. *arXiv preprint arXiv:1701.02901*.

David Vilar, Jia Xu, D'Haro Luis Fernando, and Hermann Ney. 2006. Error analysis of statistical machine translation output. In *LREC*, pages 697–702.

Guillaume Wisniewski, Anil Kumar Singh, Natalia Segal, and François Yvon. 2013. Design and analysis of a large corpus of post-edited translations: quality estimation, failure analysis and the variability of post-edition. In *Machine Translation Summit*, volume 14, pages 117–124.

Anna Zaretskaya, Mihaela Vela, Gloria Corpas Pastor, and Miriam Seghiri. 2016. Measuring post-editing time and effort for different types of machine translation errors. *New Voices in Translation Studies*, (15):63–92.

Daniel Zeman, Mark Fishel, Jan Berka, and Ondřej Bojar. 2011. Addicter: what is wrong with my translations? *The Prague Bulletin of Mathematical Linguistics*, 96(1):79–88.

孙逸群. 2019. 海洋类论文摘要机辅翻译错误剖析. 中国科技翻译, 32(2):31–33.

罗季美and 李梅. 2012. 机器翻译译文错误分析. 中国翻译, (5):84–89.

罗季美. 2014. 机器翻译句法错误分析. 同济大学学报: 社会科学版, 25(1):111–118.

层次化结构全局上下文增强的篇章级神经机器翻译

陈林卿，李军辉*，贡正仙†

（苏州大学 自然语言处理实验室，江苏 苏州 215006）

摘要

如何有效利用篇章上下文信息一直是篇章级神经机器翻译研究领域的一大挑战。本文提出利用来源于整个篇章的层次化全局上下文提高篇章级神经机器翻译性能。为了实现该目标，本文模型分别获取当前句内单词与篇章内所有句子及单词之间的依赖关系，结合不同层次的依赖关系以获取含有层次化篇章信息的全局上下文。最终源语言当前句子中的每个单词都能获取其独有的综合词和句级别依赖关系的上下文。为了充分利用平行句对语料在训练中的优势本文使用两步训练法，在句子级语料训练模型的基础上使用含有篇章信息的语料进行二次训练以获得捕获全局上下文的能力。在若干基准语料数据集上的实验表明本文提出的模型与若干强基准模型相比取得了有意义的翻译质量提升。实验进一步表明，结合层次化篇章信息的上下文比仅使用词级别上下文更具优势。除此之外，本文尝试通过不同方式将全局上下文与翻译模型结合并观察其对模型性能的影响，并初步探究篇章翻译中全局上下文在篇章中的分布情况。

关键词： 神经机器翻译；篇章上下文

Hierarchical Global Context Augmented Document-level Neural Machine Translation

CHEN Linqing, LI Junhui, GONG Zhengxian
(Natural Language Processing Laboratory, Soochow University, Suzhou, Jiangsu 215006)

Abstract

How to effectively use textual context information is always a challenge in the field of document-level neural machine translation. This paper proposes to use the hierarchical global context generated from the entire document to improve the performance of document-level neural machine translation models. In order to achieve this goal, this model obtains the dependencies between the current words in the sentence and all of the sentences and words in the document respectively, and combines the dependencies of different levels to obtain the global context containing the hierarchical contextual information, which can be use to guide translating the current sentence. Each word in the current sentence of the source language gets its own context that combines word and sentence level dependencies. In order to make full use of

基金项目：国家自然科学基金(61876120)
基金项目：国家自然科学基金(61976148)

the advantages of the parallel sentence-level corpus in training, the two-step training method is used in this paper. Based on the Transformer ,which is trained on a sentence-level corpus, the corpus containing textual information is used for secondary training to help the model gain the ability to capture and understand global context. Experiments on several benchmark corpus data sets show that the proposed model can significantly improve translation quality compared with other strong baseline models. The experiment further shows that combining hierarchical contextual information is more advantageous than word level context. In addition, this paper attempts to combine the global context with the translation model in different ways and observe its influence on the performance of the model, and studies the distribution of the global context in document-level translations.

Keywords: Neural Machine Translation , Document-level Context

1 引言

神经机器翻译近两年不断取得鼓舞人心的进展，已经成为当前机器翻译最受关注的研究领域之一。在过去几年中研究者们通过一系统模型不断提高神经机器翻译的性能(Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017; Gehring et al., 2017)。机器翻译和人工翻译之间的质量差距被这些出色的工作不断缩小。其中Transformer模型Vaswani et al., (2017)凭借多头注意力机制在句子级神经翻译任务中达到了当前最好成绩。然而Transformer在篇章级神经机器翻译任务中的表现却差强人意，主要原因在于其忽略了篇章句子间的依赖关系也没能有效利用篇章上下文。

研究者们提出各种获取上下文的方法改善前文所述问题，(Maruf and Haffari., 2018; Wang et al., 2017; Zhang et al., 2018)通过提取前文语句帮助模型翻译当前语句，此类方法没有充分建模当前语句前后上下文有较大差异的情况。当前句若只利用前侧上下文，可能由于有效信息不完整造成负面影响甚至是当前句翻译错误。同时，当前句之后语句的翻译也可能受前句的语义偏差影响造成错误累积。Miculicich et al., (2018)提出了利用全文获取上下文的方法，但仍将注意力聚焦在篇章的一定范围内。Tan et al., (2019)等人则提出了新的方法将整个篇章作为上下文来源，通过层次化网络利用句向量之间的注意力机制获取上下文向量，并将其分配给当前句中的词以帮助模型提高篇章翻译质量。完全依赖句向量将全局上下文与当前句间接结合的方法在信息高度压缩的过程中可能造成有效信息损失，也没有直接获取并利用当前句中的词与全文中其他句子或单词的依赖关系。

不同于以上方法，本文提出利用具有层次化结构信息的全局上下文提高神经机器翻译模型的性能。如图1所示在本文提出的模型中，我们通过不同注意力层分别提取来自不同层次的上下文依赖关系并将二者结合，使得获取的上下文包含多层次篇章信息。为了尽可能多获取上下文，该模型一次性从篇章全文获取前述层次化上下文而不是只使用当前语句之前的语句。由于篇章信息获取过程基于当前句中的所有单词分别计算，使得每一个词都能差异化获取来自整个篇章的有效上下文。受(Zhang et al., 2018; Miculicich et al., 2018)启发，本文使用两步训练法进行训练，从而高效利用含有篇章信息的语料。

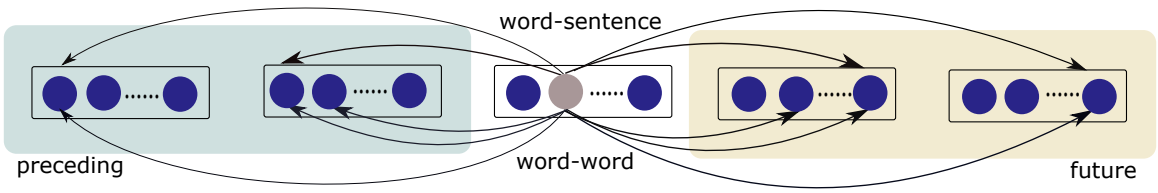


图 1: 计算当前句中的词与篇章中所有句/词的依赖关系

2 层次结构全局上下文增强的翻译模型

本文研究目标是将综合包含句子级依赖关系及单词级依赖关系的全局上下文结合进翻译模型，从而提高模型的篇章翻译质量。为了实现这个目标，我们首先利用源端编码器自注意力层

输出的词级隐藏状态获取句子向量。然后基于词级隐藏状态及句子向量分别计算当前句子中每个词与篇章中所有词及所有句子之间的依赖关系。最终，我们通过综合了两种依赖关系的权重获取具有层次化篇章结构信息的全局上下文，并利用这些上下文辅助模型进行篇章翻译。为了便于理解本文做出如下定义：含有N个语句的文档表示为：$\mathcal{X} = (X_1, \cdots, X_N)$，篇章中的句子$X_i = (x_{i,1}, \cdots, x_{i,n})$ 含有n 个词, 本文使用d_m 表示隐藏状态及词嵌入的维度。

2.1 词-句级依赖权重

为了避免之前的研究工作中仅使用当前语句前面的句子作为上下文对翻译质量造成的负面影响及错误累积。本文将篇章中的所有语句作为上下文来源。如图2(a)所示，词-句级依赖权重生成模块自下而上由编码器自注意力层，句向量嵌入层及词-句权重生成层组成。该模块将源端编码器自注意力层输出的隐藏状态以句子为单位嵌入为句向量，再通过当前句中词与全文句向量之间的注意力函数获取词-句级别的权重。

编码器自注意力层： 本文使用多头注意力函数Vaswani et al., (2017)捕获同一句子中单词间的依赖关系。篇章中的每个句子都会以词为单位被编码器编码，从而获取源端语句的词级隐藏状态:

$$S_i^{(k)} = \text{MultiHead}\left(A_i^{(k)}, A_i^{(k)}, A_i^{(k)}\right), \tag{1}$$

MultiHead表示多头注意力函数，通过将输入映射到不同子空间对输入序列之间的依赖关系进行建模。其中输出$S_i^{(k)}$ 的维度为$\mathbb{R}^{n \times d_m}$. 对于编码器的第一层来说, $A_i^{(1)} = X_i$ 而对于编码器的其他层而言$A_i^{(k)}$ 是上一层编码器的输出$A_i^{(k-1)}$。如图2所示，本文将这部分参数作为上下文生成器的共享参数。

在生成句向量之前本文使用残差网络和层标准化对编码器自注意力层的输出进行规整，得到的实际输出如下:

$$S_i^{(k)} = \text{LayerNorm}\left(S_i^{(k)} + A_i^{(k)}\right). \tag{2}$$

其中，LayerNorm是层规范化函数。出于保持模型结构图的简洁，本文在后续插图中省略了每个注意力层后的残差连接和层标准化。

句向量嵌入层： 受Lin et al., (2017)的启发，本文使用一个线性结合层获取句子向量。该层通过注意力机制将整个句子中所有单词产生的隐藏状态结合在一起从而生成句子向量。句中单词映射为句子向量的权重计算方法如下:

$$\alpha = \text{softmax}\left(W^2 \tanh\left(W^1\left(S_i^{(k)}\right)^T\right)\right), \tag{3}$$

其中$W^1 \in \mathbb{R}^{d_m \times d_m}$ ，$W^2 \in \mathbb{R}^{d_m}$是模型的参数矩阵。使用前文所述编码器自注意力层输出的词级隐藏状态及计算出的映射权重获得篇章中的句子向量:

$$v_{X_i}^{(k)} = \sum_{j=1}^{n} \alpha_{i,j} s_{i,j}^{(k)}. \tag{4}$$

其中$v_{X_i}^{(k)}$表示篇章中句子X_i经过句向量嵌入层后生成的句向量，$\alpha_{i,j}$表示句子X_i中各单词映射为句向量的权重，$s_{i,j}^{(k)}$表示句子X_i中的词经过编码器自注意力层输出的隐藏状态。

权重计算： 利用句向量嵌入层的输出计算当前句中的词与篇章中所有句子间的依赖关系，公式如下:

$$u_{i,j}^{(k)} = \text{softmax}\left(s_{i,j}^{(k)} V^{(k)} / \sqrt{d_{V^{(k)}}}\right), \tag{5}$$

其中$u_{i,j}^{(k)} \in \mathbb{R}^{1 \times N}$表示句子$X_i$中单词$x_{i,j}$与篇章中所有句子的依赖权重。$V^{(k)} = (v_{X_1}^{(k)}, \cdots, v_{X_N}^{(k)})$表示一个篇章$\mathcal{X}$中所有句子向量的集合。

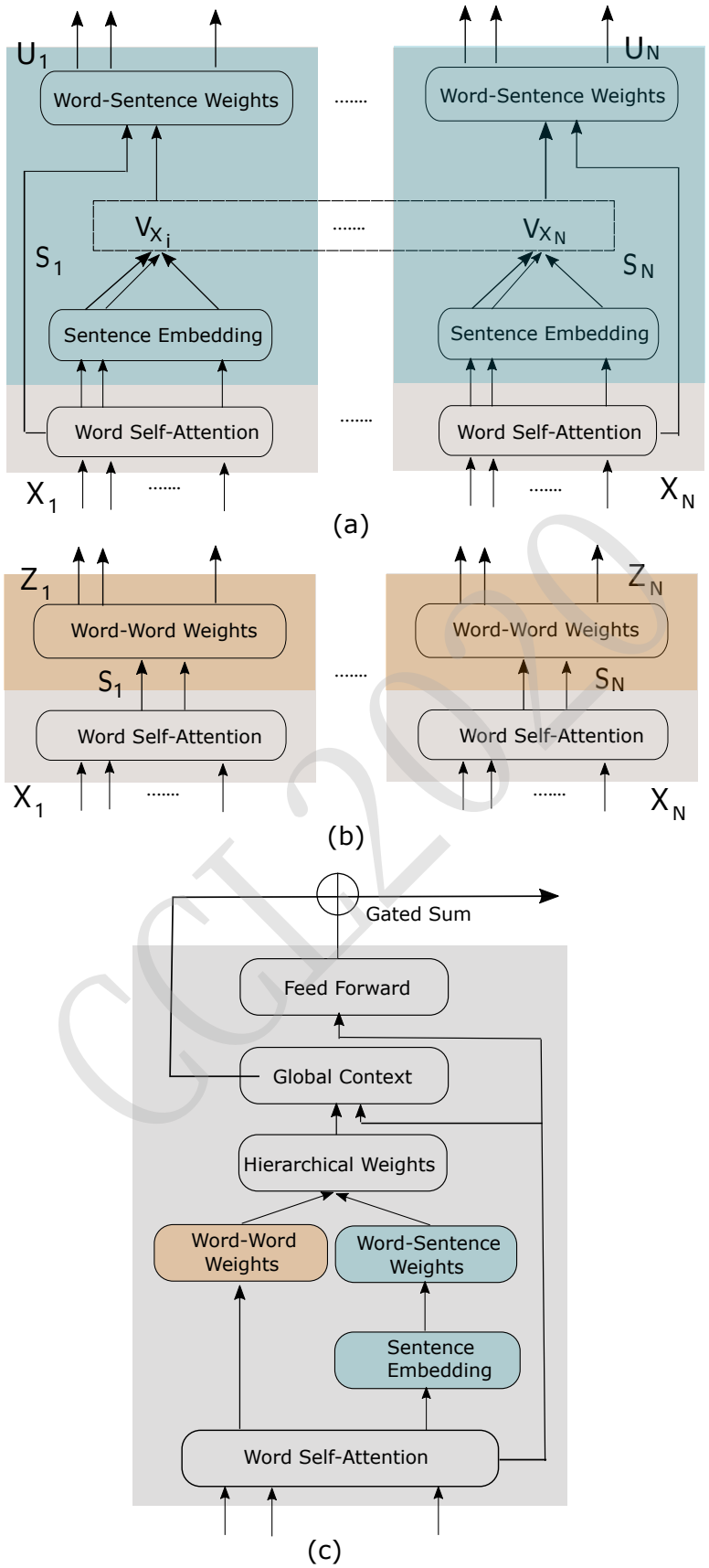


图 2: (a): 词-句级依赖权重的获取过程; (b): 词-词级依赖权重的获取过程。(c): 通过结合不同层次依赖关系获取全局上下文的过程。

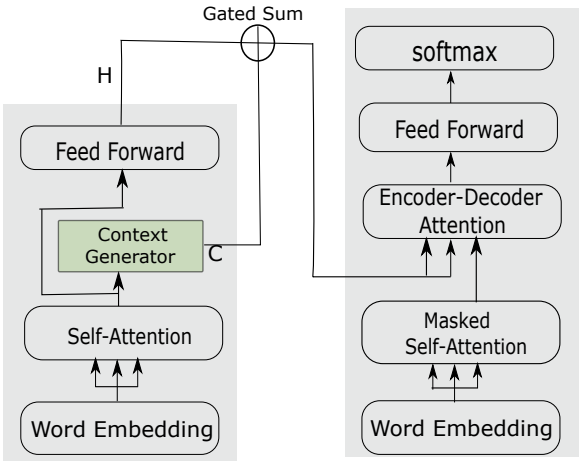


图 3: 层次化结构上下文与翻译模型的结合

2.2 词-词级依赖权重

如图2(b)所示，利用源端编码器自注意力层输出的隐藏状态，为当前句的每一个单词获取其与篇章中所有单词的依赖关系。计算公式如下：

$$z_{i,j}^{(k)} = \text{softmax}\left(s_{i,j}^{(k)}, S^{(k)}/\sqrt{d_{S^{(k)}}}\right), \tag{6}$$

其中，$M = N \times n$即篇章中所有词的个数,$z_{i,j}^{(k)} \in \mathbb{R}^{1\times M}$视为句子$X_i$中单个单词与全文单词的依赖关系权重向量。

2.3 层次化全局上下文的获取

本文所述模型使用源端篇章作为全局上下文，没有使用额外的上下文语料。为了减少计算开销，避免模型参数增加过多，我们将两个层次依赖权重的计算过程及全局上下文获取过程建立在编码器中，并共享编码器中的自注意力层参数。

如图2(c)所示，本文使用通过词-句级权重修正词-词级权重，使得最终获得的权重矩阵既含有句子层面的依赖关系，又含有整个篇章中每个单词之间的依赖关系。公式如下：

$$Q_{combine}^{i} = U_i Z_i. \tag{7}$$

其中。$U_i^{(k)} \in \mathbb{R}^{n\times N}$ 视为表示句子X_i中所有单词与篇章中其他句子之间依赖关系的权重向量，$Z_i^{(k)} \in \mathbb{R}^{n\times M}$视为篇章中句子$X_i$所有单词各自与篇章所有单词的依赖关系权重矩阵。

利用蕴含两层依赖关系的权重矩阵，将编码器自注意力层输出以篇章为单位的隐藏状态分配给当前句的每个单词。至此，当前句的每个单词都各自获取特有的蕴含自上而下不同层面依赖关系的全局上下文。

$$C_i^{hier} = Q_{combine}^{i} S^{(k)}. \tag{8}$$

其中，$C_i^{hier} \in \mathbb{R}^{n\times d_m}$即语句$X_i$获取的全局上下文。

2.4 层次化全局上下文的结合

编码器自注意力层输出的隐藏状态通过前馈全连接网络后得到翻译模型编码器输出，其表达形式如下：

$$H_i^{(k)} = \text{FNN}(S_i^{(k)}). \tag{9}$$

最终,层次化全局上下文与编码器输出通过门控单元结合。

$$H_i^{(k)} = \lambda H_i^{(k)} + (1-\lambda)\, C_i^{hier(k)}. \tag{10}$$

Set	ZH-EN		ES-EN		EN-DE	
	#SubDoc	#Sent	#SubDoc	#Sent	#SubDoc	#Sent
Training	47,758	781,524	6,531	180,853	7,491	206,126
Dev	82	1,664	33	887	326	8,967
Test	627	5,833	165	4,706	87	2,271

表 1: 训练集，开发集及测试集的统计信息

门控单元系数的计算方法如下：

$$\lambda = \text{sigmoid}\left([H_i^{(k)}; C_i^{hier(k)}]W^G\right), \tag{11}$$

其中$H_i^{(k)} \in \mathbb{R}^{n\times d_m}$是编码器经过全连接前馈神经网络层后的输出。$W^G \in \mathbb{R}^{2d_m\times d_m}$ 是模型参数矩阵。如图3所示，层次化全局上下文与编码器输出结合后进入解码器。

3 实验

本文将仅通过词-词级依赖权重获取的全局上下文称为词级上下文。将使用被词-句级权重规整过的复合权重获取的含有层次化篇章信息的全局上下文称为复合上下文。本文分别选择限定上下文获取范围及结构化上下文两类上下文结合方式中性能较好的强基线模型作为对比模型。

3.1 数据集

在中-英实验中，篇章级平行语料的训练集包括4.7万个文档中的78万个句子对[0]。我们使用NIST MT 2006数据集作为开发集，并使用MT 2002、2003、2004、2005、2008数据集作为测试集，其中测试集的合集标记为All。本文使用Jieba[1]分词将汉语句子按词切分，而英语句子则使用Moses 脚本Koehn et al., (2007)进行分词和小写处理。我们通过BPE Sennrich et al., (2016)使用3万大小的词表分别将源语言和目标语言中的单词进一步分割成子词

西班牙-英翻译任务中的训练集为IWSLT 2014和2015 Cettolo et al., (2012)，开发集为dev2010，测试集为tst2010、tst2011和tst2012。英-德翻译任务中的训练集来自IWSLT2017，本文使用tst2016和tst2017作为测试集，余下数据集作为开发集。所有数据集均使用Moses脚本进行分词和Truecasing处理。并使用3万大小的联合词表将源端及目标端语料中的单词分割成子词。由于本文词级别上下文需要计算篇章中所有词之间的依赖关系，计算开销及显存占用都十分可观。考虑到训练效率，我们将长篇章切分为最大长度为30句的段落。实验数据集的篇章数，句子数及平均篇章长度等统计信息如表1所示。

3.2 实验设置

本文基于OpneNMT[2] Klein et al., (2017)实现以平行句对为单位更新参数的基准模型Transformer，并进一步拓展为以篇章为单位更新参数的翻译模型。以篇章为单位更新使得模型可以轻易获取语料的篇章信息，从而进一步获取全局上下文。本文将模型隐藏状态的维度设为512，每个编码器解码器的层数都设置为6，多头注意力机制中的个数都设置为8，柱状搜索的大小设置为5，dropout设置为0.1。在训练过程中，我们将批大小设置为8192个字符并使用$\beta_1 = 0.1$的Adam优化器对模型进行优化Kingma and Ba., (2015)。

3.3 训练方式

受Zhang et al., (2018)启发，本文使用两步训练策略充分利用句子级平行语料在训练速度及计算开销等方面的优势。在第一步训练中使用平行句对语料对句子级别参数进行训练,在第二步训练中使用含有篇章信息的语料训练篇章级参数,该部分参数包括不同层次依赖权重的获取，含有分层结构信息全局上下文的获取，及结合上下文与编码器输出的门控等。两步训练使用的平

[0]训练集由LDC2002T01, LDC2004T07, LDC2005T06, LDC2005T10, LDC2009T02, LDC2009T15, LDC2010T03组成。

[1]https://github.com/fxsjy/jieba

[2]https://github.com/OpenNMT/OpenNMT-py

模型	MT06	MT02	MT03	MT04	MT05	MT08	All
Transformer	36.27	42.71	43.51	41.25	41.07	31.54	39.64
+ 词级上下文	37.05‡	43.79‡	44.57‡	41.98†	42.10‡	32.49‡	40.61‡
+ 复合上下文	**37.46**‡	**44.08**‡	**44.86**‡	**42.87**‡	**42.16**‡	**32.74**‡	**41.10**‡
Transformer(Zhang et al., 2018)	36.20	42.41	43.12	41.02	40.93	31.49	39.53
Transformer-DocNMT(Zhang et al., 2018)	37.12	43.29	43.70	41.42	41.84	32.36	40.22

表 2: 本文模型中-英翻译任务的性能(BLEU). † 和‡表示与Transformer基准模型相比显著性p值小于0.05/0.01

模型	西-英		英-德	
	BLEU	Meteor	BLEU	Meteor
Transformer	35.50	34.60	23.02	43.66
+ 词级上下文	37.59	36.50	24.40	45.19
+ 复合上下文	**37.75**	**36.83**	**24.98**	**45.70**
Transformer-DocNMT(Zhang et al., 2018)	37.07	36.16	24.00	44.69
HAN-DocNMT(Miculicich et al., 2018)	37.35	36.50	24.58	45.48

表 3: 本文模型在西班牙语-英语及英语-德语任务上的翻译性能(BLEU 和Meteor)

行句对语料与篇章语料是同一数据集的不同切分方式，没有引入额外语料。本文实验使用单块显存32G的Nvidia V100显卡进行训练。

3.4 评估指标

对于中-英翻译任务，本文报告了使用multi-bleu.perl脚本计算的不区分大小写的BLEU得分Papineni et al ., (2002)。对于其他翻译任务，本文报告了根据multi-bleu.perl脚本计算的区分大小写的BLEU分值和Meteor得分Lavie and Agarwal., (2007)。以上数据集和评估方法与本文比较实验的设置是一致的。我们使用paired bootstrap重采样方法评测BLEU值提升的显著性Koehn et al., (2004)。

3.5 实验结果

表2列出了汉-英翻译的性能结果。实验不但表明词级或复合上下文都能显著提高翻译性能，而且表明使用含有复合篇章信息的上下文比单独使用词级全局上下文效果更好。例如，在单一使用词级全局上下文实验中，本文方法在All测试集上的BLEU分数相比基准模型Transformer提高了0.95，在结合使用复合上下文后本文在All测试集上取得了1.36的提升。与Zhang et al., (2018)对前两句话进行建模的方法相比，在相似基准模型的前提下，本文方法(词级上下文及复合结构上下文)都取得了明显的性能提升，这表明全局上下文比当前句前两句更有助于提高文档级神经机器翻译质量。

表3列出了本文模型在西班牙-英及英-德两个篇章级翻译任务上的BLEU和Meteor得分。与中-英任务相似的是，在这两个翻译任务中利用全局上下文比只选用篇章中部分语句作为上下文更有帮助。此外，将不同层次的篇章信息结合进上下文会给翻译质量带来进一步提升。在两个翻译任务上，我们的方法相比Transformer基准模型在BLEU (Meteor)评测标准上提高了2.25(2.23)和1.96(2.04)。

4 分析与讨论

4.1 模型参数及训练时间

如表4统计数据所示，本文提出的上下文获取及结合方式由于编码器多头注意力层参数共

模型	参数（百万）
Transformer	51.3
+ 复合上下文	57.6
HAN-DocNMT(Miculicich et al., 2018)	63.0
Transformer-DocNMT(Zhang et al., 2018)	96.8

表 4: 不同模型参数比较.

享，参数增加数量较少。综合考虑了模型性能与参数及计算开销之间的平衡。同时，本文充分利用句子级平行语料在训练时间上的优势，通过两步训练法使得训练时间相比其他模型没有明显增加。

4.2 不同上下文利用方式

如图4所示，为了观察不同方式利用上下文对翻译质量的影响，我们尝试在解码器端增加专门针对全局上下文的注意力层结构。实验结果对比如表5，相比本文使用的直接融合不同层次依赖关系的方法，增加注意力层的方法增加了模型参数和计算开销，在翻译性能方面没有取得有意义的提升。

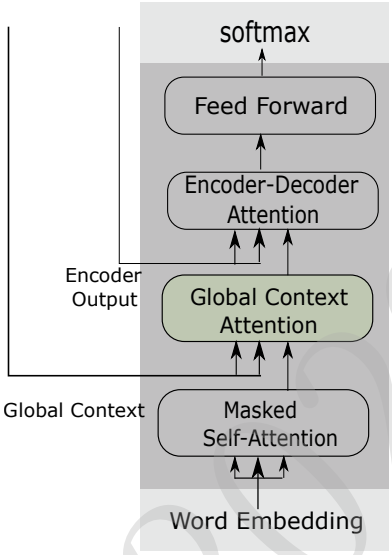


图 4: 在解码器中增加全局上下文注意力层作为上下文结合方式

结合方式	BLEU
直接融合	**41.10**
增加注意力层	41.09

表 5: 不同上下文结合方式翻译性能对比.

上下文来源	BLEU
当前句前文	40.31
当前句后文	**40.49**
无上下文	39.64

表 6: 不同来源上下文对翻译性能影响.

4.3 上下文分布

高清数字电视在美渐成主流
新华社纽约2 月12 日电(记者范小林) 随着高分辨率数字电视机(HDTV) 的售价……
刚落幕的美国“ 超级碗” 美式橄榄球赛已经成为美国广播业者迎接电视高清……。
据《金融时报》日前报道, 美国国会最近通过一项法案, 规定美国广播业者必须在……。
这意味着美国电视节目播放的全面数字化已经有了明确的时间表。
高分辨率数字电视的普及一方面要靠消费者购买电视机, 一方面……。
数字电视虽然在过去几年被一再提及, 但始终都未成为现实。
这个问题在今年出现转机。
根据美国消费电子协会的预测, 数字电视机今年在美销售量将首次超过传统电视机, ……。

表 7: 开发集篇章上下文分布样例

本文针对文中使用的全局上下文进行前后文屏蔽实验以观察当前句前文及后文各自对翻译的影响。实验结果如表6所示,二者无显著差异，屏蔽前文的翻译性能略低于屏蔽后文，这与Wong et al., (2020)的研究相符，我们推测前后文的重要性可能随语种和语料类型发生变化。本文认为该实验虽然不能直接得出当前句前后文重要性孰轻孰重的结论，但可以表明后文作为上下文对篇章翻译的重要性。

在4.2的实验中，本文使用来自整个篇章的全局上下文与翻译模型结合，其翻译质量比仅使用前文作为上下文取得了显著提升。出于探索前后上下文对篇章翻译质量影响的目的，我们对全局上下文的分布展开如下分析与实验。本文利用句子级依赖权重对开发集中的篇章句子进行统计，获取篇章中对所有句子而言都最重要的句子，并将该句使用加粗字体表示。表7中的样例可以直观表明，当前句的上下文不一定只存在于邻近语句。该现象不仅存在于本文所举样例，也存在于本文实验所使用的其他篇章语料中。

4.4 名词与代词翻译

为了观察本文提出的层次化结构全局上下文模型是如何提高翻译质量的，我们对代词和名词的翻译进行进一步的实验与分析。在代词翻译中，我们使用Miculicich et al., (2017)提出的APT度量标准评价中-英翻译实验代词翻译的准确性。如表8所示，结果表明本文提出的多层结构全局上下文模型能够更好地捕捉到每个词的全局上下文，从而提升中-英翻译实验在代词翻译的准确率。

Model	MT06	MT02	MT03	MT04	MT05	MT08	All
Transformer	69.54	73.67	68.41	65.32	67.71	71.60	68.68
+ 复合上下文	**70.24**	**74.22**	**69.02**	**65.45**	**68.29**	**71.91**	**69.40**

表 8: 代词翻译质量(APT)对比

源端	..... 今天晚上的十一二点钟左右吧。
参考翻译	..... **it** will arrive around 11 : 00 or 12 : 00 tonight .
Transformer	..... that about 11.2 pm today .
+ 词级上下文	..... **it** will be around 11.2 pm today .
+ 复合上下文	..... **it** will be around 12 : 00 tonight .

表 9: 代词翻译样例

本文在表9中列举了一个翻译例子进一步观察层次化全局上下文对代词翻译的帮助。通过实例可以看出本文提出的模型可以较好地推断出潜在代词，从而验证了该模型的代词翻译性能。对于名词翻译的分析，本文将展示另一个样例。

源端	.....一款非常优秀的基于PHP 和MySQL 数据库的社区**程序** 。
参考翻译	.....an excellent community **software** based on php and mysql database .
Transformer	.....an extremely outstanding community **procedure** based on the php..... .
+ 词级上下文	.....an extremely outstanding community **process** based on php a..... .
+ 复合上下文	.....an extremely outstanding community**programe** based on php a..... .

表 10: 名词翻译样例

表10的样例可以观察到本文提出的全局上下文比其他对比模型更好的翻译了易混淆的名词。同时不难看出相比仅使用单词级别上下文，使用层次化全局上下文对提升名词翻译质量的效果更好。

5 相关工作

(Gong et al., 2011; Hardmeier et al., 2012; Xiong et al., 2013; Tu et al., 2014; Garcia et al., 2015)在使用篇章信息提高统计翻译质量的研究领域做了大量工作。机器翻译研究热点从统计翻译转向神经翻译后不久，篇章级神经机器翻译的研究也蓬勃发展起来。根据获取上下文的范围，我们将相关研究分为两类:(1)使用部分语句作为上下文的研究;(2)使用篇章作为上下文的研究。

在第一类研究中，Tiedemann and Scherrer., (2017)基于循环神经网络(RNN)直接拼接语句作为上下文。随后(Jean et al., 2017; Wang et al., 2017; Zhang et al., 2018; Bawden et al., 2018; voita et al., 2019)的研究中，以RNNSearch和Transformer为基础使用具有不同注意力机制的多编码器提升篇章翻译质量。Miculicich et al., (2018)提出一种分层注意网络(HAN)，它通过句词的抽象表示

为当前句从前面的句子中提取上下文。Yang et al., (2019)在HAN的基础上提出一种胶囊网络将上下文信息按不同角度进行聚类。(Tu et al., 2018; Kuang et al., 2018)提出的基于缓存的方法所存储的是前面句子中的词/翻译，也归于这一类研究。

另一类研究以篇章为翻译单元，针对每个句子动态获取有用的篇章级信息。Maruf and Haffari., (2018)使用额外的存储网络将篇章转换为上下文与基于RNN的神经机器翻译模型结合。Mace and Servan., (2019)在每个源句中添加篇章标签，并将其替换为篇章级嵌入向量。Xiong et al., (2019)提出了一种二次优化策略，通过激励机制来完善第一轮翻译。Maruf et al., (2019)提出使用稀疏注意力机制选择性地捕获与当前句相关联的句子并进一步选择关键词。Tan et al., (2019)提出利用句向量之间的注意力机制获取上下文向量，并将其分配给当前句中的词。

与上述研究不同，本文提出从词-句层面和词-词层面对全局上下文进行建模从而获取复合依赖关系。使当前句的每一个词直接获取全文句子及单词中的潜在语义信息及递进关系。同时本文提出的上下文获取方式综合考量上下文范围及结合方式，既将上下文获取范围扩展至整个篇章，又没有增加额外语料或编码器。

6 总结

本文提出利用含有复合层次化篇章信息的全局上下文提升篇章级神经机器翻译质量。该模型首先为当前句中的词分别从词和句两个层面获取其篇章级依赖权重矩阵，然后通过复合权重及编码器自注意力层的输出获得全局上下文，最后将上下文与翻译模型结合。在多个基准语料数据集上的实验结果表明，与若干强基线系统相比该模型能带来显著的翻译质量提升。分析试验表明结合具有复合层次化篇章信息的全局上下文可以有助于提高篇章翻译的名词及代词翻译质量。

如何通过合理建模篇章语料中的长依赖关系获取其潜在的语义信息是一个值得不断探索的问题。我们将在未来的工作中继续对这一问题提出有意义的尝试。

参考文献

Dzmitry Bahdanau and Kyunghyun Cho and Yoshua Bengio. 2015. *Neural Machine Translation by Jointly Learning to Align and Translate*. Proceedings of ICLR.

Rachel Bawden and Rico Sennrich and Alexandra Birch and Barry Haddow. 2018. *Evaluating discourse phenomena in neural machine translation*. In Proceedings of NAACL, 1304–1313.

Matt Gardner and Joel Grus and Mark Neumann and Oyvind Tafjord and Pradeep Dasigi and Nelson Liu and Matthew Peters and Michael Schmitz and Luke Zettlemoyer. 2017. *AllenNLP: A Deep Semantic Natural Language Processing Platform*. In Proceedings of ACL Workshop for Natural Language Processing Open Source Software.

Zhengxian Gong and Min Zhang and Guodong Zhou. 2011. *Cache-based Document-level Statistical Machine Translation*. Proceedings of EMNLP, 909–919.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. *Convolutional sequence to sequence learning.*. Proceedings of the 34th International Conference on Machine Learning, 70:1243–1252.

Eva Martínez Garcia and Cristina España-Bonet and Lluís Màrquez. 2015. *Document-Level Machine Translation with Word Vector Models*. Proceedings of EAMT, 59–66.

Christian Hardmeier and Joakim Nivre and Jörg Tiedemann. 2012. *Document-Wide Decoding for Phrase-Based Statistical Machine Translation*. Proceedings of EMNLP-CoNLL, 1179–1190.

Hany Hassan and Anthony Aue and Chang Chen and Vishal Chowdhary and Jonathan Clark and Christian Federmann and Xuedong Huang and others. 2018. *Achieving Human Parity on Automatic Chinese to English News Translation*. Computing Research Repository, arXiv:1803.05567.

Sebastien Jean and Stanislas Lauly and Orhan Firat and Kyunghyun Cho. 2017. *Does neural machinetranslation benefit from larger context?*. In Computing Research Repository, arXiv:1704.05135.

Shaohui Kuang and Deyi Xiong and Weihua Luo and Guodong Zhou. 2018. *Modeling Coherence for Neural Machine Translation with Dynamic and Topic Caches*. Proceedings of COLING, 596—606.

Philipp Koehn. 2004. *Statistical significance tests for machine translation evaluation*. Proceedings of EMNLP, 388-–395.

Koehn, Philipp and Hoang, Hieu and Birch, Alexandra and Callison-Burch, Chris and Federico, Marcello and Bertoldi, Nicola and Cowan, Brooke and Shen, Wade and Moran, Christine and Zens, Richard and Dyer, Chris and Bojar, Ondřej and Constantin, Alexandra and Herbst, Evan. 2007. *Moses: Open Source Toolkit for Statistical Machine Translation*. Proceedings of ACL, (Jun):177–180.

Klein, Guillaume and Kim, Yoon and Deng, Yuntian and Senellart, Jean and Rush, Alexander. 2017. *OpenNMT: Open-Source Toolkit for Neural Machine Translation*. Proceedings of ACL, 67–72.

Diederik P. Kingma and Jimmy Ba. 2015. *Adam: A method for stochastic optimization*. Proceedings of ICLR.

Zhouhan Lin and Minwei Feng and Cicero Nogueira dos Santos and Mo Yu and Bing Xiang and Bowen Zhou and Yoshua Bengio. 2017. *A Structured Self-attentive Sentence Embedding*. Proceedings of ICLR.

Lavie, Alon and Agarwal, Abhaya. 2007. *METEOR: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments*. Proceedings of WMT, (Jun):228–231.

Valentin Mace and Christophe Servan. 2019. *Using whole document context in neural machine translation*. In Proceedings of IWSLT.

Sameen Maruf and Gholamreza Haffari. 2018. *Document Context Neural Machine Translation with Memory Networks*. Proceedings of ACL, 1275–1284.

Sameen Maruf and André F. T. Martins and Gholamreza Haffari. 2019. *Selective Attention for Context-aware Neural Machine Translation*. Proceedings of NAACL, 3092–3102.

Lesly Miculicich Werlen and Andrei Popescu-Belis. 2017. *Validation of an automatic metric for the ac- curacy of pronoun translation (APT)*. Proceedings of the Third Workshop on Discourse in Machine Translation, 17–25.

Lesly Miculicich and Dhananjay Ram and Nikolaos Pappas and James Henderson. 2018. *Document-Level Neural Machine Translation with Hierarchical Attention Networks*. Proceedings of EMNLP, 2947–2954.

Mauro Cettolo and Christian Girardi and Marcello Federico. 2012. *WIT3: Web Inventory of Transcribed and Translated Talks*. Proceedings of EAMT, 261–268.

Kishore Papineni and Salim Roukos and Ward Todd and Wei-Jing Zhu. 2002. *BLEU: a method for automatic evaluation of machine translation*. Proceedings of ACL, 311–318.

Rico Sennrich and Barry Haddow and Alexandra Birch. 2016. *Neural Machine Translation of Rare Words with Subword Units*. In Proceedings of ACL, 1715–1725.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. *Sequence to sequence learning with neural networks*. Advances in neural information processing systems, 3104-–3112.

Xin Tan and Longyin Zhang and Deyi Xiong and Guodong Zhou. 2019. *Hierarchical Modeling of Global Context for Document-Level Neural Machine Translation*. In Proceedings of EMNLP-IJCNLP, 1576–1585.

Mei Tu and Yu Zhou and Chengqing Zong. 2014. *Enhancing Grammatical Cohesion: Generating Transitional Expressions for SMT*. Proceedings of ACL, 850–860.

Tiedemann, Jörg and Scherrer, Yves. 2017. *Neural Machine Translation with Extended Context*. In Proceedings of the Third Workshop on Discourse in Machine Translation, ”82–92.

Zhaopeng Tu and Yang Liu and Shuming Shi and Tong Zhang. 2018. *Transactions of the Association for Computational Linguistics*. Transactions of the Association for Computational Linguistics, (6):407–420.

Ashish Vaswani and Noam Shazeer and Niki Parmar and Jakob Uszkoreit and Llion Jones and Aidan N.Gomez and Lukasz Kaiser and Illia Polosukhin. 2017. *Attention Is All You Need*. In Proceedings of NIPS, 5998–6008.

Elena Voita and Pavel Serdyukov and Rico Sennrich and Ivan Titov. 2018. *Context-Aware Neural Machine Translation Learns Anaphora Resolution*. Proceedings of ACL, 1264–1274.

Elena Voita and Rico Sennrich and Ivan Titov. 2019. *When a Good Translation is Wrong in Context: Context-Aware Machine Translation Improves on Deixis, Ellipsis, and Lexical Cohesion*. Proceedings of ACL, 1198–1212.

KayYen Wong and Sameen Maruf and Gholamreza Haffari. 2020. *Contextual Neural Machine Translation Improves Translation of Cataphoric Pronouns*. In Proceedings of ACL, 2826–2831.

Longyue Wang and Zhaopeng Tu and Andy Way and Qun Liu. 2017. *Exploiting cross-sentence context for neural machine translation*. In Proceedings of EMNLP, 2826–2831.

Lesly Miculicich Werlen and Andrei Popescu-Belis. 2017. *Validation of an Automatic Metric for the Accuracy of Pronoun Translation (APT)*. Proceedings of Workshop on Discourse in Machine Translation, 17–25.

Deyi Xiong and Yang Ding and Min Zhang and Chew Lim Tan. 2013. *Lexical Chain Based Cohesion Models for Document-Level Statistical Machine Translation*. Proceedings of EMNLP, 1563–1573.

Hao Xiong and Zhongjun He and Hua Wu and Haifeng Wang. 2019. *Modeling coherence for discourse neural machine translation*. In Proceedings of AAAI, 7338–7345.

Zhengxin Yang and Jinchao Zhang and Fandong Meng and Shuhao Gu and Yang Feng and Jie Zhou. 2019. *Enhancing Context Modeling with a Query-Guided Capsule Network for Document-level Translation*. Proceedings of EMNLP, 1527–1537.

Jiacheng Zhang and Huanbo Luan and Maosong Sun and Feifei Zhai and Jingfang Xu and Min Zhang and Yang Liu. 2018. *Improving the Transformer Translation Model with Document-Level Context*. Proceedings of EMNLP, 533–542.

基于多语言联合训练的汉-英-缅神经机器翻译方法

满志博[1,2]，**毛存礼**[*1,2]，**余正涛**[1,2]，**李训宇**[1,2]，**高盛祥**[1,2]，**朱俊国**[1,2]
1.昆明理工大学，信息工程与自动化学院，昆明，650500
2.昆明理工大学，云南省人工智能重点实验室，昆明，650500
270004294@qq.com, maocunli@163.com, ztyu@hotmail.com
1242041057@qq.com, gaoshengxiang.yn@foxmail.com, jgzhu@mtlab.hit.edu.cn

摘要

多语言神经机器翻译是解决低资源神经机器翻译的有效方法，现有方法通常依靠共享词表的方式解决英语、法语以及德语相似语言之间的多语言翻译问题。缅甸语属于一种典型的低资源语言，汉语、英语以及缅甸语之间的语言结构差异性较大，为了缓解由于差异性引起的共享词表大小受限制的问题，提出一种基于多语言联合训练的汉英缅神经机器翻译方法。在Transformer框架下将丰富的汉英平行语料与汉缅、英缅的语料进行联合训练，模型训练过程中分别在编码端和解码端将汉英缅映射在同一语义空间降低汉英缅语言结构差异性对共享词表的影响，通过共享汉英语料训练参数来弥补汉缅数据缺失的问题。实验表明在一对多、多对多的翻译场景下，提出方法相比基线模型的汉-英、英-缅以及汉-缅的BLEU值有明显的提升。

关键词： 汉语-英语-缅甸语；低资源语言；多语言神经机器翻译；联合训练；语义空间映射；共享参数

Chinese-English-Burmese Neural Machine Translation Method Based on Multilingual Joint Training

Zhibo Man[1,2], **Cunli Mao**[*1,2], **Zhengtao Yu**[1,2], **Xunyu Li**[1,2], **Shengxiang Gao**[1,2], **Junguo Zhu**[1,2]
1. Faculty of Information Engineering and Automation, Kunming University of Science and Technology Kunming 650500, China
2. Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and Technology Kunming 650500, China
270004294@qq.com, maocunli@163.com, ztyu@hotmail.com
1242041057@qq.com, gaoshengxiang.yn@foxmail.com, jgzhu@mtlab.hit.edu.cn

Abstract

Multilingual Neural Machine Translation is an effective method to solve low-resource Neural Machine Translation. Existing methods usually rely on a shared vocabulary to solve the problem of multilingual translation between similar languages in English, French, and German. Burmese language is a typical low-resource language. The language structure between Chinese, English and Burmese is quite different. In order to alleviate the problem of the limited size of the shared vocabulary caused by the difference, a multilingual combination is proposed. Trained Chinese-English-Burmese neural machine translation method. The rich Chinese-English parallel corpus and the Chinese-Burmese and English-Burmese corpus are jointly trained under the Transformer framework. During the model training process, the Chinese and English-Burmese maps are mapped to the same semantic space on the encoding side and the decoding side to

*毛存礼(通信作者):maocunli@163.com
国家自然科学基金重点项目(61732005)；国家自然科学基金(61662041，61761026, 61866019，61972186)；云南省自然科学基金重点项目(2019FA023)；云南省中青年学术和技术带头人后备人才项目(2019HB006)

reduce the difference in Chinese-English-Burmese language structure. The influence of sex on the shared vocabulary is to compensate for the lack of Chinese-Burmese data by sharing Chinese and English material training parameters. Experiments show that in one-to-many and many-to-many translation scenarios, the proposed method has a significant improvement over the baseline models of Chinese-English, English-Burmese, and Chinese-Burmese BLEU.

Keywords: Chinese-English-Burmese , Low Resource Language , Multilingual Neural Machine Translation , Joint training , Semantic space mapping , Shared parameters

1 引言

目前，多语言神经机器翻译（Multilingual Neural Machine Translation, MNMT）(Aharoni et al., 2019; Wang et al., 2018; Lee et al., 2017; Wang et al., 2019)在低资源机器翻译方面取得了较好的效果，与标准双语翻译的模型相比，其通过构建多种语言之间的联合训练(Caruana, 1997)模型，能够共享资源丰富语言的模型参数来提升资源稀缺语言机器翻译性能(Lignos et al., 2019)，但是，目前的方法主要利用在相似的语言之间，例如，英语、德语、法语等，这些语言之间有大量的“同源词”或者相同的子词词根，在进行多语言词汇共享词表时，这些语言的词表会有很多相同的词语可以得到共享。

基于统计的缅甸语机器翻译方法(Nwet et al., 2011a; Nwet et al., 2011b)依赖于大规模的汉缅平行语料，缅甸语是一种典型的资源稀缺型语言，多语言机器翻译可以有效解决汉缅数据缺乏的问题，但是，汉语、缅甸语以及英语三种语言的结构差异较大，没有相同的词语或者词根，共享词表时由于受到词表大小的限制会造成许多词语无法在词表中出现，导致三种语言的语义空间无法对齐，如图1，汉语“他”对应两个缅甸语单词，英语单词“very much”对应一个缅甸语单词，利用汉英缅数据进行多语言模型训练时，会出现无法准确的在编码过程中将三种语言的词语对齐的问题。

Figure 1: 汉语、英语、缅甸语互译句子示例

针对以上问题，提出的基于多语言联合训练的汉英缅神经机器翻译方法是在Johnson等人(2017)提出模型的基础上进行改进，在Transformer框架下，利用丰富的汉英平行语料与汉缅、英缅的语料进行联合训练，在编码、解码过程中，将汉语、英语以及缅甸语三种语言进行语义映射，缩小三种语言之间的语义距离，同时，丰富三种语言共享词表中的词语。在Tranformer的框架中，每个归一化和残差连接层中，将所有语言对的训练参数进行共享学习，提升汉缅神经机器翻译性能。

本文的贡献如下：

(1)通过在Transformer的编码-解码端的汉语、英语以及缅甸语三种语言词嵌入映射共享语义空间解决汉英缅由于差异性较大引起的词表共享受限制的问题。

(2)将汉英缅三种语言对的训练参数进行共享，利用汉英高资源语言对的训练参数弥补汉缅低资源语言对训练不充分的问题。

本文的第2节介绍了针对低资源语言以及缅甸语的机器翻译相关工作；第3节介绍了多语言神经机器翻译的研究背景；第4节描述了基于多语言联合训练的神经机器翻译方法；第5节通过

在一对多和多对一两种翻译场景下进行实验对比证明本文方法的优势；第6节对全文进行总结并指出进一步的研究工作。

2 相关工作

本文将相关工作的分为两类，分别是缅甸语的神经机器翻译以及低资源语言的神经机器翻译方法研究：

针对于缅甸语的机器翻译的研究工作：目前，由于缅甸语与其他语言的双语资源较少，针对于缅甸语的机器翻译的研究工作较少，Nwet等人(Nwet et al., 2011b)提出一种通过缅甸语-英语词对齐的英缅统计机器翻译方法，由于这种方法在一定程度上受到词表大小的限制，Nwet等人(Nwet et al., 2011a)进一步提出通过扩展英缅双语的平行语料的机器翻译方法。以上针对于缅甸语的机器翻译研究都是基于统计的方式，基于统计的方式需要大规模的双语词典或者是双语平行语料，缅甸语是一种典型的资源稀缺型语言，利用统计的方式不能完全适用于缅甸语。

缅甸语是一种资源稀缺型语言，解决低资源语言的神经机器翻译的方法主要包括：

(1)基于枢轴的神经机器翻译方法(Kim et al., 2019)：借助枢轴语言构建低资源的神经机器翻译模型，提高神经机器翻译性能。英语-缅甸语以及缅甸语-汉语本身的语料就缺乏并且翻译性能不佳，使用枢轴的方式对缅甸语机器翻译性能提高并不明显。

(2)基于迁移学习的神经机器翻译方法(Lakew et al., 2018; Dabre et al., 2019; Firat et al., 2016b; Sachan and Neubig, 2018)：借助预训练的思想，将资源丰富语言对训练模型或参数迁移到低资源语言对的训练过程中。缅甸语和其他语言之间的语法差异性极大，直接利用迁移学习的思想将模型迁移到缅甸语上的效果性能不佳。

(3)近些年来，利用多语言神经机器翻译联合训练逐渐成为解决低资源机器翻译的主流方法。例如，对所有源语言使用相同编码，目标语言使用不同的解码器。Dong等人(2015)在一对多的翻译场景下，提出将多语言翻译过程中的源语言的编码器共享，为每个目标语言分配不同的解码器的方法。Lee等人(2017)在多对一的翻译场景下，提出在编码器端采用字符级输入并将多种源语言共享编码器的方法。对所有源语言到目标语言的语言对使用不同的编码器、解码器，分别为每种语言对训练翻译模型。例如，Firat等人(2016a)提出一种基于共享注意力机制的多路、多对多的神经机器翻译方法。Zoph等人(2016)提出将注意力机制进行联合的多到一的神经机器翻译方法。以上的两类方法一方面需要每种语言专用的编码器或解码器，从而限制了翻译模型的泛化性。另一方面，训练不同的语言对的翻译模型在一定程度上也加大了模型训练的成本。对所有源语言、目标语言均使用相同的编码器、解码器。例如，Ha等人(2016)和Johnson等人(2017)训练了用于多语言翻译的单个NMT模型，使用目标语言符号作为翻译方向的指导。这种方法将不同的语言合并为一个联合表示空间，但是忽略了语言的多样性。Zhang等人(2019)提出了一种具有语言敏感机制的多语言神经机器翻译方法。在多语言神经机器翻译的训练过程中增加敏感机制的表示，达到共享的同时又不丢失语言本身的多样性。

由于汉英缅三种语言的差异性较大，不具备相同的词语或者词根。在多语言神经机器翻译框架下，仅仅利用共享词表的方法会在一定程度上限制三种语言的词汇表征能力，以上方法还不能完全应用于缅甸语的机器翻译问题。为此，本文在(Johnson et al., 2017; Wang et al., 2019)基础上，提出将汉英缅三种语言进行语义空间映射降低语言差异性，解决三种语言由于语言结构差异导致的词表受限制的问题，在公共语义空间中共享多语言联合训练模型参数，来提升缅汉机器翻译的性能。

3 多语言神经机器翻译研究背景

3.1 编码器-解码器

本文的主要工作是基于Transformer(Vaswani et al., 2017)的架构基础上进行多语言的神经机器翻译。在本文中选择Johnson等人(2017)以及Ha等人(2016)提出的方法作为实验中的基准模型。在多语言的神经机器翻译框架(Johnson et al., 2017)下，给定一句包含n个单词的源语言的句子为$\mathrm{x} = (x_1, x_2, ..., x_{|\mathrm{n}|})$，其包含目标语言的参考译文为$\mathrm{y} = (y_1, y_2, ..., y_{|\mathrm{m}|})$，以及对应的目标语言的标签为$T$，例如，2EN,2ZH以及2MY。在整个编码-解码的结构中主要包括：源语言、目标语言词嵌入、编码器、解码器和输出层。在词嵌入层每个语言和目标语言的多语言神经机器翻译的单词都会映射为一个向量矩阵W_E来表示。编码器-解码器(Sutskever et al., 2014)的表示

如下：

$$\mathrm{H_{enc}} = \mathrm{Encoder}([\mathrm{T}, \mathrm{x}]) \tag{1}$$

$$\mathrm{S_{dec}} = \mathrm{Decoder}([\mathrm{y}, \mathrm{H_{enc}}]) \tag{2}$$

其中，$\mathrm{H_{enc}} \in R^{|\mathrm{x}|\times d}$，$\mathrm{S_{dec}} \in R^{|\mathrm{y}|\times d}$分别表示编码器和解码器输出，$d$表示模型的尺寸。

与NMT模型相比，多语言模型一定程度上是将多个语言对进行联合训练，并将所有源语言和目标语言应用统一的神经机器翻译框架。针对统一的编码器-解码器的框架，在整个翻译的迭代过程中的目标函数:

$$L_{m-T}(D;\theta) = \sum_{l=1}^{L}\sum_{d=1}^{|D_l|}\sum_{T=1}^{M} \log P(y_t^l \left| x^l, y_{<t}^l; H_{enc}, S_{dec}, \theta_{attention}\right.) \tag{3}$$

其中，L表示联合训练语言的句子对个数，M表示目标语言句子的长度，$P(y_t^l \left| x^l, y_{<t}^l\right.)$表示第$l$个翻译对中第$d$个句子的第$t$个单词的翻译概率。$\theta_{attention}$表示训练过程中的注意力机制参数。

3.2 Transformer

Transformer是一种基于编码器-解码器的框架，由多个网络层堆叠而成。其中，编码器是6个相同的堆栈层组成，每个层包含一个自注意力机制层和基于词语位置的前馈子层，利用位置信息可以较好的将句子中每个词语的位置显式的加入到了神经网络中。解码器也遵循类似的结构。除了以上的自注意力机制层以及基于词语以及基于位置的前馈神经网络层以外，在解码器自注意力机制后是多头交叉的注意力机制网络。在编码器中，$e_{i,t}$是源语言向量表征与目标语言向量表征的相似度分数。

$$e_{i,t} = \frac{1}{\sqrt{d}} q_i k_t^T \tag{4}$$

$$q_i = W_q.H_{enc}^i, k_t = W_k.S_{dec}^t \tag{5}$$

在机器翻译中，Transformer将编码器中的隐状态视为一组键（Key）值（Vaule）对的集合。W_q和W_k表示的是交叉注意力机制的训练参数，d表示模型的尺寸。

4 基于多语言联合训练的汉-英-缅神经机器翻译模型

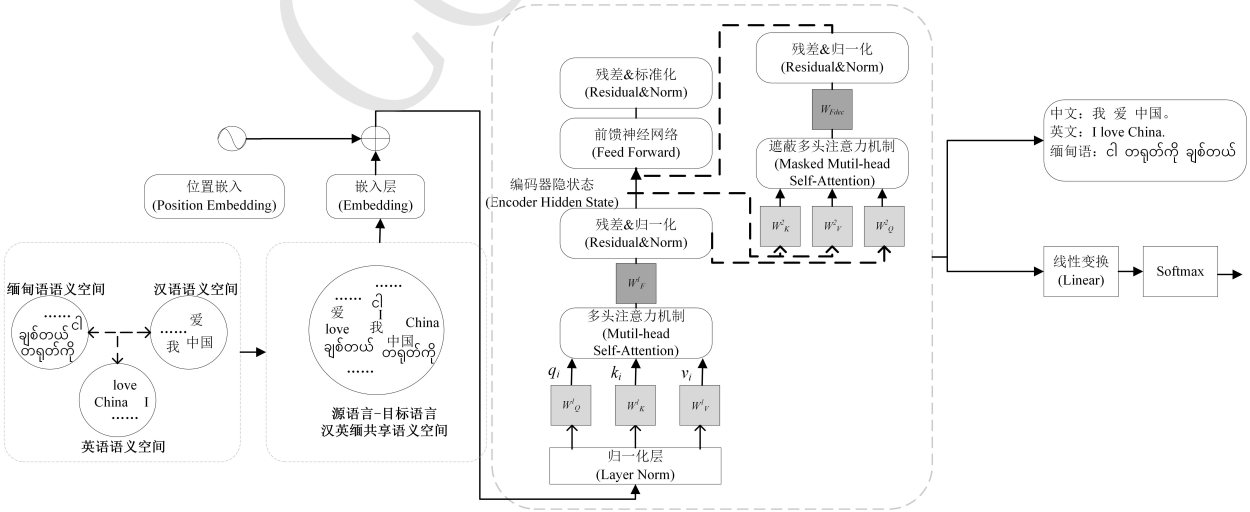


Figure 2: 基于多语言联合学习的汉英缅神经机器翻译模型框架

针对于汉英缅三种语言各自语义空间不同以及翻译训练过程中参数共享导致的翻译模型的性能不佳的问题。本文在Transformer的架构下进行汉英缅多语言神经机器翻译，本文的具体模型架构如图2所示。

4.1 汉英缅多语言语义空间映射

传统的多语言机器翻译方法都是采用共享词表。但是，当多种差异性较大的语言进行共享词表时，在词表中每种语言可用的词表内容会变小，因此，每种语言的词汇表征能力会相应地降低，这就是当相似语言进行共享此表示效果会比差异性性较大的语言共享词表效果更好的原因。如图2所示，汉语、英语以及缅甸语三种语言之间的差异性较大。汉语、英语都属于SVO语序的语言，其语句特点是有中心语的后置和前置。缅甸语则属于SOV语序（和其他语序为宾语先于谓语的语言）语言倾向于将形容词放在名词前面，此类顺序称之为中心语后置。这就意味着，在缅甸语和英语的句法构成中存在很大不同，而且缅甸语的虚词和助词在英语中却不存在对应的词。因此会出现不同语言的相同含义的词语之间无法对应，另外还存在一种语言中的单词可能被翻译成另一种语言的一个或多个单词等问题，以上的语言特点都会导致三种语言的语义空间较为独立(Win, 2011)。

针对以上的问题，在汉英缅多语言联合训练的神经机器翻译任务中，我们在三种语言共享词表之前，首先将三种语言词嵌入进行共享，这样可以保证我们得到的是三种语言共享词嵌入的语义空间而不是三种语言的词汇相似程度。具体如下：

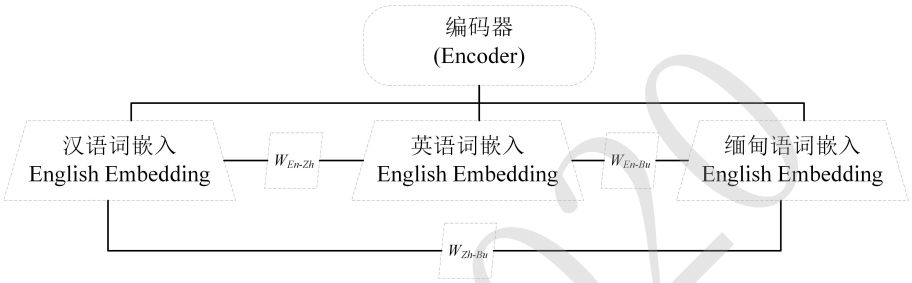


Figure 3: 汉英缅多语言词嵌入语义映射图

(1)如图3，分别训练汉英缅三种语言的单语词嵌入，我们利用Mikolov等人(Mikolov et al., 2013)提出的skip-gram算法得到汉英缅三种语言的单语词嵌入E_{Bu}，E_{Zh}。

(2)给定一句包含n个单词的源语言英语句子为$\mathrm{x_{En}} = (x_1, x_2, ..., x_{|\mathrm{n}|})$我们将编码器端生成的英语词向量表示$E_{EN}$进行提取。任意一个英语单词的经过lookup在${R_{En}}^{D\times V^{En}}$操作后的词向量表示为$E_{EN}(x_{En})$。其中，$D$表示词嵌入的维度，在本文中，我们设置为512维，$V^{En}$表示为英语的词表。

(3)给定一句包含n个单词的源语言的中文句子、缅甸语句子为$\mathrm{x_{Zh}} = (x_1, x_2, ..., x_{|\mathrm{n}|})$，$\mathrm{x_{Bu}} = (x_1, x_2, ..., x_{|\mathrm{n}|})$。分别将汉语和缅甸语与英语的词向量学习跨语言线性映射矩阵$W_{En-Zh} \in {R_{En-Zh}}^{D\times D}$，$W_{En-Bu} \in {R_{En-Bu}}^{D\times D}$最小化表示：

$$\sum_{(x_{Zh},x_{En})\in D_{Zh-En}} \|W_{En-Zh}E_{Zh}(x_{Zh}) - E_{En}(x_{En})\|2 \tag{6}$$

$$\sum_{(x_{Bu},x_{En})\in D_{Bu-En}} \|W_{En-Bu}E_{Bu}(x_{Bu}) - E_{En}(x_{En})\|2 \tag{7}$$

(4)将(3)中得到的$W_{En-Bu}E_{Bu}(x_{Bu})$和$W_{En-Zh}E_{Zh}(x_{Zh})$在进行跨语言线性最小化的表示：

$$\sum_{(x_{Zh},x_{Bu})\in D_{Zh-Bu}} \|W_{En-Bu}E_{Bu}(x_{Bu}) - W_{En-Zh}E_{Zh}(x_{Zh})\|2 \tag{8}$$

由于缅甸语属于低资源语言，几乎没有开源的英缅以及汉缅的双语词典，其中，步骤3、步骤4中的词典$D_{Bu-En}, D_{Zh-En}, D_{Zh-Bu}$是通过将汉语、英语以及缅甸语三种语言的单语数据利用GIZA++[-1]词对齐的方式得到。通过词对齐的方式我们可以获得较高质量的英语-缅甸语，缅甸语-汉语以及汉语-英语的双语词典数据。

[-1]https://github.com/moses-smt/giza-pp

(5)最后，利用学习到的缅甸语词嵌入矩阵对源语言的词嵌入矩阵进行替换，替换后的词嵌入矩阵具有汉语、英语以及缅甸语三种语言的语义信息，更进一步的增强了三种语言对齐的语义空间。

通过步骤(1)-步骤(5)，我们得到了汉英缅三种语言的共同的语义表征方法。我们将汉英缅三种语言的词语进行了词嵌入转换，通过在数学空间中进行匹配转换不考虑特定语言的字母，缓解了在嵌入级别的汉英缅三种语言词汇不匹配的问题。

4.2 汉英缅多语言参数共享

如图2，在神经机器翻译的结构中，当编码器在对汉语、英语以及缅甸语进行编码训练的过程时，每个子层都会产生训练过程中的参数(Sachan and Neubig, 2018)，为了将汉英缅三种语言训练过程中的语义距离减小，我们将所有子层生成的参数进行共享。除了3.1节中的共享词嵌入映射在相同的语义空间之外，具体参与共享的参数还有：(1)自注意力机制中的线性变化产生的参数：$W_K^1, W_Q^1, W_V^1, W_F^1$(2)编码器-解码器的注意力机制：$W_K^2, W_Q^2, W_V^2, W_F^2$。在汉英缅多语言神经机器翻译中，为了将三种语言对之间的训练参数最大化共享。本文将以上的所有训练参数进行共享，减小语义差距。具体共享过程如下：

Step1：共享前馈神经网络中的子层的参数($\Theta = \{W_E, H_{enc}, W_{L1}, W_{L2}\}$),其中，$W_E$表示经过语义映射后的词嵌入表示，$H_{enc}$表示编码过程中的隐状态表示，$W_{L1}, W_{L2}$表示输入共享语义空间的向量以及前馈神经网络中的参数。

Step2：共享自注意力子层中的特征权重($\Theta = \{W_E, H_{enc}, W_K^1, W_Q^1, W_V^1, W_F^1\}$)，其中，$W_K^1, W_Q^1, W_V^1$表示注意力机制层生成的参数，$W_F^1$表示前馈神经网络生成的训练参数。

Step3：共享编码器-解码器注意力机制的子层($\Theta = \{W_E, H_{enc}, W_K^2, W_Q^2, W_V^2, W_F^2\}$)，其中，$W_K^2, W_Q^2, W_V^2$表示注意力机制层生成的参数，$W_F^2$表示前馈神经网络生成的训练参数。

5 实验

5.1 实验数据

汉英缅语义空间映射实验数据：在本文中，我们利用GIZA++的方式对汉英、英缅以及汉缅的双语词语进行对齐，我们使用构建的10万汉缅、英缅的双语句子以及开源的9.0M英汉双语句子进行词对齐的训练，获得英汉、英缅以及汉缅的双语词典。另外，在实验过程中，我们将训练汉英缅多语言映射的实验数据进行训练集、测试集以及验证集的划分比例为：8:1:1。

汉英缅翻译实验数据：本文在一对多和多对多的两个翻译场景下进行汉英缅的多语言神经机器翻译。在本文中，一对多的翻译场景是指源语言端是汉语，目标语言端是英语和缅甸语，源语言端是英语，目标语言端是汉语和缅甸语以及源语言端是缅甸语，目标语言端是英语和汉语。多对多的翻译场景是指源语言端是汉语、英语以及缅甸语，目标语言端也是汉语、英语以及缅甸语。针对于汉语和英语的双语语料，我们利用来自2018国际机器翻译大会（WMT-18）[0]的中英数据集。目前，公开的英缅，汉缅数据较少。本文利用构建了10万句英缅以及10万句汉缅的双语语料。具体的语料的信息如表1所示。

数据集	训练集	验证集	测试集
汉英	9.0M	2000	2000
英缅	98K	2000	2000
汉缅	98K	2000	2000

Table 1: 实验数据集表

5.2 实验设置

多汉英缅多语言语义空间映射实验设置：在实验过程中，我们将词向量的维度设置为512维，训练过程中的迭代次数设置为10。

[0]http://www.statmt.org/wmt18/translation-task.html

翻译实验设置：在实验中，使用比特对编码(Byte Pair Encoding, BPE)[1](Sennrich et al., 2015)对汉语、英语以及缅甸语的单词进行了亚词切分。词表大小为35K。使用NIST的BLEU脚本[2]对翻译结果进行评测。实验环境为Ubuntu16.04，Linux系统，Tensorflow版本为tensorflow-gpu的1.13.2，编译语言为Python3.7。我们选择Transformer_Base作为我们实验的基础框架设置。在实验过程中，我们设置Transformer的编码器和解码器均为6层，在编码器和解码器中的词向量的维度以及注意力机制的单元为512。我们使用Adam优化器对训练过程中的学习率进行调整。另外，为了防止过拟合的问题，我们设置Dropout参数为0.1，在训练过程中我们设置每个batch包含的源语言单词为2048，每3000个batch对验证集做一次解码，如果连续10次的验证集中的BLEU不再提高，则提前终止训练，防止模型过拟合。

5.3 实验结果及对比分析

实验一：一对一及一对多翻译场景下不同模型实验结果对比分析

在实验一中，我们设置翻译场景为一对多，一种源语言对应多种目标语言。具体如下表2所示，分别选择汉语、英语以及缅甸语作为源语言对应另外两种不同的目标语言。我们设置对比实验如下：

(1)谷歌多语言神经机器翻译模型(MNMT)：Johnson等人(2017)提出了一种基于双向LSTM的多语言神经机器翻译模型。

(2)Dong等人(2015)在一对多的翻译场景下，提出将多语言翻译过程中的源语言的编码器共享，为每个目标语言分配不同的解码器的方法。

(3)Transformer：我们将比较在仅使用一对一的翻译场景下的源语言到目标语言的实验结果。

(4)Basline：基线模型是指在Transformer的框架下，不使用共享语义空间以及共享参数的思路进行的翻译实验。

翻译场景	方法	源语言-目标语言					
		英汉	英缅	汉英	汉缅	缅英	缅汉
一对一	Transformer	26.02	14.55	25.08	14.09	16.77	16.00
一对多	MNMT	25.56	13.09	23.45	13.78	15.40	16.14
	Dong等人	24.77	14.87	23.60	13.55	15.30	16.40
	Baseline	26.04	15.06	24.89	14.35	16.80	16.30
	本文方法	**26.80**	**16.42**	**26.20**	**15.77**	**17.68**	**19.30**

Table 2: 一对一及一对多翻译场景下的实验结果表

如表2所示，在一对多的翻译场景下，本文提出的方法在汉语-缅甸语的翻译方向上BLEU值达到了15.77。相比较于谷歌提出的多语言神经机器翻译模型有明显的提升，提升了1.99个BLEU值，这说明基于Transformer的多语言神经机器翻译框架中的遮蔽注意力机制可以更好地对目标语言的进行翻译；Dong等人提出的方法的翻译效果在一对多的翻译场景下和谷歌提出的多语言神经机器翻译模型方法相当，在翻译方向相同的情况下，本文提出的方法对于汉语-缅甸语以及缅甸语-汉语的效果都要更好。另外，在相同的翻译方向，例如，汉语-缅甸语，本文提出的方法在一对多的情况下相比较于一对一的实验结果提升了1.68个BLEU值，这说明利用大规模的汉语-英语的双语平行语料和较少的汉缅平行语料进行多语言联合神经机器翻译可以充分的弥补汉语-缅甸语之间数据缺乏导致模型效果不佳的问题。

实验二：一对一及多对多翻译场景下实验结果对比分析

在多到多的翻译场景下，我们将源语言端、目标语言端设置为汉语、英语以及缅甸语三种语言，我们将对比其他低资源神经机器翻译方法，具体如下：

[1]https://github.com/bheinzerling/bpemb
[2]https://www.nist.gov/itl/iad/mig/tools

(1)谷歌多语言神经机器翻译模型：Johnson等人(2017)提出了一种基于双向的LSTM的多语言神经机器翻译模型。

(2)基于RNNSearch(RS)的多语言神经机器翻译方法：在编码和解码端均使用RNN的网络结构(Bahdanau et al., 2014)。

(3)基于枢轴的神经机器翻译方法：kim等人(2019)提出一种基于枢轴的神经机器翻译方法，本文在保持其他设置不变的情况下，利用基于枢轴的方法对汉英缅进行神经机器翻译，在实验中仅使用汉英、英缅数据获得汉缅的机器翻译模型。

翻译场景	方法	源语言-目标语言					
		英汉	汉英	英缅	缅英	汉缅	缅汉
一对一	Transformer	26.02	25.08	14.55	16.77	14.09	16.00
多对多	MNMT	24.01	24.30	13.90	14.66	13.09	15.05
	RS	25.67	23.43	13.76	15.22	14.56	15.67
	枢轴	-	-	-	-	14.05	17.09
	Baseline	25.44	24.80	14.22	14.50	15.44	15.86
	本文方法	**25.89**	**24.86**	**17.06**	**17.33**	**16.82**	**18.91**

Table 3: 一对一及多对多翻译场景下的实验结果表

如表3所示，在汉语-缅甸语的翻译方向，本文提出的方法相比较于基于枢轴的方法提高了2.77个BLEU值。在缅甸语-汉语的翻译方向，本文提出的方法相比较于基于枢轴的方法提高了1.82个BLEU值，这说明利用多语言的联合学习的方式可以有效的通过高资源语言对弥补低资源语言对的数据稀缺的问题。在相同的翻译方向，本文方法都比RNNSearch的BLEU效果明显，利用Transformer可以更好地将翻译信息融合，利用参数共享的思想可以将语义之间的距离缩小。

同时，对比表3和表4我们可以发现当翻译方向相同时，多对多的翻译场景的效果好于一对多的翻译场景。例如，汉-缅和缅-汉的多对多的翻译效果均优于一对多的翻译场景。以上的实验现象说明，利用共享编码器的思想可以较好地将三种语言之间进行映射，减小语言之间的差异性。

在英语-汉语翻译方向，本文的方法BLEU值为25.89，一对一的Transformer模型的BLEU值为26.02，本文方法下降了0.13个BLEU值，因为，在多到多的翻译场景下，汉语-缅甸语的数据在汉语-英语的数据中增加了部分的噪声，导致模型性能下降。

实验三：多对多翻译场景下不同的词嵌入表征方式对汉英缅翻译效果的影响

在多到多的翻译场景下，我们将讨论利用不同的词嵌入表征方式对翻译效果的影响，具体实验结果如表4所示：

(1)单独表示：对汉英缅每种语言都进行单独的初始化，不共享词表及词嵌入的表示。

(2)Weight Tying：press等人(2017)提出一种将机器翻译中所有的输入和输出词向量同时共享的方法。

(3)本文方法：将汉英缅三种语言的词嵌入共享，编码及解码都进行相同语义空间的映射表示。

如表4所示，本文方法的BLEU值高于其他两种词嵌入的表示方法，说明在实验过程中，将三种语言进行映射利用相同的表征会缩小语义空间的距离，提升翻译模型的性能。同时，也验证了本文方法的有效性。

实验结果显示了本文方法对于汉缅、英缅的显著提升，使用不同的向量空间的表示方法对实验结果也有一定的影响，汉语、英语以及缅甸语的语义空间相对较为独立，使用单独的表示方法将三种语言的语义空间独立出来，缅甸语相对英语和汉语之间的差异性极大，这样的方法没办法将三种语言的语义空间统一缩小。利用Weight Tying的方法虽然可以较好的缩小三种语

翻译场景	词嵌入方法	源语言-目标语言					
		英汉	汉英	英缅	缅英	汉缅	缅汉
多对多	单独表示	24.62	23.50	15.40	16.61	14.22	16.56
	Weight Tying	24.67	23.55	15.76	16.72	14.56	16.42
	本文方法	**25.89**	**24.86**	**17.06**	**17.33**	**16.82**	**18.91**

Table 4: 不同的词嵌入表征方式对汉英缅翻译效果的影响

言的语义空间，减小语义差异，但是针对于缅甸语这种低资源语言，会出现汉缅、英缅之间较多的一对多或者多对一的情况，本文方法更好的缩小了这种语言之间的差异性，使三种语言更好的表示在同一个语义空间。

实验四：不同的词汇表设置对汉英缅翻译效果的影响

在实验过程中，我们设置汉语、英语以及缅甸语的词汇表为独立的35K大小的词汇表，我们对比分析基线系统中独立使用词汇表对实验结果的影响。具体如下表5所示。

翻译场景	方法	源语言-目标语言					
		英汉	汉英	英缅	缅英	汉缅	缅汉
一对多	Baseline(独立词表)	25.55	23.20	13.72	14.04	13.17	15.57
	本文方法	**26.80**	**26.20**	**16.42**	**17.68**	**15.77**	**19.30**
多对多	Baseline(独立词表)	24.45	23.98	14.10	13.44	14.97	13.26
	本文方法	**25.89**	**24.86**	**17.06**	**17.33**	**16.82**	**18.91**

Table 5: 不同的词汇表设置对汉英缅翻译效果的影响

在独立词表的情况下，将本文方法与Baseline的方法进行对比，在一对多和一对多两个场景下，本文方法的BLEU都明显优于Baseline的方法。设置独立词表对比基线系统和本文提出的方法，在缅甸语、汉语以及英语的词表大小相同的情况下，本文提出的方法可以更好地将实验过程中的参数以及训练策略共享，通过高资源语言对的训练信息弥补低资源语言优于缺少数据导致翻译性能不佳的问题。

5.4 翻译示例分析

如图4、图5所示，我们将翻译方向定位汉语-缅甸语，当对输入的源语言汉语句子中的“中国”进行翻译时，基线模型的翻译输出误判为两个缅甸语单词，本文方法输出的缅甸语句子正确的将两种语言的词语进行对齐，将汉缅句子中的单词在不同的语义空间中进行了校正，另外，对于基线模型输出的英语句子中“Today”时漏翻译以及错翻译，翻译示例证明了本文方法不但能将缅甸语、英语以及汉语三语语义空间不齐的问题进行了校正，也能够将互译的英缅、汉缅以及英汉单词进行校正也验证了本文方法的有效性。

针对于汉缅、英缅翻译效果的显著提升，在文中我们通过减小三者的语义空间缩小不同语言之间的距离，通过构建三语的词向量映射，构建词典将低资源语言缅甸语和英语、汉语进行语义映射，得到更好的双语表示。因此，通过实例证明本文方法得到了较好的翻译效果。

6 结论

针对汉英缅结构差异大导致多语言翻译时共享词表能力受限的问题，本文利用汉语、英语

1. 源语言句子（汉语）	今天，我们将从缅甸的仰光市回到中国。
目标语言句子（缅甸语）	
基线系统	ယနေ့ကျွန်ုပ်တို့သည်မြန်မာနိုင်ငံမှရန်ကုန်မှ**မြည်တရု**သို့ပြန်သွားပါမည်။
本文方法	ယနေ့ငါတို့ မာနိုမြန်မာနိုင်ငံကရန်ကုန်ကနေ **တရုတ်**ပြည်ပြန်တော့မယ်။
参考译文	ဒီနေ့ငါတို့ မြန်မာနိုင်ငံကရန်ကုန်ကနေ **တရုတ်**ပြည်ပြန်တော့မယ်။
目标语言句子（英语）	
基线系统	We **are going to** return to China from Yangon, Myanmar.
本文方法	We will return to China from Yangon, Myanmar, **today**.
参考译文	**Today,** we will **return to** China from Yangon, Myanmar.

Figure 4: 汉缅，汉英翻译示例

2. 源语言句子（缅甸语）	**မြန်မာ** - တရုတ်ဆက်ဆံရေးသည်**ရှေးအချိန်ကတည်းက**အလွန်**ဖော်ရွေခဲ့သည်**။
目标语言句子（汉语）	
基线系统	缅甸语言一直和中国的关系很融洽。
本文方法	从很早的时候，缅甸和中国的关系就很友好。
参考译文	缅甸与中国的关系自古以来就非常友好。
目标语言句子（英语）	
基线系统	**Fast** - China has been **very popular since the beginning**.
本文方法	**Myanmar**-China relations have been **better since the beginning**.
参考译文	**Myanmar**-China relations have been very **friendly since ancient times**.

Figure 5: 汉缅，汉英翻译示例

以及缅甸语三种语言进行联合语义的表征来提升缅汉机器翻译模型的性能。实验结果表明提出的方法在一对多的翻译场景下，汉-缅的翻译方向上达到了15.77的BLEU值，在多对多的翻译场景下，汉-缅的翻译方向上达到了16.82的BLEU值，相比较于基线模型均有明显的提升。在下一步的工作中，我们将在多语言翻译框架下探索不同的参数共享方式对翻译效果的影响，从而提升缅甸语的机器翻译性能。

参考文献

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. pages 3874–3884.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv: Computation and Language*.

R Caruana. 1997. Multitask learning. *Machine Learning*, 28(1):41–75.

Raj Dabre, Atsushi Fujita, and Chenhui Chu. 2019. Exploiting multilingualism through multistage fine-tuning for low-resource neural machine translation. pages 1410–1416.

Daxiang Dong, Wu Hua, He Wei, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *Meeting of the Association for Computational Linguistics International Joint Conference on Natural Language Processing*.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016a. Multi-way, multilingual neural machine translation with a shared attention mechanism.

Orhan Firat, Baskaran Sankaran, Yaser Alonaizan, Fatos T Yarman Vural, and Kyunghyun Cho. 2016b. Zero-resource translation with multi-lingual neural machine translation. *arXiv: Computation and Language*.

Thanh Le Ha, Jan Niehues, and Alexander Waibel. 2016. Toward multilingual neural machine translation with universal encoder and decoder.

Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda B Viegas, Martin Wattenberg, Greg S Corrado, et al. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5(1):339–351.

Yunsu Kim, Petre Petrov, Pavel Petrushkov, Shahram Khadivi, and Hermann Ney. 2019. Pivot-based transfer learning for neural machine translation between non-english languages. pages 866–876.

Surafel Melaku Lakew, Aliia Erofeeva, Matteo Negri, Marcello Federico, and Marco Turchi. 2018. Transfer learning in multilingual neural machine translation with dynamic vocabulary. *arXiv: Computation and Language*.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2017. Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics*, 5:365–378.

Constantine Lignos, Daniel Cohen, Yenchieh Lien, Pratik Mehta, W Bruce Croft, and Scott Miller. 2019. The challenges of optimizing machine translation for low resource cross-language information retrieval. pages 3495–3500.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *Computer Science*.

Khin Thandar Nwet, Khin Mar Soe, and Lar Thein Ni. 2011a. Building bilingual coupus based on hybrid approach for myanmar-english machine translation. *International Journal of Scientific Engineering Research*, Volume 2(Issue 8).

Khin Thandar Nwet, Khin Mar Soe, and Lar Thein Ni. 2011b. Developing word-aligned myanmar-english parallel corpus based on the ibm models. *International Journal of Computer Applications*, 27(8).

Ofir Press and Lior Wolf. 2017. Using the output embedding to improve language models. 2:157–163.

Devendra Singh Sachan and Graham Neubig. 2018. Parameter sharing methods for multilingual self-attentional translation models. *arXiv: Computation and Language*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *Computer Science*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *arXiv: Computation and Language*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. pages 5998–6008.

Yining Wang, Jiajun Zhang, , Feifei Zhai, and Jingfang Xu. 2018. Three strategies to improve one-to-many multilingual translation. In *Conference on Empirical Methods in Natural Language Processing*.

Yining Wang, Long Zhou, Jiajun Zhang, Feifei Zhai, and Chengqing Zong. 2019. A compact and language-sensitive multilingual translation method. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Aye Thida Win. 2011. Words to phrase reordering machine translation system in myanmar-english using english grammar rules. 3:50–53.

Barret Zoph and Kevin Knight. 2016. Multi-source neural translation.

基于跨语言双语预训练及Bi-LSTM的汉-越平行句对抽取方法

刘畅[1,2]，高盛祥*[1,2]，余正涛[1,2]，黄于欣[1,2]，尤丛丛[1,2]
1.昆明理工大学，信息工程与自动化学院，昆明，650500
2.昆明理工大学，云南省人工智能重点实验室，昆明，650500
liuxiaochang32@163.com, gaoshengxiang.yn@foxmail.com
ztyu@hotmail.com, huangyuxin2004@163.com, 1257767625@qq.com

摘要

汉越平行句对抽取是缓解汉越平行语料库数据稀缺的重要方法。平行句对抽取可转换为同一语义空间下的句子相似性分类任务，其核心在于双语语义空间对齐。传统语义空间对齐方法依赖于大规模的双语平行语料，越南语作为低资源语言获取大规模平行语料相对困难。针对这个问题本文提出一种利用种子词典进行跨语言双语预训练及Bi-LSTM（Bi-directional Long Short-Term Memory）的汉-越平行句对抽取方法。预训练中仅需要大量的汉越单语和一个汉越种子词典，通过利用汉越种子词典将汉越双语映射到公共语义空间进行词对齐。再利用Bi-LSTM和CNN（Convolutional Neural Networks）分别提取句子的全局特征和局部特征从而最大化表示汉-越句对之间的语义相关性。实验结果表明，本文模型在F1得分上提升7.1%，优于基线模型。

关键词： 汉-越；平行句对抽取；跨语言预训练；公共语义空间；Bi-LSTM

Chinese-Vietnamese Parallel Sentence Pair Extraction Method Based on Cross-lingual Bilingual Pre-training and Bi-LSTM

Chang Liu[1,2], Shengxiang Gao*[1,2], Zhengtao Yu[1,2],
Yuxin Huang[1,2], Congcong You[1,2]
1. Faculty of Information Engineering and Automation,
Kunming University of Science and Technology Kunming 650500, China
2. Yunnan Key Laboratory of Artificial Intelligence,
Kunming University of Science and Technology Kunming 650500, China
liuxiaochang32@163.com, gaoshengxiang.yn@foxmail.com
ztyu@hotmail.com, huangyuxin2004@163.com, 1257767625@qq.com

Abstract

The extraction of Chinese-Vietnamese parallel sentence pairs is an important method to alleviate the scarcity of Chinese-Vietnamese parallel corpus data. Parallel sentence pair extraction can be converted into sentence similarity classification task in the same semantic space, the core of which is to achieve bilingual semantic space alignment. The traditional semantic space alignment method relies on large-scale bilingual parallel corpus, and it is relatively difficult for Vietnamese to obtain large-scale parallel corpus as a low-resource language. To address this problem, this paper proposes a bilingual dictionary for cross-lingual bilingual pre-training and Bi-LSTM (Bi-directional Long Short-Term Memory) Chinese-Vietnamese parallel sentence pair extraction method. Only a large number of Chinese-Vietnamese monolingual and a Chinese-Vietnamese

*通信作者：高盛祥，email地址：gaoshengxiang.yn@foxmail.com
基金项目：国家自然科学基金（61761026，61972186，61732005，61672271，61762056）；国家重点研发计划（Nos.2019QY1802，2019QY1801，2019QY1800）；云南省自然科学基金2018FB104；云南高科技人才项目（201606）；昆明理工大学省级人培项目（KKSY201703005）

seed dictionary are required for pre-training. By using the Chinese-Vietnamese seed dictionary to map the Chinese-Vietnamese bilingual to the common semantic space for word alignment.Then, Bi-LSTM and CNN (Convolutional Neural Networks) are used to extract the global and local features of sentences to maximize the semantic relevance between Chinese-Vietnamese sentence pairs. Experimental results show that the model in this paper improves F1 score by 7.1%, which is better than the baseline model.

Keywords: Chinese-English-Burmese , Low Resource Language , Multilingual Neural Machine Translation , Joint training , Semantic space mapping , Shared parameters

1 引言

平行语料库的规模和质量对于机器翻译的性能至关重要。在大规模语料中的机器翻译如汉-英神经机器翻译中都已经获得很好的结果。由于汉-越低资源语言很难获得足够多的平行句对，从而导致汉-越机器翻译性能较差。人工手动构建大规模高质量的平行语料库耗时耗力，通过大量的文本研究发现在同一段时间内报道的新闻或具有同一结构的网页都能获得大量的可比语料，从可比语料中抽取平行句对是扩充翻译语料的重要方法之一。本文的目的是从汉越可比语料中抽取汉越平行句对。

目前双语平行句对抽取的方法大致可以分为以下四类：首先是利用统计机器翻译和神经机器翻译方法从可比语料库中抽取平行句对是比较有效的。在统计机器翻译方面，Rauf等人(Rauf and Schwenk, 2011)的方法是将目标语言翻译成源语言，利用跨语言信息检索技术从可比语料库中抽取平行句对，提高了统计机器翻译的性能；在神经机器翻译方面，(Marie and Fujita, 2017; Choudhary et al., 2018)提出了基于词嵌入在大型单语语料库中抽取平行句对从而提升了神经机器翻译的性能。Utiyama等人(Masao Utiyama, 2013)经过两次机器翻译，首先将日语句子翻译得到n-best英语译文，再把英语译文翻译成汉语，构建中日平行语料库。这些方法都是通过有效抽取平行句对来提升机器翻译的性能，但需要在翻译模型性能比较好的基础上才能进行。

其次在基于特征工程方面，(Chuang et al., 2004; España-Bonet et al., 2017; Luong et al., 2015)提出了在双语词典信息的基础上结合了标点符号统计信息和词汇信息的双语平行文本对齐的方法；Gale等人(Gale and Church, 1991)介绍了一种基于字符长度的统计模型对齐平行文本中的句子的方法，识别一种语言的句子和另一种语言的句子之间的长度对应关系。Peng等人(Peng et al., 2010)提出了一种Fast-Champollion句子对齐算法，它结合了基于长度和基于词典信息，通过将输入的双语文本分割成小块进行对齐的过程，提升句子对齐的效果。Ann 等人(Masao Utiyama, 2013)基于现有的翻译系统，将源语言翻译成目标语言得到候选句子，然后对候选句子对进行打分排序，从而获得平行句子。Chu等人(Chu et al., 2016)从对齐的文章中通过笛卡尔乘积生成所有可能的句子对，并过滤掉不满足条件的句子对，保留尽可能匹配的句子对，然后使用少量平行句对训练分类器，以从候选者中识别平行句对。Tillmann等人(Tillmann and Xu, 2009) 提出了一种用于可比数据的新颖句子对提取算法，直接在句子级别对大量候选句子对进行评分。通过一个简单对称评分函数实现句子级别的提取。但这些方法通常依赖于大量的与语言相关的特征知识，虽然证明了抽取平行句对的有效性，但是由于句对分类准确性不高，无法取得较好的效果。

然后在基于深度学习方面，Francis Gregoire 等人(Grégoire and Langlais, 2017)提出基于双向递归神经网络对源语言和目标语言分别进行编码，然后经过分类器区分源句子和候选目标句子是否平行；Munteanu等人(Munteanu and Marcu, 2005)提出一种利用最大熵分类器从大量可比语料中抽取平行句对的方法，从零开始构建了汉英翻译系统。Grover 等人(Grover and Mitra, 2017)训练模型以获取双语单词嵌入，然后在两个句子的单词之间创建相似度矩阵，并使用卷积神经网络（CNN）将句子分类。Bouamor等人(Bouamor and Sajjad, 2018)通过将多语言句子级嵌入，并与神经机器翻译和监督分类配对的混合方法，来分类法语-英语语料库中的平行句子对。首先通过双语分布式表示模型学习的每个源－目标句子对的连续向量表示对目标翻译候选进行过滤。然后，使用神经机器翻译系统或二进制分类模型选择最佳翻译。它们能有效利用深度学习的方法从可比语料中抽取平行句对但它们在训练过程中需要大量双语平行句对。

最后在句子相似度计算的方面，(Cheon and Youngjoong, 2017; Azpeitia et al., 2018)提出了一种利用语言资源的顺序匹配在句子之间执行相似度计算从而查找相似句子的方法，从维基百科构建英语和韩语之间的平行语料库。Alberto 等人(Barrón-Cedeño et al., 2015)通过余弦和跨语言信息检索中的长度因子来计算句子对之间的相似性，从而对齐来自维基百科的特定于域的并行文档。这种方法是从句子级扩充训练数据，从而构建高质量的平行语料库但也都是针对资源丰富型语言（例如英语-法语），但在低资源语言（如汉语-越南语）上的性能较差，并且抽取出的句子噪声较大。

以上方法在预训练过程中均是利用大量的双语平行句对作支撑，但汉语和越南语都是独立派系的语言且汉越双语训练数据稀缺。通过大量的文本研究发现在同一段时间内报道的新闻或具有同一结构的网页都能获得大量的可比语料，因此如何从汉越可比语料库中获得平行句对具有重要意义。考虑到汉越双语平行句对很难获取而得到汉越单语句子相对容易，结合汉-越句子特性，受Francis Gregoire 等人(Grégoire and Langlais, 2018) 和Artetxe等人(Artetxe et al., 2016; Artetxe et al., 2017; Artetxe et al., 2018)思想启发，提出了一个基于跨语言双语预训练及Bi-LSTM的汉-越平行句对抽取方法，从汉越可比语料中抽取汉越平行句对，来提升低资源语言机器翻译的性能。其主要思想是在汉越双语预训练中将汉越双语句子映射到公共语义空间下，通过汉-越种子词典来缩小汉越双语在语义空间中的距离，从而加强汉越双语的语义相关性。在本文方法中针对的是汉语到越南语两种语言，由于汉语到越南语没有公开的数据集，因此考虑从维基百科文章中抽取的汉-越段落语料以及收集的汉-越段落语料添加到一个语料库中，以训练模型的性能。

2 基于汉-越双语预训练及Bi-LSTM的平行句对抽取模型

针对上文问题，提出一个基于汉-越双语预训练及Bi-LSTM的平行句对抽取方法，具体模型结构体系如图1所示。该模型主要分为三个部分。第一部分是基于汉-越双语预训练，第二部分是由Bi-LSTM和CNN组成的汉-越句子特征提取部分的编码器，第三部分是全连接层进行汉-越平行和句非平行句分类。

首先，将汉语-越南语跨语言双语词嵌入映射到公共的语义空间进行预训练，使得汉语-越南语的语义相似词在该空间中接近，增强汉语和越南语语义空间中的相关性。设x = $(x_1, x_2, ..., x_m)$ 表示表示输入的汉语单词, y = $(y_1, y_2, ..., y_n)$ 表示输入的越南语单词。在双语预训练中，汉-越种子词典在没有大规模平行语料情况下可以实现在汉越统一空间语义对齐，并以自学习的方式迭代地生成新词典。再利用汉-越种子词典来学习词嵌入并指导后面Bi-LSTM和CNN在公共语义空间进行统一编码。将训练好的词向量输入Bi-LSTM来获取单词前后信息特征，并用CNN来提取双语句子更深层语义特征。最后对汉语句子和越南语句子进行编码，通过使用元素乘积和元素绝对差将它们提供给全连接层，使用输出概率作为汉越句对是否为平行语句对的度量来捕获其匹配信息。

3 汉越跨语言词向量预训练

3.1 词向量预训练方法

在双语中，利用单独语料进行独立训练的方法如Mikolov等人(Mikolov et al., 2013)的word2vec(CBOW/Skip-grim)训练出有语义相似性的词嵌入向量。在各自语料上进行独立训练，导致两种语言词嵌入矩阵在分布上也是独立不相关。在汉语和越南语词向量表征中也是如此。双语词嵌入将两种不同语言的词映射到公共的语义空间，公共语义空间中每个单词嵌入之间的距离则暗示着一定的语义关系。这可以保证在单语语义不变性情况下确保具有两个相同语义的词在公共语义空间中的距离非常近，但双语词嵌入的学习都依赖于大规模平行语料库，这对于资源稀缺型语言对（汉语-越南语）是难以获得的。

我们在汉越跨语言词向量预训练中提出了一种自学习的方法。该方法利用了嵌入空间的结构相似性，结合基于汉语-越南语种子词典的映射技术，降低了汉语-越南语双语资源的需求。该自学习的方法框架先是对汉语和越南语在各自的单语语料库上进行独立训练，再通过线性变换来最小化汉越双语词典中的距离从而将汉语-越南语跨语言映射在同一语义空间。通常需要大规模双语词典进行训练，针对汉语-越南语难以获取大规模词典，跨语言预训练中将大型双语词典的需求减少到较小的种子词典，通过不断迭代使更新的种子词典来学习新的映射矩阵，直至收敛。汉-越跨语言双语词嵌入预训练具体细节如下图2所示:

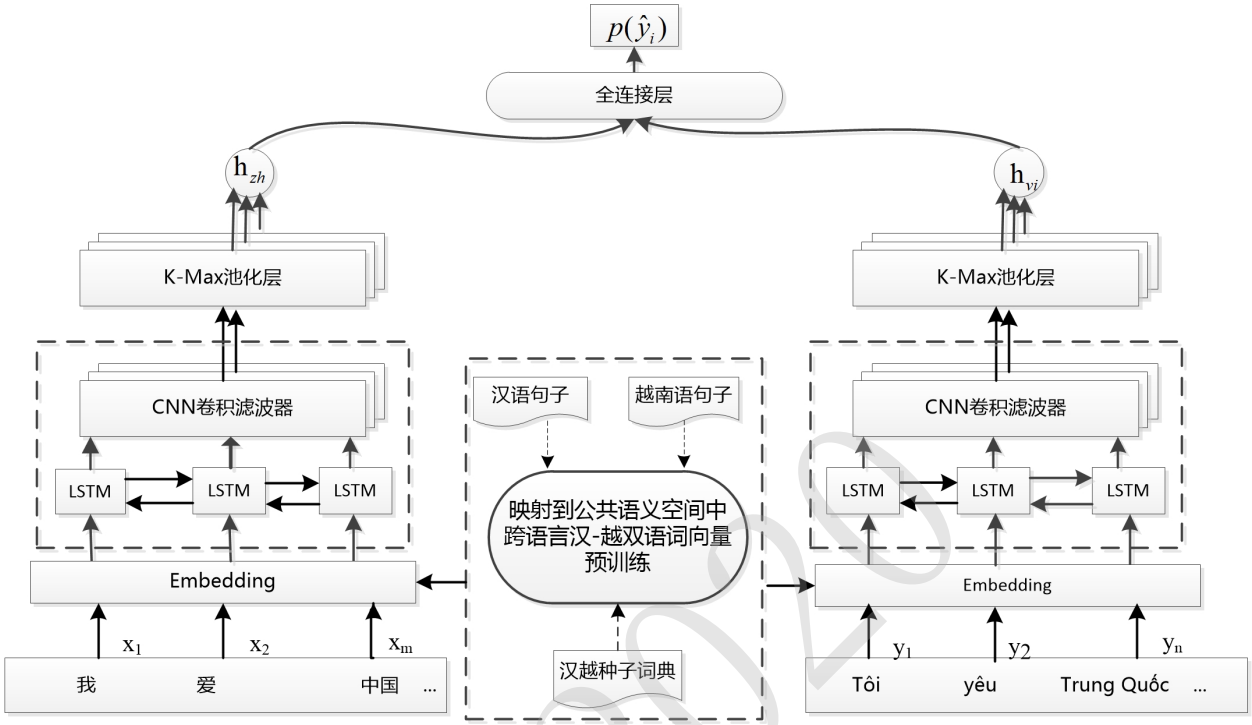


Figure 1: 基于汉-越双语预训练及Bi-LSTM的平行句对抽取模型

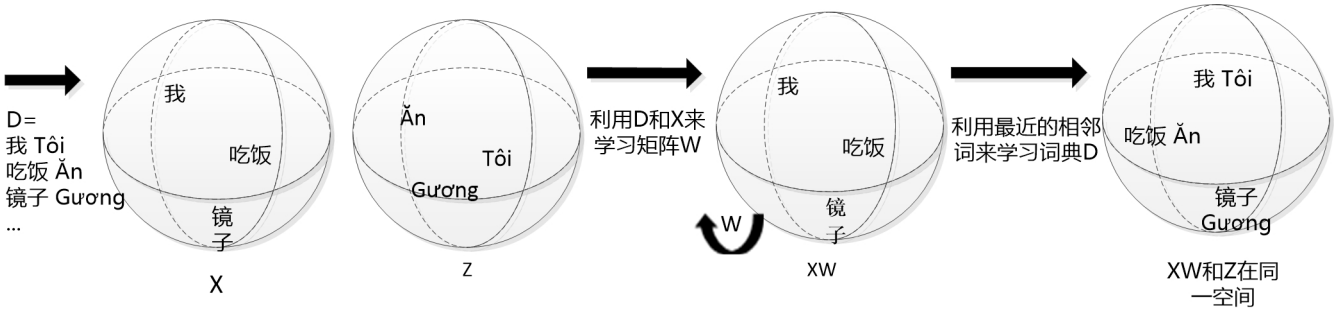


Figure 2: 汉-越跨语言双语词嵌入预训练过程

3.2 词向量预训练的基本步骤

首先，构建一个汉语和越南语同时映射的特征向量空间，汉语语料训练得到的词嵌入矩阵X，越南语语料中训练的词嵌入矩阵Z。将种子字典表示为一个二进制矩阵D。$D_{ij}=1$时代表越南语中的第j个单词是汉语中第i个单词的翻译。然后找到最佳映射矩阵W ,让汉语词向量和越南语词向量分布在同一个向量空间，使得映射汉语词嵌入$X_{i*}W$与越南语词嵌入Z_{j*}之间的欧几里德距离的平方和最小，映射矩阵：

$$W^* = \underset{W}{argmin} \sum_i \sum_j D_{ij} ||X_{i*}W - Z_{j*}||^2 \tag{1}$$

其中，预处理步骤对词嵌入矩阵X和Z进行长度归一化和平均居中，最后再进行一次归一化处理，并将W约束为正交矩阵即$WW^T = W^TW = I$，以强制执行汉语和越南语的单语不变性，防止单语性能的降低，同时可以产生更好的汉语-越南语跨原语言双语映射。在这种正交性约束下，最小化平方欧几里德距离就等于最大化点积，因此因此映射矩阵被定义为如下公式(2)所示：

$$W^* = \underset{W}{argmax}\, Tr(XWZ^TD^T) \tag{2}$$

其中，$Tr(\cdot)$表示主对角线上的所有元素之和，$W^* = UV^T$给出了此问题的最佳正交解，其中$X^TDZ = U\sum V^T$是X^TDZ的奇异值分解。由于字典矩阵D是稀疏的，这可以有效地在线性时间内对字典条目数进行计算。

获得了这个映射矩阵W之后，对于字典外的任何一个没有翻译的单词，可以根据映射后的空间余弦相似度来进行词对齐。在最近邻检索中，为每个源语言单词分配了目标语言中最接近的单词，我们将映射的源语言嵌入和目标语言嵌入之间的点积用作相似度度量。最后，通过矢量化相似矩阵XWZ^T并进行不断迭代计算，找到该矩阵的最大值，从而达到优化目标。

$$cos_{\text{dic}}(wx_i, z_j) = \frac{\sum_{i=1}^{n} wx_i z_j}{\sqrt{\sum_{i=1}^{n}(wx_i)^2 \sum_{j=1}^{n}(z_j)^2}} \tag{3}$$

4 基于Bi-LSTM和CNN公共语义空间编码

基于LSTM模型充分考虑了长距离单词之间的依赖性，并保留了诸如单词顺序之类的功能。同时CNN模型可以提取丰富的组合特征及卷积核的多样性。但是由于LSTM不使用反向单词编码信息，因此不能在双向单词编码中学习到语义信息特征，而Bi-LSTM可以考虑单词的双向编码。再使用CNN卷积并合并Bi-LSTM的输出以提取句子的关键语义特征。为了考虑上述特征，编码器由两层Bi-LSTM和CNN堆叠成一个基本的编码单元，依次从源语句和目标句中接受每个单词的单词嵌入矩阵$W_x \in R^{d\times|Vx|}$来输入单词x,其中d为单词嵌入向量的维数,V_x为所有输入单词的集合。每个时刻内，由词汇表V_x中的整数索引k定义的第i个句子中的标记表示为one-hot向量$w_k^S \in 0,1^{|V|}$,该one-hot向量与词嵌入矩阵$E^{S^T} \in R^{|Vx|\times d_e}$相乘，以获得该标记的连续向量表示$w_i^S$，其作为Bi-LSTM编码器的前向和后向循环状态的输入。前向LSTM读取变长句，并从第一个标记到最后一个标记更新其递归状态，从而创建一个固定大小的句子连续向量表示；后向LSTM反向处理该句子，然后将第二层相同位置上每个时间步长的两个方向的编码器输出都拼接在一起$h_i = [\overrightarrow{h}_i^S, \overleftarrow{h}_i^S]$，作为卷积神经网络的输入。前向递归状态和后向递归状态分别计算如下：

$$w_i^S = E^{S^T} w_k^S \tag{4}$$

$$\overrightarrow{h}_i^S = \phi(\overrightarrow{h}_{i-1}^S, w_i^S) \tag{5}$$

$$\overleftarrow{h}_i^S = \phi(\overleftarrow{h}_{i-1}^S, w_i^S) \tag{6}$$

$$h_i = [\overrightarrow{h}_i^S, \overleftarrow{h}_i^S] \tag{7}$$

其中E表示单词嵌入，$\phi(\cdot)$是LSTM模块。

原始的CNN由卷积层，池化层和全连接层组成。对于句子长度为n的句子，可以将它表示成$x_{1:n} = x_1 \oplus x_2 \oplus ... \oplus x_n$，$\oplus$表示全连接，$x_i \in \Re^d$表示的是第$i$个词向量，$d$表示的是词向量的维度。卷积运算的核心是对滑动窗口的大小的序列应用在过滤器上以产生新的特征，如下公式所示，

$$c_i = f(W \cdot x_{i:i+h-1} + b) \tag{8}$$

其中，$b \in \Re$是一个偏移向量，f是非线性函数（比如Sigmoid，ReLU）。长度为n的句子可以通过卷积层获得句子中任何连续单词序列的深层语义特征，如公式所示，

$$c = [c_1, c_2, ..., c_{n-h+1}] \tag{9}$$

本文将窗口大小为$F = [F(0)...F(\mathrm{m}-1)]$的卷积核与Bi-LSTM的输出向量进行卷积以获得特征向量，如公式所示:

$$c = tanh[(\sum_{i=0}^{m-1} h(t+i)^T F(i)) + b] \tag{10}$$

b是偏移向量，F和b是过滤器的参数。从典型的CNN结构可以看出，池化层构建在卷积层之上。在本文中，通过K-Max Pooling，每个滤波器最大值k会被保留，$\hat{c} = c_{k-max}$。

5 模型训练与分类

基于以上步骤，具有融合功能的Bi-LSTM 和CNN提取出源语句和目标句的语义特征，即C_i^S，C_i^T，然后使用元素积和绝对元素差来捕获它们的匹配信息，然后反馈到全连接的层以评估汉语-越南语句对相互翻译的可能性大小。具体公式如下:

$$C_i^a = C_i^S \odot C_i^T \tag{11}$$

$$C_i^a = |C_i^S - C_i^T| \tag{12}$$

$$C_i = tanh(W^a C_i^a + W^b C_i^b + b) \tag{13}$$

$$p(y_i|c_i) = \sigma(W^c c_i + c) \tag{14}$$

$$L = -\sum_{i=1}^{n(1+m)} y_i log\sigma(W^c h_i + c) - (1-y_i) log(1-\sigma(W^c h_i + c)) \tag{15}$$

其中$\sigma(\cdot)$是sigmoid激活函数W^a，W^b，W^c，b，c是模型参数，其中n是汉语句子的数量，m是候选越南语句子的数量。通过最小化标记的汉越句对的交叉熵作为损失函数来训练模型：对于预测，如果句子对的概率大于或等于设置的决策阈值ρ，则将其分类为平行；如果小于决策阈值ρ，则将其分类为不平行。

$$p(\widehat{y_i}) = \begin{cases} 0 & if \quad p(y_i = 1 \mid h_i) \geq \rho, \\ 1 & otherwise \end{cases} \tag{16}$$

6 实验与分析

6.1 实验数据

	汉越平行句对	汉越非平行句对
训练集	130k	130k
测试集	10k	10k
验证集	10k	10k

Table 1: 实验数据集表

本文将汉越平行句对抽取问题转化为二分类问题。由于在汉语到越南语低资源语言上，目前尚未找到用于训练的公开数据集，所以本文实验数据集的来源主要是从汉越新闻网站上检索和从维基百科dump获得的汉语-越南语翻译文章，该文章经过处理获取汉语句子315211，越南语单语句子316243，手动对齐以获得13万个汉越平行句对，使用VecMap工具训练高质量的跨语言双语词向量。同时基于每个平行句对的负采样样本数设置为1:1，随机构造了13万个汉越非平行句对，设置种子词典规模大小设置为3852个词条。同时为了衡量本文中汉越平行句对抽取模型分类器的性能，设置1万句汉越平行语料和1万句汉越非平行语料作为测试集。表1为本文基于跨语言双语预训练及Bi-LSTM的汉-越平行句对抽取模型的语料规模。

6.2 实验设置与评价指标

利用TensorFlow编写实现，单词嵌入维度和隐藏单元数均为300，隐藏层为1。批处理大小设置为64，训练的epochs为15，梯度截取设置为5.0，学习率设置为0.0002，Dropout设置为0.7-0.8，使用Adam优化器，激活函数采用sigmoid函数，损失函数为交叉熵损失函数。在评估指标方面，使用“精度”，“召回率”和“ F1值”作为衡量模型是否可以正确分类汉语-越南语是否为平行句子的指标。其中精度是所有提取的句子对中真正平行句子对的比例；召回率是测试集中所有平行句子对中真正平行提取的句子对的比例；F1值是精度和召回率的调和平均值。具体公式如下：

$$Precision = \frac{|TP|}{|TP + FP|} \tag{17}$$

$$Recall = \frac{|TP|}{|TP + FN|} \tag{18}$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \times 100 \tag{19}$$

其中，TP是提取句子中真正平行的句对的数量，FP是提取句子中非平行句对的数量，FN是测试集中未被提取的平行句对的数量。

6.3 实验结果与分析

跨语言双语预训练使用Artetxe等人(Artetxe et al., 2017)提出的VecMap开源框架对加强汉越语言相关性及Bi-LSTM和CNN更好地捕获取句子上下文信息和局部信息，设计了以下三组对比实验，再通过上面的评价指标进行实验评价与分析。

实验一：为了验证预训练方法的有效性，设置阈值为0.90，将经过预训练的Bi-LSTM和CNN汉越平行句对抽取模型与不经过预训练的效果进行对比。我们还将仅使用Bi-LSTM抽取汉越双语平行句对的基线方法进行比较，同时，为了突出分类器构造比传统机器学习更深入学习具有更好的准确性，同时还比较了Munteanu D S等人(Munteanu and Marcu, 2005)提出的最大熵模型。具体实验结果见表2。 从表2中可以看出，在汉-越数据集上，本文模型的F1得分优于基线模型和其他模型。使用深度学习方法的Bi-LSTM模型与机器学习是支持向量机模型（SVM）和线性回归（LR）分类模型相比具有更好的效果，主要原因是Bi-LSTM模型可以更好的学习句子向量特征，并且孪生网络将汉越两种语言共享到同一语义空间中可以一定程度上解决跨语言的问题而机器学习方法无法解决跨语言的问题使效果明显下降。经过深度学习训练的特征提取分类器比最大熵模型具有更好的性能。其主要原因是神经网络能够自动学习并提取更好的特征。Bi-LSTM和CNN的结合优化于简单使用Bi-LSTM，是因为通过CNN可以获得更多的语义特征信息。基线模型的效果为63.6%,而本文方法的F1值达到了70.7%,与不做预训练和CNN特征提取相比提高了7.1%。经过跨语言预训练的模型比单独使用Bi-LSTM和CNN编码的效果要好是因为将汉-越两种语言映射到相同空间，语义相关性更好。

实验二：为了进一步证明本文提出的汉语-越南跨语言预训练方法的有效性，设置在阈值为0.9，做了一组将本文在词向量表征部分与word2vec(Mikolov et al., 2013)的词向量表征模型的对比实验，具体实验结果如表3所示。 从表3中可以看出，本文提出的预训练方法VecMap比word2vec在汉越双语抽取工作中的效果要好，其主要原因是VecMap是跨语言双语词向量预训练将汉越双语映射到公共语义空间训练加强汉越跨语言相关性，从而能抽取到更高质量的汉越双语平行句对。

方法	R(%)	P(%)	F1(%)
最大熵模型	54.2%	49.6%	51.8%
LR	57.9%	53.8%	55.7%
SVM	62.2%	57.3%	59.6%
LSTM	65.8%	59.7%	62.6%
Bi-LSTM	67.2%	60.5%	63.6%
BiLSTM+CNN	69.9%	61.4%	65.3%
本文方法(VecMap+BiLSTM+CNN)	75.6%	66.5%	70.7%

Table 2: 不同模型对比实验结果

方法	R(%)	P(%)	F1(%)
word2vec-BiLSTM-CNN	72.8%	64.2%	68.2%
本文法方法（VecMap）	75.6%	66.5%	70.7%

Table 3: 不同词向量表征方法对比实验结果

实验三：为了验证选取不同阈值时是否会影响模型的效果，为抽取到更高质量的汉越双语平行句对提供阈值参数基础，设置了在本文提出方法上不同阈值的对比实验，实验结果如表4所示。

不同的阈值M	R(%)	P(%)	F1(%)
M=0.8	77.3%	68.6%	72.7%
M=0.85	75.6%	66.5%	70.7%
M=0.90	73.9%	64.7%	68.9%

Table 4: 不同阈值对比实验结果

从表4中可以看出，不同的阈值M对实验结果的影响。其中，实验设置阈值参数越大，抽取汉越双语平行句对的F1分值反而越低。阈值M作为汉越双语平行句对抽取的判别值。

实验四：为了验证本文方法抽取出的汉-越平行句对对神经机器翻译模型性能的影响。本文选择了目前比较主流的神经网络模型Seq2seq+Attention(Vaswani et al., 2017)作为机器翻译模型，编码器和解码器的单词嵌入和循环状态的维度都设置为512，训练20个epochs，其中句对的批量大小为64。我们挑选了10万条汉-越平行句对作为基础训练集，并与添加了从可比语料中抽取的5万平行句对作对比，表5显示了不同规模数据的BLEU得分。从表5可以看出，在训练集中添加本文系统抽取到的5万平行句对后，翻译系统的BLEU得分分为15.89，提高了0.34，优于直接利用10万平行句对训练的翻译模型。实验结果证实了本文模型抽取到的汉越平行句对的质量，表明了可比语料库中存在大量语义空间相近的汉越平行句对。

数据规模	BLEU
100k	15.45
+（抽取50k）	15.89(+0.34)

Table 5: 平行句对对神经机器翻译性能的影响

7 结论

针对汉-越神经机器翻译数据稀缺的问题，本文提出了一种基于跨语言预训练及Bi-LSTM方法抽取汉越双语平行句对。在没有大规模汉越平行语料情况下，该方法利用汉越种子词典进行汉越跨语言预训练，将汉越双语表征到同一语义空间中，实现语义对齐。利用深度神经网络Bi-LSTM和CNN分别提起汉越句对的上下文信息和局部信息从而抽取出匹配度更高，噪声更小的汉越双语平行句。实验结果表明，经过跨语言预训练的平行句提取方法在准确率和召回率上高于基线模型，并且抽取到的汉-越平行句的语义更近。在未来的工作中，我们会探索该模型用于多语言平行句对抽取且在机器翻译的效果。

参考文献

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2016. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Andoni Azpeitia, Thierry Etchegoyhen, and Eva Martınez Garcia. 2018. Extracting parallel sentences from comparable corpora with stacc variants. In *Proceedings of the 11th Workshop on Building and Using Comparable Corpora*, pages 48–52.

Alberto Barrón-Cedeño, Cristina España-Bonet, Josu Boldoba, and Lluís Màrquez. 2015. A factory of comparable corpora from wikipedia. In *Eighth Workshop on Building And Using Comparable Corpora*.

Houda Bouamor and Hassan Sajjad. 2018. H2@ bucc18: Parallel sentence extraction from comparable corpora using multilingual sentence embeddings. In *Proc. Workshop on Building and Using Comparable Corpora*.

Juryong Cheon and K. O. Youngjoong. 2017. Automatically extracting parallel sentences from wikipedia using sequential matching of language resources. *Ieice Transactions on Information And Systems*, E100.D(2):405–408.

Himanshu Choudhary, Aditya Kumar Pathak, Rajiv Ratan Saha, and Ponnurangam Kumaraguru. 2018. Neural machine translation for english-tamil. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2016. Parallel sentence extraction from comparable corpora with neural network features. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2931–2935, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Thomas C. Chuang, Jian Cheng Wu, Tracy Lin, Wen Chie Shei, and Jason S. Chang. 2004. Bilingual sentence alignment based on punctuation statistics and lexicon. In *Natural Language Processing-ijcnlp, First International Joint Conference, Hainan Island, China, March, Revised Selected Papers*.

Cristina España-Bonet, Ádám Csaba Varga, Alberto Barrón-Cedeño, and Josef Van Genabith. 2017. An empirical analysis of nmt-derived interlingual embeddings and their use in parallel sentence identification. *IEEE Journal of Selected Topics in Signal Processing*, 11(8):1340–1350.

William A. Gale and Kenneth Ward Church. 1991. A program for aligning sentences in bilingual corpora. In *Meeting of the Association for Computational Linguistics*.

Jeenu Grover and Pabitra Mitra. 2017. Bilingual word embeddings with bucketed CNN for parallel sentence extraction. In *Proceedings of ACL 2017, Student Research Workshop*, pages 11–16, Vancouver, Canada, July. Association for Computational Linguistics.

Francis Grégoire and Philippe Langlais. 2017. A deep neural network approach to parallel sentence extraction.

Francis Grégoire and Philippe Langlais. 2018. Extracting parallel sentences with bidirectional recurrent neural networks to improve machine translation.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*.

Benjamin Marie and Atsushi Fujita. 2017. Efficient extraction of pseudo-parallel sentences from raw monolingual data using word embeddings. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*.

Hitoshi Isahara Masao Utiyama. 2013. A comparison of pivot methods for phrase-based statistical machine translation. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *Computer Science*.

Dragos Stefan Munteanu and Daniel Marcu. 2005. Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics*, 31(4):477–504.

Li Peng, Maosong Sun, and Xue Ping. 2010. Fast-champollion: A fast and robust sentence alignment algorithm. In *COLING 2010, 23rd International Conference on Computational Linguistics, Posters Volume, 23-27 August 2010, Beijing, China*.

Sadaf Abdul Rauf and Holger Schwenk. 2011. Parallel sentence generation from comparable corpora for improved smt. *Machine Translation*, 25(4):p.341–375.

Christoph Tillmann and Jian-ming Xu. 2009. A simple sentence-level extraction algorithm for comparable data. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 93–96, Boulder, Colorado, June. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

基于子词级别词向量和指针网络的朝鲜语句子排序

闫晓东
中央民族大学
信息工程学院
国家语言资源监测
与少数民族语言中心
yanxd3244@sina.com

解晓庆
中央民族大学
信息工程学院
国家语言资源监测
与少数民族语言中心
xqplex@yeah.net

摘要

句子排序是多文档摘要系统和机器阅读理解中重要的任务之一，排序的质量将直接影响摘要和答案的连贯性与可读性。因此，本文采用在中英文上大规模使用的深度学习方法，同时结合朝鲜语词语形态变化丰富的特点，提出了一种基于子词级别词向量和指针网络的朝鲜语句子排序模型，其目的是解决传统方法无法挖掘深层语义信息问题。本文提出基于形态素拆分的词向量训练方法(MorV)，同时对比子词n元词向量训练方法(SG)，得到朝鲜语词向量；采用了两种句向量方法:基于卷积神经网络(CNN)、基于长短时记忆网络(LSTM)，结合指针网络分别进行实验。结果表明本文采用MorV和LSTM的句向量结合方法可以更好地捕获句子间的语义逻辑关系，提升句子排序的效果。

关键词： 词向量；形态素拆分；指针网络；句子排序

Korean Sentence Ordering Based on Sub Word Level Word Vector and Pointer Network

Xiaodong Yan
Minzu University of China
National language resource
monitoring &Research Center
Minority Languages Branch
yanxd3244@sina.com

Xiaoqing Xie
Minzu University of China
National language resource
monitoring &Research Center
Minority Languages Branch
xqplex@yeah.net

Abstract

Sentence sorting is one of the most important tasks in multi document summarization system and machine reading comprehension. The quality of sorting will directly affect the coherence and readability of abstracts and answers. Therefore, this paper adopts the deep learning method which is widely used in both Chinese and English, combined with the characteristics of the rich morphological changes of Korean words, puts forward a Korean sentence ordering model based on the sub word level word vector and pointer network, the purpose of which is to solve the problem that traditional methods can not mine deep semantic information. In this paper, a morpheme split based word vector training method (morv) is proposed, and the Korean word vector is obtained by

comparing the sub word n-ary word vector training method (SG). Two sentence vector methods are used: convolution neural network (CNN) and long-term memory network (LSTM), combined with pointer network. The results show that the combination of morv and LSTM can better capture the semantic logic relationship between sentences and improve the effect of sentence ordering.

Keywords: Word vector , Morpheme split , Pointer network , Senternce ordering

1 引言

句子排序是多文档自动摘要任务和阅读理解答案融合的关键技术。在多文档自动摘要任务中，对文摘句子进行排序是一项关键任务，其效果直接影响最后生成的摘要的可读性。在阅读理解的答案排序过程中，也涉及到句子排序问题，其最终结果也会决定答案的可读性。

朝鲜语是我国具有文字的少数民族语言之一，在朝鲜语信息化处理的过程中(Bi, 2011)，同样也有多文档自动摘要和阅读理解答案融合任务。因此朝鲜语句子排序也是一个值得关注的问题。本文结合朝鲜语的特点，提出了基于子词级别词向量的朝鲜语句子排序模型，可以增强句子语义逻辑关系的捕获能力，进而获取句子的合理排序。为后续的朝鲜语多文档自动摘要、朝鲜语机器阅读理解等任务提供一些基础。

通常，在一个文本段落中，语义的连贯性是通过句子的顺序来保证的。对于句子排序问题，前人已经做了大量的工作：徐永东提出了一种多文档摘要中基于时间信息的句子排序方法，利用基于规则的时间信息抽取、语义计算及时序推理方法来解决句子排序问题(Xu et al., 2009)；姚超提出了一种基于内聚度的多文档文摘的句子排序方法，通过将相同话题的句子聚合到一起，避免话题中断，改善文摘可读性(Yao et al., 2006)；薛涛将条件熵引入到句子排序工作中，通过在源文档中计算句子对的转移信息量来衡量句子的关联程度，同时提出上下文对比算法来加强句子邻近度学习的准确性(Xue and Wang, 2017)；郭红建将潜在语义分析聚类算法引入文摘句子排序过程中，将话题聚类之后采用模板对文摘句子进行两趟排序(Guo and Huang, 2013)……

但是，随着大数据、云计算等技术的发展，深度学习方法在自然语言处理任务中广泛应用，很多深度学习方法被引入到句子排序中。康世泽利用神经网络将几种前人提出的句子排序方法融合，并在此基础上提出了一种基于马尔科夫随机游走模型的句子排序算法(Kang et al., 2016)。Chen尝试了基于卷积神经网络(convolutional neural netw-orks, CNNs)，长短期记忆网络(long short-term memory network, LSTM)的句子排序方法，使用CNN、LSTM等模型判断句子的前后句关系，并利用集束搜索算法求解句子的最优排序(Chen et al., 2016)。Logeswaran提出了一种基于循环神经网络的句子排序方法，通过判断句子在每个位置的可能性，求得最优排序结果(Logeswaran et al., 2016)。Gong提出了一种基于端到端的指针网络的句子排序方法，通过端到端的指针网络判断每个位置上的句子的可能性，求得较优排序结果(Gong et al., 2016)。

本文的主要贡献如下：

1）对朝鲜语句子排序问题进行研究；

2）将同形异义词信息融入到朝鲜语词向量的训练；

3）使用形态素和子词级别n元进行词向量训练，并对比效果；

4）使用两种词向量训练方法得到词向量，再使用两种不同的句向量训练方法得到句向量，最后进行句子排序实验，并对比效果。

2 朝鲜语句子排序模型

2.1 任务描述

在机器阅读理解的答案融合任务和多文档自动文摘任务中，候选的句子集是从不同的文档中抽取的，因此，无法根据句子在原文中的位置或者一些显式的连接词对乱序的句子集合进行排序。句子排序任务要解决的问题就是把一组乱序的句子，排列成连贯、通顺的段落。设给定一组乱序的句子$S = s_1, s_2, \cdots, s_n$，句子排序的任务目标是将其排列成顺序$o^*$，对于顺序$o^*$有：

$$s_{o_1^*} > s_{o_2^*} > \cdots > s_{o_n^*} \tag{1}$$

在给定句子集S的情况下，顺序o^*的概率$P(o^*|S)$大于其他任何顺序的概率，可以表示为式(2)。其中o表示句子集S的任一种排序，而Ψ表示句子集S的所有可能的排序的集合。

$$P(o^*|S) > P(o|S), \forall o \in \Psi \tag{2}$$

2.2 模型架构

我们采用指针网络模型(Pointer Network)(Vinyals et al., 2015)对句子集S进行排序。指针网络(Pointer Network)是Nallapati等(Nallapati et al., 2016)提出的基于注意力机制的序列到序列模型的一个变种。它不是把一个序列转换成另一个序列，而是产生一系列指向输入序列元素的指针。最基础的用法是对可变长度序列或集合的元素进行排序，也适用于句子排序问题。

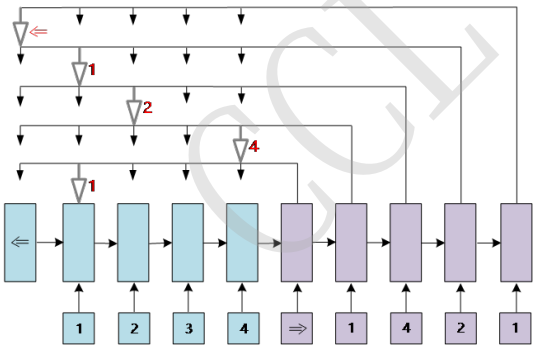


图 1: 指针网络模型结构

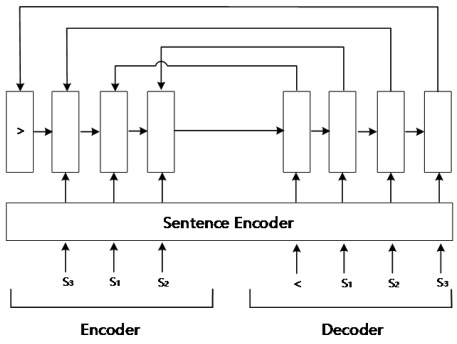


图 2: 基于指针网络的句子排序模型

指针网络模型非常简洁如图1所示，结构是基本的seq2seq+attention。基于指针网络的句子排序模型的架构如图2所示。以顺序o为集合S排序的概率$P(o|S)$的计算公式为(3)。

$$P(o|s) = \prod_{i=1}^{n} P = (o_i|o_{i-1}, \cdots, o_1, s) \tag{3}$$

概率$P = (o_i|o_{i-1}, \cdots, o_1, s)$可以通过指针网络计算，为式(5)，(6)，其中$e_j$，$d_i$分别是指针网络编码端和解码端的输出。

$$P = (o_i|o_{i-1}, \cdots, o_1, s) = softmaxt(u^i)_{o_i} \tag{4}$$

$$u_j^i = v^T \tanh(W_1 e_j + W_2 d_i) \tag{5}$$

2.2.1 编码端

指针网络的编码器模型可以表示为式(6)，其中，$Enc(s_{o_j})$表示句子so_j的编码。

$$e_j = LSTM(Enc(s_{o_j}, e_{j-1}), j = (1, \cdots, n) \tag{6}$$

2.2.2 解码端

指针网络的解码器模型可以表示为式(7)，其中，$Enc(s_{o_i})$表示句子s_{o_i}的编码。

$$d_i = LSTM(Enc(s_{o_i}, d_{i-1}), i = (1, \cdots, n) \tag{7}$$

2.3 句子顺序概率

我们将句子集的顺序表示为：$P(o|s)$，将最佳句子顺序表示为$\hat{o}$:

$$\hat{o} = \underset{o}{argmax}\, P(o|s) \tag{8}$$

找到句子集s的最佳顺序是一个NP问题，有两种策略可以用来解决这个问题：贪心算法和集束搜索算法。

2.3.1 贪心算法

贪心算法（Greedy Algorithm）的思想是指，在对问题求解时，总是做出在当前看来是最好的选择，也就是说，不从整体最优上加以考虑，它所做出的选择是在某种意义上的局部最优解。在指针网络的解码阶段，用贪心算法表示顺序$\hat{o} = \hat{o}_1, \cdots, \hat{o}_n$的生成过程可以表示为式(9)。

$$\hat{o}_i = \underset{o_i}{argmax}\, P(o_i|\hat{o}_{i-1}, \cdots, \hat{o}_1, s) \tag{9}$$

2.3.2 集束搜索算法

集束搜索(Beam Search)是一种启发式图搜索算法，通常用在图的解空间比较大的情况下，为了减少搜索所占用的空间和时间，在每一步深度扩展的时候，剪掉一些质量比较差的结点，保留下一些质量较高的结点。这样减少了空间消耗，并提高了时间效率。在求解最优解时，集束搜索算法的每一步总是保留最优的b个候选项。对于第t步来说，每个候选解可以表示为$\hat{o}_1^t = \hat{o}_1, \cdots, \hat{o}_t$，其概率为式(10)。其中概率最靠前的$b$个候选项将会在第$t$步被保留。

$$P(\hat{o}_1^t|s) = \prod_{i=1}^{t} P(\hat{o}_i|\hat{o}_{i-1}, \cdots, \hat{o}_1, s) \tag{10}$$

3 模型训练

3.1 词向量训练

在自然语言处理的发展过程中，单词的分布式表示不断发展。世界各国的研究学者提出了许多模型。这些模型大多数应用于英语，通过不同的向量来表示词汇表中的每个单词，但会忽略单词的内部结构的变化。不同于英语，对于形态丰富的语言，例如朝鲜语，很多词语在训练语料库中很少出现（或根本没有出现），这使得学得的词向量语义捕获能力差。

朝鲜语句子由多个语节构成，而每个语节(eojeol)由一个或多个形态素组成。其中语节是朝鲜语中的一个分写单位，而形态素则是具有意义的最小语言单位。例如，图3的句子中共有5个语节，其中每个语节由一个或多个形态素构成，图中以“+”作为形态素的分隔符。若仅仅通过语节来训练词向量，那么由于朝鲜语的词尾形态变化丰富，使得训练得到的词向量的语义表示能力不足。为了解决这一问题，本文将采取以下两种朝鲜语的词向量训练方法：1）先将语节拆分成多个形态素（变换原形）的组成形式，再对拆分好的形态素进行词向量训练；2）以朝鲜语子词（音节和字母）为单位，用skip-gram模型训练词向量。上述两种方法都考虑了朝鲜语的形态信息，训练得到的词向量语义表达能力更强。

3.1.1 形态素词向量(Morpheme Vector, MorV)

在朝鲜语中，一部分形态素在句子中的写法与原形之间存在差异。例如，개발했다（实际写法）⇒개발+하+었+다（形态素原形）。可以看到，在形态素分析过程中，“했”形态素转化为“하”，“었”这是因为“했”（表示已经做完）属于缩略语，其中包括了词干信息“做”和时态信息“已经”。解决这一问题的常用方法是利用语料库建立形态素变形词典，并利用词典完成形态素原形恢复。然而基于词典的形态素原形恢复方法受限于语料库质量，存在处理不好的未登录词等问题。针对这一问题，本文采用结合词性信息的多任务seq2seq模型来解决这一问题。在朝鲜语中，由空格分开的单元是语节。由于朝鲜语所有的语节数量非常庞大，实验中用到的语料中有624,655个不同的语节，不太适合直接作为seq2seq模型的输入。本文考虑将一个语节看作一个音节序列。例如语节‘개발했다’是4个音节‘개’，‘발’，‘했’，‘다’组成，可以看作一个音节序列。同样的，形态素也是由音节组成的，也可以看作一个音节序列。本文中实验用到的语料中的音节数量为5,245个，远小于语节的数量。因此，文本以音节为单元作为seq2seq模型的输入，进行朝鲜语形态素拆分模型训练。

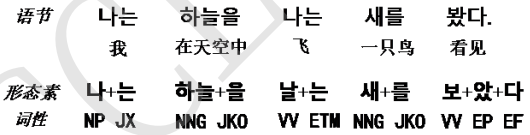


图 3: 朝鲜语句子中的语节和形态素

由于朝鲜语的形态素和词性息息相关(Song and Park, 2019)，例如语节“하늘을（在天空中）”可以被分为两个形态素“하늘（名词）”和“을（宾格助词）”，通过词性就可以把这个语节拆分成两个形态素。本文利用21世纪世宗计划语料库，在训练模型的过程中加入词性信息，可以更准确地进行形态素的拆分。此外，朝鲜语中还存在同形异义词。例如，在图2中语节“나는”出现了两次，意思截然不同，但通过词性可以识别出这是两个不同的词语。本文通过词性的不同将表达两种意思的“나는”，分别记为“나는__01”，“나는__02”，如果还有其他意思就顺次编号：“나는__03”，“나는__04”……这样可以把同形异义词当作不同的词来进行训练。用通过这种方法得到的形态素来训练词向量，可以得到语义表示能力更强的词向量。模型的示意图如图4所示。

本文使用的seq2seq模型是基于2018年Anastasopoulos和Chiang(Anastasopoulos and Chiang, 2018)提出的triangle多任务学习模型，将相互关联的任务放在同一个神经网络中同时训练，可以有效提升训练效果。本文将词性标记和形态素拆分这两个任务同时训练，并将同形异义词

进行编号。通过此模型拆分得到的形态素将更加准确且考虑同形异义信息。

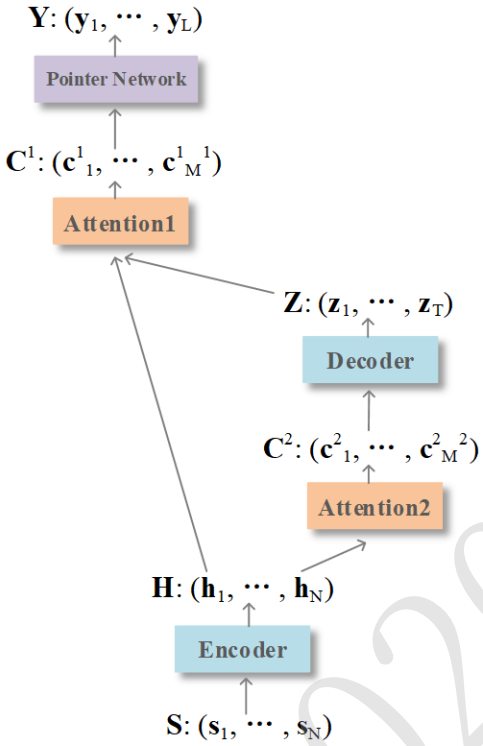


图 4: 朝鲜语句子中的语节和形态素

模型有4个部分：编码模块、解码模块、注意力机制模块1、注意力机制模块2、指针网络模块(Nallapati et al., 2016)。图4所示的是模型的整体框架。编码模块(Encoder)的作用是将输入的音节序列S ：$(s_1, \cdots, s_N)$转化成隐藏状态序列H ：$(h_1, \cdots, h_N)$；注意力机制模块1(Attention1)的作用是将隐藏状态序列H：$(h_1, \cdots, h_N)$转化成考虑上下文信息的隐藏状态序列C^1：$(c_1^1, \cdots, c_{M^1}^1)$，并输入给解码模块(Decoder)；解码模块(Decoder)的作用是将隐藏状态序列C^1：$(c_1^1, \cdots, c_{M^1}^1)$转化成词性标记序列$Z$：$(z_1, \cdots, z_T)$；注意力机制模块2(Attention2)的作用是考虑词性信息的同时，将隐藏状态序列H：$(h_1, \cdots, h_N)$转化成考虑上下文信息的隐藏状态序列C^2：$(c_1^2, \cdots, c_{M^2}^2)$，并输入到指针网络(Pointer Network)；指针网络模块(Pointer Network)的作用是通过softmax函数形成指针，从输入音节序列或给定的音节表中选择音节（或空格），生成形态素序列Y：$(y_1, \cdots, y_L)$。给定输入序列S：$(s_1, \cdots, s_N)$，S表示的是输入的语节，将语节拆分成音节组成的序列，其中s_i表示的是第i个音节，输出序列是Y：$(y_1, \cdots, y_L)$，Y表示的是拆分好的形态素序列，其中Y_i表示的是第i个形态素。

用训练好的模型进行形态素原形转换拆分，朝鲜语的最小单位是形态素。通过形态素原形转换，去除了朝鲜语词尾形态变化丰富这一干扰因素。采用Word2vec进行形态素向量（即词向量）训练，得到关于形态素的分布式表示，具有较强的语义表示能力。

3.1.2 融入子词级别信息(Subword Gram, SG)

形态素拆分过程比较复杂，容易出现错误，提出了基于字母和音节的词向量表示方法(Park et al., 2018)。将一个语节拆分成字母序列，再进行音节级别和字母级别的n元划分。

音节拆分规则:

每个朝鲜语音节可拆分成由3个字母组成的序列，例如“달”可拆分成{ㄷ,ㅏ,ㄹ}。如果有的音节只有两个字母组成，那么就用一个占位符“e”代替第三个字母，例如“해”拆分成{ㅎ,ㅐ,e}。使用“<”作为语节的开始标志，“>”作为语节的结束标志，这样语节“강아지”可以拆分成字母序列{<,ㄱ,ㅏ,ㅇ,ㅇ,ㅏ,e,ㅈ,ㅣ,e,>}。

音节级别的n元划分：

语节“강아지”的一元划分可以表示为：{ㄱ,ㅏ,ㅇ}, {ㅇ,ㅏ,e}, {ㅈ,ㅣ,e}；二元划分可以表示为：{ㄱ,ㅏ,ㅇ,ㅇ,ㅏ,e}, {ㅇ,ㅏ,e,ㅈ,ㅣ,e}；三元划分可以表示为：{ㄱ,ㅏ,ㅇ,ㅇ,ㅏ,e, ㅈ,ㅣ,e}。

字母级别的n元划分：

由于朝鲜语的粘着性，只考虑音节级别的n元，无法捕捉到形态变化信息，因此还需要考虑字母级别。

关于语节“강아지”，字母级别的三元划分可以表示为：{<,ㄱ,ㅏ}, {ㄱ,ㅏ,ㅇ}, {ㅏ,ㅇ,ㅇ}, {ㅇ,ㅇ,ㅏ}, {ㅇ,ㅏ,e}, {ㅏ,e,ㅈ}, {e,ㅈ,ㅣ},{ㅈ,ㅣ,e}, {ㅣ,e,>}。

然后用这两个级别的n元，通过skip-gram方法(Bojanowski et al., 2017; Mikolov et al., 2013)进行词向量训练。我们使用该方法与上文提出的方法均用来训练词向量，将训练得到的词向量再进行下一步处理，最后观察句子排序的结果。

3.2 句向量表示

句向量又可以称为句嵌入(Cer et al., 2018)，句嵌入模型的输入为词向量，输出为表示句子的向量，该向量可以作为具体任务的输入进行预测和训练。自然语言处理的任务大多数都是序列化的信息，序列化信息的特点就是不同时间步上的信息会有交叉作用，如何发掘序列化输入的信息是自然语言处理任务的关键。在当前研究成果中，主要分为两大解决方法：一是以循环神经网络为基础的解决方案；二是以卷积神经网络为基础的解决方案。本文将采用这两种方案对句子进行向量化，并对比不同的句向量训练方法对句子排序结果的影响。

3.2.1 卷积神经网络模型

卷积神经网络(Convolutional neural networks, CNN)(Simard et al., 2003)仿造生物的视知觉机制，包含卷积计算且具有深度结构的前馈神经网络，是深度学习的代表算法之一。将包含n_w个单词的句子s通过卷积神经网络编码的过程可以表示为公式(11)，(12)。其中$W_{cov} \in R^{(dl_f)d_f}$和$b_{cov} \in R^{d_f}$是可训练的参数，其中$\phi(\cdot)$是tanh函数。$k = 1, \cdots, n_w - l_f + 1$。其中的$l_f$和$d_f$都是卷积神经网络模型中的超参数，分别是过滤器(filter)的长度和特征图(feature map)的个数。

$$cov_k = \phi(W_{cov}^T(\oplus_{u=0}^{l_f-1} w_{k+u}) + b_{cov}) \tag{11}$$

$$Enc(s) = \max_k cov_k \tag{12}$$

3.2.2 长短时记忆网络模型

长短时记忆网络(Long short term memory, LSTM)(Hochreiter and Schmidhuber, 1997)是一种特殊的RNN，主要是为了解决长序列训练过程中的梯度消失和梯度爆炸问题。LSTM的存储单元$c \in R^{d_r}$由三种门控制：输入门$i \in R^{d_r}$、遗忘门$f \in R^{d_r}$输出门$o \in R^{d_r}$，表示为公式(13)-(15)。其中，$W_g \in R^{(d+d_r)4d_r}$和$b_g \in R^{4d_r}$是可训练的参数，d_r是表示存储单元和门控单元的维

度的一个超参数。$t = 1, \cdots, n_w$其中$\sigma(\cdot)$是sigmoid函数，$\phi(\cdot)$是tanh函数。

$$\begin{bmatrix} i_t \\ o_t \\ f_t \\ \widetilde{c}_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \phi \end{bmatrix} (W_g^T \begin{bmatrix} w_t \\ h_{t-1} \end{bmatrix} + b_g) \tag{13}$$

$$c_t = c_{t-1} \odot f_t + \widetilde{c}_t \odot i_t \tag{14}$$

$$h_t = o_t \odot \phi(c_t) \tag{15}$$

我们将通过长短时记忆网络编码的句子向量表示为：

$$Enc(s) = h_{n_w} \tag{16}$$

3.3 目标函数训练

设有m个训练样本$(x_i, y_i)_{i=1}^m$，x_i表示的是一个句子集合，这个句子集合有一个唯一特定的排序序列y_i，y_i的句子顺序是最优顺序o^*。为了得到更多的训练数据，本文在训练模型的过程时，在每个epoch中为句子集合x_i随机生成新的排序。目标函数可以表示为公式(17)。其中，$P(y_i|x_i;\theta) = P(o^*|S = x_i;\theta)$，$\lambda$是正则项的超参数。$\theta$表示所有可训练的参数。此外，本文采用AdaGrad(Duchi et al., 2011)结合小批量梯度下降(Turian et al., 2010)优化算法来训练模型。

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} logP(y_i|x_i;\theta) + \frac{\lambda}{2}\|\theta\|_2^2 \tag{17}$$

4 实验

4.1 数据集

本文从延边日报朝鲜语版、人民网朝鲜语版等新闻网站爬取了20000篇朝鲜语新闻作为语料。将每篇新闻进行语段分隔，选取句子数目大于2的语段作为一个数据单元，将每个数据单元的句子进行打乱编号。例如将语段[s1，s2，s3，s4]编码为[4,1,2,3]，然后再对该语段编码随机打乱为[3,2,4,1]。这样我们就得到一个训练样本([句1,句2,句3,句4]，[4,1,2,3]，[3,2,4,1])，第一项为顺序句子集合，第二项为正确顺序，第三项为乱序顺序。按照上述形式对所有数据单元进行编码再打乱，得到样本集合。对这些样本集合进行训练集、验证集和测试集的划分。划分结果如表1所示：

Models	PM	LSR	PMR
新闻类型	训练集	验证集	测试集
经济	19,223	2,465	2,497
政治	15,495	1,943	1,866
科技	821,795	102,584	102,892
体育	84,689	10,624	10,453
教育	13,273	1,619	1,695
娱乐	5,201	708	670
法律	216,153	26,819	26,854

表 1: 实验所用语料

Initial learning rate	$\alpha = 0.5$
Regularization	$\lambda = 10^{-5}$
Hidden layer size of Ptr-Net	h=200
Filter length of CNN	l_f=3,4,5
Number of features maps	d_f=128
Hidden size of LSTM	d_r=200
Size of embedding	d_e=100
Beam size	b=64
Batch size	128

表 2: 超参数设置

4.2 超参数设置

表2展示了上述模型中的超参数的设置。卷积神经网络的句子编码模型使用了3种不同长度l_f的过滤器(Kim et al., 2016)。

4.3 评测方法

本文采用了3中不同的模型评测方法：(1)成对度量法；(2)最长序列比法；(3)最佳匹配比法。

4.3.1 成对度量法

成对度量法(Pairwise metrics, PM)指的是，预测的相对顺序与原本真正顺序相同的句子对的分数越高越好。成对度量法可以表示为三个量化分数：精确值P、召回率R和F值，如公式(18)-(20)所示。其中，函数$S(\cdot)$表示一段文本中所有句子对的集合，绝对值符号表示的是集合的大小。

$$P = \frac{1}{m}\sum_{i=1} m\frac{|S(\hat{o}_i) \cap S(o_i^*)|}{|S(\hat{o}_i)|} \tag{18}$$

$$R = \frac{1}{m}\sum_{i=1} m\frac{|S(\hat{o}_i) \cap S(o_i^*)|}{|S(\hat{o}_i^*)|} \tag{19}$$

$$F = \frac{2 * P * R}{P + R} \tag{20}$$

设$\{\hat{o} = (2,3,1,4), o^* = (1,3,4)\}$，其中第二句话是一个噪声项。对于这个例子，成对度量分数可以表示为：$P = 1/6, R = 1/3, F = 2/9$。

4.3.2 最长序列比法

最长序列比法(Longest sequence ratio, LSR)计算最长正确子序列的比（不需要连续性，越高越好）。最长序列比法可以表示为三个分数：精确值P、召回率R和F值，如公式(21)-(23)所示。其中，函数$L(\cdot)$表示的是最长正确子序列中元素的个数。那么，$L(\hat{o} = (2,3,1,4), o^* = (1,3,4))$的值就是2。

$$P = \frac{1}{m}\sum_{i=1}^{m} \frac{|L(\hat{o}_i, o_i^*)|}{|\hat{o}_i|} \tag{21}$$

$$P = \frac{1}{m}\sum_{i=1}^{m}\frac{|L(\hat{o}_i, o_i^*)|}{|o_i^*|} \tag{22}$$

$$F = \frac{2 * P * R}{P + R} \tag{23}$$

4.3.3 最佳匹配比法

最佳匹配比法(Perfect match ratio，PMR)计算的是确切的匹配项的比例，如公式(24)，(25)所示。其中，$P(\cdot)$表示$\hat{o}_i$和o_i^*的最佳匹配子序列的长度。

$$PMR = \frac{1}{m}\sum_{i=1}^{m}1\{\hat{o}_i = o_i^*\} \tag{24}$$

$$\{\hat{o}_i = o_i^*\} = \frac{P(\hat{o}_i \cap o_i^*)}{\hat{o}_i} \tag{25}$$

4.4 实验结果和分析

我们用两种不同的词向量训练方法，两种不同的句向量训练方法对句子进行编码，然后通过指针网络进行句子排序，在进行句子排序的过程中，使用两种不同的搜索算法：贪心算法和集束搜索算法，结果分别用三种评测方法进行评测。结果如下表所示： 根据表3我们可以看出，

Models	PM	LSR	PMR
+greedy algorithm			
MorV+CNN	80.21	74.33	39.12
MorV+LSTM	84.02	78.25	43.68
SG+CNN	78.35	73.28	37.21
SG+LSTM	81.37	77.92	41.69
+beam search			
MorV+CNN	80.68	76.87	40.36
MorV+LSTM	**85.13**	**79.20**	**44.32**
SG+CNN	79.28	76.97	37.49
SG+LSTM	82.63	78.51	43.56

表 3: 不同方法的句子排序结果对比

使用本文提出的形态素拆分模型(MorV)将语节拆分成形态素，再进行词向量训练，在三种评测方法下，可以使得朝鲜语句子排序效果更好。使用LSTM进行句子编码相对于CNN，句子排序效果更好。增加集束搜索(beam search)过程后，句子排序的效果也有所提升。 从图5中也可以直观得出结论：使用MorV词向量训练模型+LSTM句编码模型，句子排序效果最佳。表4给出的是句子排序的实例

5 总结

句子排序是近年来自然语言处理中多文档摘要生成和机器阅读理解答案融合任务中的一个十分重要子任务。以往的研究主要是基于传统的机器学习方法，但随着深度学习方法的不断发展，句子排序任务也可以使用一些深度学习方法来解决。

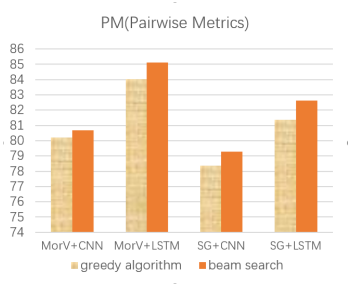


图 5: PM评测结果

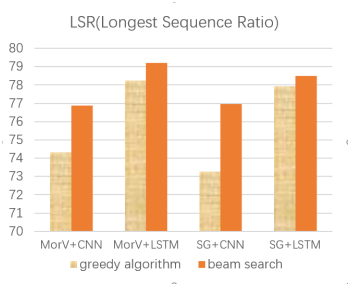


图 6: LSR评测结果

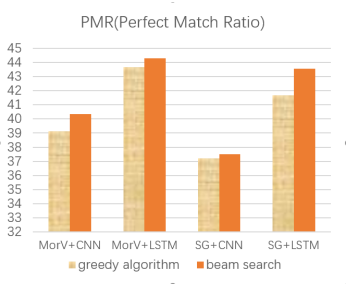


图 7: PMR评测结果

正确顺序语段	도로시는 1910 년에 당시 영국의 식민지였던 이집트의 까히라에서 태여났다. 1928 년에 옥스퍼드대학의 화학과에 진학해 생화학을 전공했다. 그녀는 1932 년에 대학을 졸업했지만 직장을 구할 수 없었다. 도로시는 소개를 통해 케임브리지대학의 화학과 교수를 알게 됐고 그 교수는 다시 버널 밑에서 공부하도록 주선해 주었다. 버널은 엑스선을 리용해 단백질을 비롯한 생물학적 결정을 연구하고 있었다. 。
顺序编码	3，2，5，4，1
乱序编码	4，5，1，2，3
排序结果	3，2，5，1，4
排序结果对应语段	도로시는 1910 년에 당시 영국의 식민지였던 이집트의 까히라에서 태여났다. 1928 년에 옥스퍼드대학의 화학과에 진학해 생화학을 전공했다. 그녀는 1932 년에 대학을 졸업했지만 직장을 구할 수 없었다. 버널은 엑스선을 리용해 단백질을 비롯한 생물학적 결정을 연구하고 있었다. 도로시는 소개를 통해 케임브리지대학의 화학과 교수를 알게 됐고 그 교수는 다시 버널 밑에서 공부하도록 주선해 주었다. 。

表 4: 句子排序示例

在朝鲜语信息化进程中，也需要跟上深度学习发展的步伐。本文将深度学习模型用于朝鲜语信息化处理，使用多任务seq2seq模型进行形态素拆分，并且将指针网络用于朝鲜语句子排序，取得了较好的效果。接下来，我们将继续结合朝鲜语本身的特点，继续提高句子排序的效果，并将其用于多文档摘要任务中。

参考文献

Antonios Anastasopoulos and David Chiang. 2018. Tied Multitask Learning for Neural Speech Translation. *Association for Computational Linguistics*. Proceedings of NAACL-HLT 2, New Orleans, Louisiana, 82–91.

Yude Bi. 2011. On the study of Korean natural language processing. *Journal of Chinese Information Processing*, 25(6):166–169. *In Chinese*.

Piotr Bojanowski, Edouard Grave, Armand Joulin and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, Ray Kurzweil. 2018. Universal Sentence Encoder. *ArXiv*. abs/1803.11175.

Xinchi Chen, Xipeng Qiu and Xuanjing Huang. 2016. Neural sentence ordering. *ArXiv*. abs/1607.06952.

John Duchi, Elad Hazan and Yorma Singer. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12:2121–2159.

Jingjing Gong, Xinchi Chen and Xipeng Qiu. 2016. Neural sentence ordering. *ArXiv*. abs/1611.04953.

Hongjian Guo and Bing Huang. 2013. The application of latent semantic analysis clustering algorithm in abstract sentence ordering. *Application Research of Computers*, 30(11):3299–3301. *In Chinese*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Shize Kang, Hong Ma and Ruiyang Huang. 2016. A method of sentence ordering based on neural network model. *Journal of Chinese Information Processing*, 30(5):195–202. *In Chinese*.

Yoon Kim, Yacine Jernite, David Sonta and Alexander M. Rush 2016. Character-aware neural language models. *ArXiv*. abs/1508.06615.

Lajanugen Logeswaran, Honglak Lee and Dragomir Radev. 2016. Sentence ordering using recurrent nerual networks. *ArXiv*. abs/1611.02654.

Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *ArXiv*. abs/1301.3781.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre and Bing Xiang. 2018. Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. *Association for Computational Linguistics*. Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning, Berlin, Germany, 280–290.

Sungjoon Park, Jeongmin Byun, Sion Baek, Yongseok Cho and Alice Oh. 2018. Subword-level Word Vector Representations for Korean. *Association for Computational Linguistics*. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 2429–2438.

Patrice Y. Simard, Dave Steinkraus and John C. Platt. 2003. Best Practices for Convolutional Neural Networks. Seventh International Conference on Document Analysis and Recognition, Edinburgh, UK, 958–963.

Hyun Je Song and Seong Bae Park. 2019. Korean Morphological Analysis with Tied Sequence-to-Sequence Multi-Task Model . *Association for Computational Linguistics*. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Hong Kong, China, 1436–1441.

Joseph Turian, Lev-Arie Ratinov and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. *Association for Computational Linguistics*. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, 384–394.

Oriol Vinyals, Meire Fortunato and Navdeep Jaitly. 2015. Pointer Networks. Advances in Neural Information Processing Systems 28, 2692–2700.

Yongdong Xu, Yadong Wang and Yang Liu. 2009. Research on the strategy of sentence ordering based on time information in multi document summarization. *Journal of Chinese Information Processing*, 23(4):27–33. *In Chinese*.

Tao Xue and Heng Wang. 2017. Research on sentence ordering based on conditional entropy and context proximity. *Application Research of Computers*, 34(9):2680–2684. *In Chinese*.

Chao Yao, Sheng Li and Shu Zhang. 2006. Sentence ordering of multi document abstracts based on cohesion. The academic conference of the 25th anniversary of the Chinese information society. *In Chinese*.

基于统一模型的藏文新闻摘要

闫晓东	解晓庆	邹煜	李维
中央民族大学	中央民族大学	中央民族大学	中央民族大学
信息工程学院	信息工程学院	信息工程学院	信息工程学院
国家语言资源监测	国家语言资源监测	国家语言资源监测	国家语言资源监测
与少数民族语言中心	与少数民族语言中心	与少数民族语言中心	与少数民族语言中心
yanxd3244@sina.com	xqplex@yeah.net	17820314536@163.com	1289773612@qq.com

摘要

Seq2seq神经网络模型在中英文文本摘要的研究中取得了良好的效果，但在低资源语言的文本摘要研究还处于探索阶段，尤其是在藏语中。此外，目前还没有大规模的标注语料库进行摘要提取。本文提出了一种生成藏文新闻摘要的统一模型。利用TextRank算法解决了藏语标注训练数据不足的问题。然后，采用两层双GRU神经网络提取代表原始新闻的句子，减少冗余信息。最后，使用基于注意力机制的Seq2Seq来生成理解式摘要。同时，本文加入了指针网络来处理未登录词的问题。实验结果表明，ROUGE-1评分比传统模型提高了2%。

关键词： 文本摘要；藏文； TextRank；指针网络； Bi-GRU

Abstractive Summarization of Tibetan News Based on Hybrid Model

Xiaodong Yan$_{1,2}$	Xiaoqing Xie$_{1,2}$	Yu Zou$_{1,2}$	Wei Li$_{1,2}$
yanxd3244@sina.com	xqplex@yeah.net	17820314536@163.com	1289773612@qq.com
Minzu University of China$_1$,		Minzu University of China$_1$,	
National language resource		National language resource	
monitoring &Research Center		monitoring &Research Center	
Minority Languages Branch$_2$		Minority Languages Branch$_2$	

Abstract

The sequence-to-sequence neural network model has achieved good results in the task of text rsummarization in Chinese and English, but the research of text summarization in low-resource languages is still in the exploratory stage, especially in Tibetan. What' s more, there is no large-scale annotated corpus for summary extraction. In this paper, a hybrid model is proposed to generate Tibetan news summarization. We use the TextRank algorithm to solve the problem of lacking labeled training data in Tibetan. Then, we take two-layer Bi-GRU neural network to extract the sentences which represent the original news, and reduce redundant information. Finally, the Seq2Seq with attention model is used to generate the abstractive sumarization. Meanwhile, we add the pointer-network to deal with out-of-vocabulary words. The experimental results show that ROUGE-1 score increases by 2% than traditional model.

Keywords: Text Summarization , Tibetan , TextRank , Pointer Network , Bi-GRU

1 引言

随着信息的爆炸式增长，人们难以高效、快速、准确地获取有价值的信息。为了解决这一问题，自动文本摘要技术应运而生，产生了对输入文本的简洁表示。文本自动摘要是自然语言处理领域的一个重要分支。它是一种利用计算机实现文本分析、内容归纳和自动文摘生成的信息压缩技术(Mani and Maybury, 1999)，帮助研究人员分析和总结冗长的文本，过滤掉多余的信息，从而提高浏览文本的速度。

文本摘要在信息检索中得到了广泛的应用，并取得了良好的效果。根据实现方法，文本摘要可以分为两类：抽取式摘要和理解式摘要。抽取式摘要是从原文中选择句子并将其组合起来生成摘要。而理解式摘要是对原文的重新解读而不是摘抄，对原文在语义上进行深层次理解，重新对文本进行表述，更加贴近人为表述方式。但这需要更先进的文本生成技术。由于抽取式摘要比理解式摘要更准确和可读，因此大多数研究都集中在抽取式摘要上(Gambhir and Gupta, 2017)。

随着深度学习技术的发展，基于注意力机制的seq2seq模型在文摘中取得了良好的效果(Rush et al., 2015)。与汉英相比，藏文文本摘要还处于探索阶段，面临着许多困难和挑战。首先，递归神经网络能够很好地对一个句子或一段文本进行编码，但不能很好地对整篇藏文文本进行编码。其次，缺乏大规模的文本摘要标注数据。最后，基于词的理解式摘要可能会出现未登录词的问题，从而影响摘要的可读性。

本文提出了一种将抽取式摘要和理解式摘要相结合的藏文摘要生成统一模型。首先，本文使用双向Bi-GRU神经网络从藏文新闻中提取句子。其次，将指针网络融入到基于注意力的seq2seq模型中，生成摘要。与其他模型相比，该模型能够有效地生成藏文摘要。

本文的主要贡献如下：

1) 提出了一个统一模型，它同时利用了抽取式和理解式的摘要方法。使用两层神经网络来提取能够表达原始语义的句子。采用基于注意力机制的seq2seq模型生成摘要，解决了藏文新闻篇幅过长的问题；

2) 引入文本秩算法对抽取的训练语料进行标记，作为神经网络模型的输入。它可以解决藏文标注语料库不足的问题；

3) 利用指针网络提高了藏文未登录词的处理精度，增加了摘要的可读性和新颖性。

2 相关工作

本文首先介绍抽取和理解式摘要的相关工作，然后介绍藏文文本摘要的相关工作。

IBM的Luhn首先提出了基于词频和分布的句子评分模型来提取“自动摘要”，这是机器生成的提取摘要的第一个例子(Luhn, 1958)。摘要抽取的目的是抽取句子来概括文章的中心思想。这些句子被称为关键句，它们是通过分析词频、标题、位置、句法结构、线索词等获得的。传统的提取算法大致分为四类：(1)基于统计的方法。句子的权重是根据词频、位置等信息计算出来的，然后按降序排列。权重值最高的句子被确定为摘要。这种方法的提取速度快，但不能提取

句子的内部信息，导致摘要质量差(Brandow, 1995)。(2)基于图的方法。将文章转化为拓扑图，通过递归和迭代运算使句子权重稳定。对句子进行排序和加权，选择权重最大的句子作为总结，例如TextRank和LexRank算法(Mihalcea and Tarau, 2004)。(3)基于文档主题的方法，利用主题模型提取隐藏信息，例如LDA算法(Sun, 2017)。(4)基于整数规划的方法。它通过将抽取的摘要转化为整数线性规划来寻找全局最优解(Xie, 2011)。目前，随着大数据、云计算等技术的发展，深度学习方法在NLP任务中取得了良好的效果，尤其是在文本摘要方面。SummaRuNNer是一个典型的文本过滤网络(Nallapati et al., 2017)，它将句子抽取问题转化为二分类问题。在英语语料库中，ROUGE-1的得分达到39.6%。Yin等人提出了一种新的基于CNN的网络语言模型（CNNNLM），将句子表示为一个密集向量进而计算句子冗余度，ROUGE-1评分达到42.3%(Yin and Pei, 2015)。Cheng等人使用基于注意力机制的LSTM对每个句子进行分类(Cheng and Lapata, 2016)，在长文本中，ROUGE-1得分达到33%。

随着语料库的不断扩展，机器学习方法被应用于抽象文摘中。传统的统计方法可分为三类：(1)朴素贝叶斯模型，它将朴素贝叶斯分类器与自动摘要结合起来(Chopra et al., 2016)。(2)隐马尔可夫模型，它将隐马尔可夫模型与自动摘要结合起来(Nallapati et al., 2016)。(3)将条件随机场与自动摘要结合起来的概率图模型(See et al., 2017)。同时，这种深度学习方法也取得了较好的效果。2015年，Rush等人使用序列到序列模型（Seq2Seq）和注意力机制生成文本摘要(Rush et al., 2015)。模型采用了编解码框架。编码器使用LSTM网络嵌入句子，解码器使用RNNLM生成摘要。在DUC-2004和Gagiword数据集中，ROUGE-1得分达到28.18%。但是嵌入层无法学习到深层的语义信息。为了解决这个问题，Sumit等人改进了模型，在编码器层，CNN被用来压缩字符作为GRU-RNN的输入。然后在解码层使用RNN(Chopra et al., 2016)。ROUGE-1得分达到32.75%。但是，摘要中的单词都来自词汇表，总是不断重复。2018年，谷歌推出了指针网络来解决词汇表外（OOV）问题(See et al., 2017)，它指向源文本并复制词汇表中没有出现的单词。此外，它还使用了覆盖机制来跟踪摘要的内容，从而减少重复。ROUGE-1得分达到39.53%。

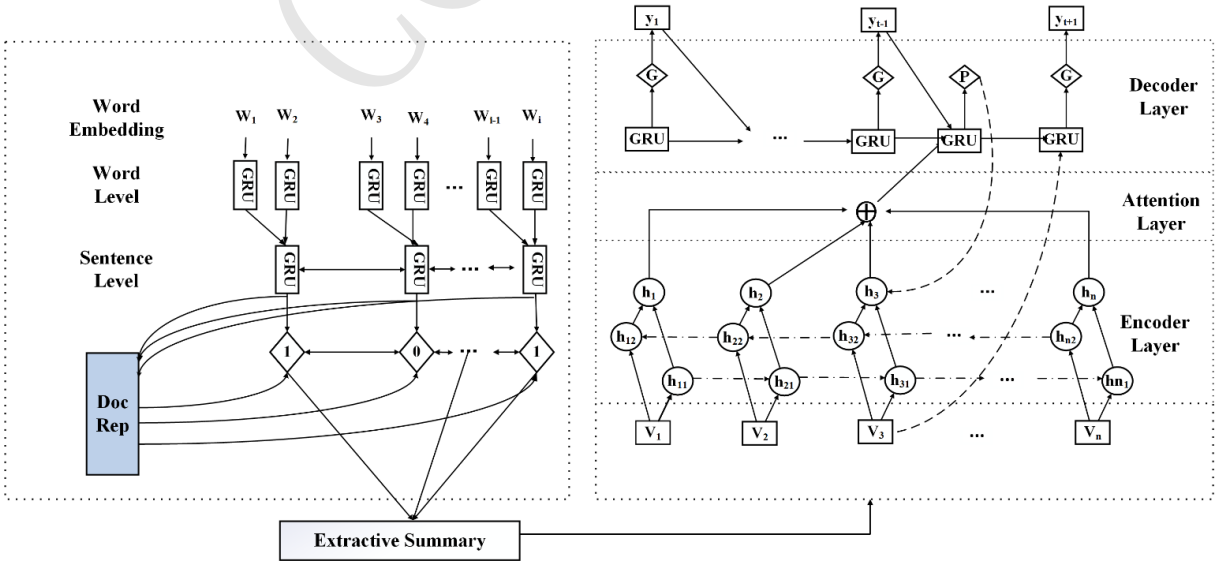


图 1: 统一模型示意图

由于缺乏大规模的训练语料库，目前对藏文文本摘要的研究较少。安见才让对藏文新闻进行了爬取过滤，提取了词频、标题、位置、句法结构、指示短语等五个特征作为权重(Anjian, 2010)。选取权重最大的句子作为新闻摘要的抽取。在此基础上，南奎娘若融合了分词、词性标注、缩略词、句子边界识别、停词选句等功能(Nankui and Anjian, 2016)。除此之外，在藏语中，没有提取摘要的基线，也没有关于提取摘要的研究。

3 模型架构

本文提出的模型架构如图1所示。该模型不仅采用TextRank算法构造了抽取式摘要的训练语料库，而且还训练了一个双层Bi-GRU网络来抽取藏文新闻中的句子。然后将提取出的句子输入seq2seq模型，根据注意力机制和指针机制生成摘要。模型主要由三部分组成:

1) 采用TextRank算法解决了低资源语言训练语料库不足的问题，并利用外部知识库对藏文新闻进行了标注。然后，在对TextRank算法进行迭代后，得到一个可用于训练抽取型网络的训练语料库。

2) 利用标注的语料库训练双层Bi-GRU抽取型网络，如图1左图所示。对于第一层，它用于获取字级信息。第二层则是从句子层面获取信息，获取藏文新闻中的文献信息。最后，根据新闻的文档表示、提取的摘要和隐藏层的状态来确定当前语句是否标记为1或0。

3) 以藏语句子1作为理解型模型的输入。如图1右图所示，理解型模型的总体架构采用seq2seq，编码端采用Bi-GRU，解码端采用RNN。为了获取关键信息，在解码端引入了注意力机制，结合指针网路解决了OOV问题。

4 模型描述

4.1 基于改进TextRank的摘要抽取

TextRank算法是PageRank算法的变体，PageRank算法是一种链接分析算法。谷歌用它来分类和估计网页的价值。它通常用于有向图中，并按指向前驱和后继的边数迭代。迭代运算如公式(1)所示。

$$S(V_i) = (1-d) + d \times \sum_{V_j \in In(V_i)} \frac{1}{|Out(V_j)|} \cdot S(V_j) \tag{1}$$

其中$S(V_i)$表示节点V_i的权重。$In(V_i)$是节点的入度，即指向此网页的URL数。$Out(V_j)$表示节点的出度数。d是阻尼系数，通常取0.85 。

本文将此思想应用于文本摘要的提取。根据句子的关联图确定句子的相对重要性。传统的算法大多忽略了文档的语义和语法信息。只把新闻看作是一个独立的词的集合，而没有考虑词与词之间的联系。将语料库等外部知识融入到文本摘要算法中，提高了算法的准确性。具体方法如下。首先，TextRank算法根据藏文新闻生成拓扑图，表示为$G = (V, E)$ 。 G表示无向图，其中V是顶点集，即新闻中的句子。E是一组边，表示句子之间的关系。本文使用TextRank算法迭代图模型直到收敛。然后每个顶点都有一个表示句子重要性的分数。分数最高的句子被提取出来作为摘要。该过程主要分为四个步骤:

1) 在对藏文新闻句子进行分割后，将每一个句子作为节点添加到图模型中;

2) 句子的矢量表示是同一维度上所有词矢量的平均值。边表示句子之间的相似度，如公式(2)所示。

$$WS(S_i, S_j) = \cos(S_{i1} \cdots S_{im}, S_{j1} \cdots S_{jm}) \tag{2}$$

其中S_i和S_j是句子向量，cos是句子S_i和S_j之间的余弦距离，n表示单词向量的维数。本文还比较了不需要用高维向量表示的共现矩阵计算的相似度。两个句子中同时出现的词的平均权重用作边的权重。

3) 迭代算法直到收敛，如公式(3)所示。

$$WS(V_i) = (1-d) + d \times \sum_{V_j \in In(V_i)} \frac{W_{ij}}{\sum_{V_k \in Out(V_j)} W_{kj}} \cdot WS(V_j) \tag{3}$$

其中W_{ij}表示节点V_i和V_j之间的边的权重，该权重由相似度表示。$In(V_i)$表示指向的V_i节点，$Out(V_j)$表示从V_j指向其他节点的节点。在TextRank算法中，所有节点的初始得分一般为1，当某个节点的误差小于0.0001时，迭代停止。

4) 根据收敛得分对节点进行排序。与标题相似的句子更有可能是摘要，因此在本文提出的改进模型中适当增加了这些句子的权重。在生成的藏文摘要中，几个权重较大的句子的相似度一般很大。本文引入惩罚系数以避免摘要中的句子冗余问题。摘要相似度高的句子乘以惩罚系数以降低权重。

4.2 基于Bi-GRU的文摘抽取

经过TextRank算法处理后，藏文新闻文章可以表示为一个由0和1组成的向量，这个向量的维数就是句子的个数。0表示句子未被选中，1表示摘要被选中。这样，句子抽取问题就可以概括为序列标记问题。递归神经网络（RNN）能很好地求解序列数据。但是，由于后面的节点对前面节点的感知度较低，本文使用了一种称为GRU的RNN变体。GRU由一个更新门和一个重置门组成。更新门决定将以前的内存保存到当前状态的程度，重置门决定如何将新信息与以前的信息融合。在时刻t，本文根据传输状态h_{t-1}和电流输入x_t得到两个门控状态。r_t是重置门，z_t是控制更新状态的门。更新和重置规则如式(4)-(5)所示。

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \tag{4}$$

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \tag{5}$$

其中σ是sigmoid函数，其目的是将输入数据转换为0-1的范围。W_r和W_z是训练好的参数。在GRU的隐藏层中，先前的状态h_{t-1}被重置并用x_t拼接。使用激活函数tanh得到的输出h_t。然后，使用更新门z执行遗忘和存储选择，如公式(6)-(7)所示。最后，得到新的隐藏状态h_t和输出y_t，如公式(8)所示。

$$h_t = \tanh(W_h \cdot [r_t \times h_{t-1}, x_t]) \tag{6}$$

$$h_t = (1-z_t) \times x_t + z_t \times h_t \tag{7}$$

$$y_t = \sigma(W_y \cdot h_t) \tag{8}$$

其中y_t是值为0或1的句子的标签。单向GRU只能获得一个方向的信息，而双向GRU(Bi-GRU)可以连接隐藏层中的前向和后向传播状态。

本文采用两层Bi-GRU来更好地提取深层语义特征。第一层获得字级信息。执行最大池操作，将每个句子中单词的隐藏状态作为第二层句子单元的输入。第二层得到藏文新闻d的句子级信息和文档表示，如式(9)所示。

$$d = \tanh(W_d \frac{1}{N_d} \sum_{j=1}^{N^d} [h_j^f, h_j^b] + b) \tag{9}$$

其中h_i^f和h_i^b是句子的前向和后向隐藏层状态，N_d是文档中句子的个数，矩阵W_d和偏置b是可训练的参数。

在分类过程中，模型根据文档表示、隐藏层状态、位置信息、生成摘要四个方面共同决定句子是否被选中，如公式(10)所示。

$$P(y_j = 1|h_j, s_j, d) = \sigma(W_c h_j + h_j^T W_s d - h_j^T W_r \tanh(s_j) + W_{ap} p_j^a + W_{rp} p_j^r + b) \tag{10}$$

其中W_c，W_s，W_r，W_{ap}，W_{rp}和b是需要训练的参数，y_j表示是否选择此句子作为摘要，h_j是表示句子级网络隐藏状态的输出，d是经过非线性变换后的文本表示，s_j是位置j^th的动态摘要表示，p_j^a是绝对的位置向量，p_j^r是相对位置向量。减法运算用于删除冗余信息。

4.3 基于指针网络的文摘

1) 基于注意力机制的seq2seq 序列到序列模型结合了两个递归神经网络。一个负责接收提取的句子；另一个负责根据前一个网络的隐藏状态生成藏文新闻摘要，分别称为编解码过程。编码过程实际上是利用RNN的记忆功能，根据上下文的顺序关系，将字向量按顺序输入网络，并保留最后的隐藏状态。它同样可以压缩整个句子并将其存储为上下文向量。

在解码过程中引入了注意力机制来分配序列的权重。通过加权变换提高了精度。如公式(11)-(12)所示生成摘要。

$$s_r = f(y_{t-1}, s_{t-1}, c) \tag{11}$$

$$Y_i = softmax(S_t) \tag{12}$$

其中Y_i表示生成的藏文摘要的i^th单词，由三个状态确定：y_{i-1}，s_i，c_i。s_i表示时刻i的隐藏状态，该状态由c_i，s_{i-1}，y_{i-1}决定。c_i表示注意加权的内容向量，其内容向量c_i如式(13)-(15)所示。

$$c_i = \sum_{j=1}^{T_x} \alpha_{i,j} h_j \tag{13}$$

$$e_{i,j} = a(s_{i-1}, h_j) \tag{14}$$

$$\alpha_{i,j} = \frac{exp(e_{i,j})}{\sum_{k=1}^{T_x} exp(e_{i,k})} \tag{15}$$

其中$e_{i,j}$表示解码器隐藏层状态s_{i-1}和编码器隐藏层状态h_j的线性组合，$\alpha_{i,j}$表示通过注意机制学习的每个单词的权重。

2) 指针机构

本文在解码层采用指针网络来解决藏文OOV问题。指针机制是注意力机制的变体。它用于确定目标单词y_t是由词汇表中的RNN选择的，还是通过设置指针开关直接从输入文本中复制的。当选择P模式时，解码器从输入句子中复制单词。当选择G模式时，解码器从词汇表中选择单词。该模型使用注意分布矩阵来确定所选择的指针模式，如公式(16)所示。

$$p(C_i|C_1,\cdots,C_{i-1},X) = softmax(e^t) \tag{16}$$

其中$C_{j(1<j<i-1)}$是已生成的摘要，X是解码器层的当前状态，e^t是输入的注意权重。

5 实验

5.1 数据集

英语文本摘要有一些开源数据集，如DUC(Barrera and Verma, 2011)、gigaword(Napoles et al., 2012)和CNN/Daily数据集(Nallapati et al., 2016)。由于缺乏大规模的藏文文本摘要数据集，本文采用新闻标题作为参考摘要。语料库来源于中央民族大学自然语言处理实验室的舆情项目，共收录藏文新闻约50,000条。

5.2 数据预处理

藏语是一种以字符为基本单位，以“·”分隔的字母语言。一条竖线“།”表示短句的结尾。因此，本文首先将句子以“།”分隔，然后使用TIP-LAS工具对爬网语料库进行分段和词性标记(Li et al., 2018; Li et al., 2015)，然后，将数据作为Word2vec和Fastext模型的输入，分别生成词向量。最后，该模型还生成句子向量。

5.3 评测方法

评测方法是自动文摘研究的关键。评价方法可分为内部评价法和外部评价法。前者通过直接分析摘要的质量来评价。后者将其应用于特定的任务，如自动问答、文本分类等，并根据客观结果对其性能进行评估。相较于自动评价方法，手工评估方法成本高，主观上缺乏一定的公平性。目前，Lin等人参考机器翻译自动评测方法BLEU(Papineni et al., 2002)，提出了ROUGE(Recall-Oriented Understudy for Gisting Evaluation)评测方法(Lin, 2004)。它首先形成由多个专家生成的标准汇总集。然后，与模型生成的自动摘要进行比较。最后，对重叠的基本单元进行统计，评价摘要的质量。ROUGE已成为总结评价技术的通用标准之一。ROUGE系列评价指标包括ROUGE-N、ROUGE-L、ROUGE-S、ROUGE-W。最常见的评价指标是ROUGE-N。它基于n-gram共现统计。n的范围是从1到4。计算如公式(17)所示。

$$ROUGE-N = \frac{\sum_{S\in\{Refsummaries\}}\sum_{n-grams\in S} Count_{match}(n-gram)}{\sum_{S\in\{Refsummaries\}}\sum_{n-grams\in S} Count(n-gram)} \tag{17}$$

其中$Refsummaries$表示引用摘要，$Count(n-gram)$表示引用摘要中的个数，$Count_{match}(n-gram)$表示生成的摘要和引用摘要中的公用个数。

ROUGE-L是基于最长公共子串的统计，ROUGE-S基于词对的统计序列，ROUGE-W则被认为是基于ROUGE-S的字符串的连续匹配。不同的方法对不同类型的总结评价有不同的影响。

5.4 参数设置

本文设置了TextRank、抽取型网络和理解型网络模型的参数，如表1-3所示。

Parameter	Value
Gini coefficient	0.75
Iteration number	1000
Stop iteration value	0.001
Redundancy coefficient	0.5

表 1: TextRank主要参数

Parameter	Value
Hidden size	64
N_layers	2
Batch_size	20
Epoch	50
Learning_rate	0.01

表 2: 抽取型网络主要参数

Parameter	Value
Hidden size	64
Batch_size	20
Epoch	100
Learning_rate	0.01
Vocab_size	5000

表 3: 理解型网络主要参数

5.5 实验结果

1) 抽取式摘要

本文使用ROUGE作为评价，并进行以下实验。

TF-IDF：本文使用TF-IDF方法计算单词的权重。词的权重之和构成句子权重。提取的按权重排序的摘要用作基线。

TR+WF：本文使用TextRank算法提取句子，并用词频共生矩阵计算相似度。

TR+Fasttext：在TextRank迭代中，本文使用Fasttext模型生成句子向量并计算相似度。

TR+Word2vec：在TextRank迭代中，本文使用Word2vec模型生成句子向量并计算相似度。

Bi-GRU：本文使用双层Bi-GRU神经网络提取句子作为摘要。

为了提高TextRank算法的性能，引入外部知识库。Word2vec和Fastext模型生成的藏文文件大小见表4，实验结果见表5。

Word2vec		Fasttext	
Corpus	3.2GB	Corpus	3.2GB
Size	167MB	Size	157MB
Dimension	100	Dimension	300

表 4: 语料库以及Word2vec和FastText生成文件大小

Model	Rouge-1	ROUGE-2	ROUGE-L	Time(h)
TF-IDF	16.4	7.9	11.4	0.5
TR+WF	21.3	10.4	21.6	14
TR+Fasttext	26.6	11.1	22.6	30
TR+Word2vec	32.7	18.9	29	23
Bi-GRU	20.1	11.3	15.2	1

表 5: 抽取式摘要结果

根据表5可以发现，TextRank算法比其他方法取得了更好的性能。与传统的词共现矩阵和TF-IDF相比，在整合外部知识库时，ROUGE评分分别提高了5.3%、0.7%和1.0%。这意味着

Title:

དབྱི་རན་གྱི་ཧྲུང་ཧྲང་ལྷའུ་ཀ་ཉེས་ཡུག་རེན་གྱི་གནམ་རྒྱའི་དོན་ཆུན་གྱི་རྒྱུ་ཆུན་ལ་ཐུར་བས་རྐོག་ཞིབ་བྱ་རྒྱའི་རེ་བ་བཏོན་པ།

(Iranian President Rouhani calls for further investigation into the cause of the Ukrainian plane incident.)

Source Text:

དབྱི་རན་གྱི་ཧྲུང་ཧྲང་ལྷའུ་ཀ་ཉེ་ཡིས་ཚེས་14ཉིན་དབྱི་རན་རྒྱལ་གཞུང་བརྙན་འཕྲིན་ཁང་གི་ཐད་གཏོང་བྱེད་དབྱི་རན་ཁྲིམས་འཛིན་མཁས་པའི་ཚོགས་ཆུང་ལས་ཁྲིམས་ར་"འཇུག་དགོས་པའི་རྩང་བྱ་བཏོན་ནས། གོམ་གང་མདུན་ཐུར་ཡང་ཞིབ་ཕྱོགས་ཀྱི་ལས་གནམ་རྒྱའི་དོན་ཆུན་གྱི་རྒྱུ་ཆུན་ལ་རྟོག་ཞིབ་དགོས་པ་བཤད། ཁོ་ང་གིས།"འཕྲུལ་རྒྱུད་གནམ་ཐུ་ས་ལ་ཕྱུང་བའི་དོན་ཆུན་ནི་རྒྱུན་ཧྲན་ཞེས་མི་རྒྱ། "འཛམ་གླིང་ཡོངས་ཀྱིས་དབྱི་རན་གྱིས་ཇི་ཕྱ་ཐག་གཅོད་པར་ཕྱ་བཞིན་ཡོད།"ཅེས་གསུངས་ཡོད། དུས་མཚུངས་སུ། ལྷའུ་ཀ་ཉེ་ཡིས་དབྱི་རན་དམག་སྡེགས་ཀྱིས་འདི་ཇོན་དུ་"འཛེམ་ཟོལ་མེད་པར་ནོར་འཁྱལ་ལས་བརྙངས་པའི་"རྣམ་འགྱུར་ལ་ཡི་རང་ཡོད་པ་དང་། རྐོག་ཞིབ་མཇུག་འཕྲས་ཀྱིས་མེ་ཚང་མར་ལན་ཁ་སྦྱོང་ཞུབ་པའི་རེ་བ་ཡོད་ཅེས་བཤད།

(Iranian President Rouhani said on Iran's national television live broadcast on the 14th that the Iranian judiciary called for the establishment of a "special court" consisting of ten high courts and experts. Further investigate the causes of the Ukrainian aircraft incident. He said,"The crash of the passenger plane is abnormal. The whole world is watching how Iran handles it." At the same time, Rouhani said that the Iranian military expressed its admiration for its "sincere admission". I hope the survey results will give you a response.)

Extract Result:

དབྱི་རན་གྱི་ཧྲུང་ཧྲང་ལྷའུ་ཀ་ཉེ་ཡིས་ཚེས་14ཉིན་དབྱི་རན་རྒྱལ་གཞུང་བརྙན་འཕྲིན་ཁང་གི་ཐད་གཏོང་བྱེད

(Iranian President Rouhani said on Iran's national television live.)

གོམ་གང་མདུན་ཐུར་ཡང་ཞིབ་ཕྱོགས་ཀྱི་རྒྱུ་ཆུན་ལ་རྟོག་ཞིབ་དགོས་པ་བཤད།

(Further investigation into the cause of the Ukrainian plane incident.)

འཕྲུལ་རྒྱུད་གནམ་ཐུ་ས་ལ་ཕྱུང་བའི་དོན་ཆུན་ནི་རྒྱུན་ཧྲན་ཞེས་མི་རྒྱ།

(the airliner crash is unusual.)

ལྷའུ་ཀ་ཉེ་ཡིས་དབྱི་རན་དམག་སྡེགས་ཀྱིས་འདི་ཇོན་དུ"འཛེམ་ཟོལ་མེད་པར་ནོར་འཁྱལ་ལས་བརྙངས་པའི་"རྣམ་འགྱུར་ལ་ཡི་རང་ཡོད་པ་དང་།

(Rouhani said that the Iranian military expressed its admiration for its "sincere admission".)

Our Model:

དབྱི་རན་ ལྷག་ཀ་ཉེན་ ཇོན་པ་ ཇོན་པ་ གནམ་རྒྱ། རྒྱུན་ཧན་མེན་པ་ བརྟག་དཔྱད། དབྱི་རན་

(Iran Rouhani says says planes abnormal investigated Iran.)

表 6: 基于统一模型的摘要生成实例

外部知识库提高了TextRank算法的抽取性能。与Fastext模型相比，Word2vec模型的ROUGE评分提高了9%。证明了Word2vec模型的有效性。然而，TextRank算法的迭代时间太长，不适合大规模语料库。Bi-GRU神经网络的性能不如TextRank算法。但是，在相同的语料库规模下，所需的迭代时间小于1小时，更适合大规模语料库。

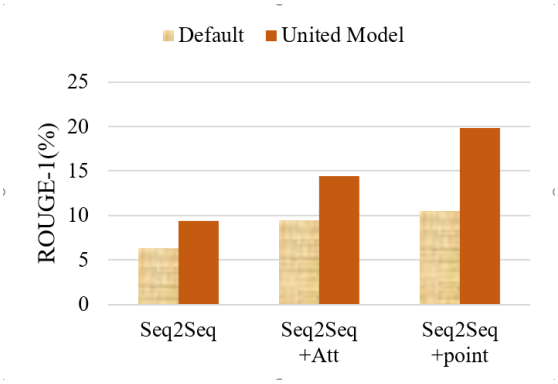


图 2: ROUGE-1实验评测结果

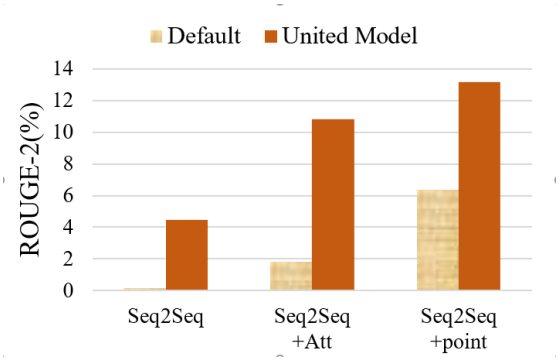


图 3: ROUGE-2实验评测结果

2) 理解式摘要

默认：本文分别使用带有注意机制的Seq2Seq模型和带有指针机制的Seq2Seq模型生成摘要。

统一模型：首先使用双层Bi-GRU提取最能表达原始新闻的句子，然后分别使用带有注意力机制的Seq2Seq模型和带有指针机制的Seq2Seq模型生成摘要。实验结果如图2和图3所示。

从图2，3中，本文可以观察到以下情况： 1) 默认模型显示的总体效果很差。主要由于藏文新闻文本太长，导致神经网络无法对新闻进行良好编码。在藏文新闻篇幅不缩短、摘要直接从原文中产生的条件下，“ROUGE-1”和“ROUGE-2”的得分趋于接近零，说明所产生摘要的可读性和连贯性都较差。与默认模型相比，统一模型得到的结果有了很大的改进，进一步证明了抽取式摘要和理解式摘要的结合不仅可以压缩文档，而且可以删除冗余信息，从而解决了藏文新闻长文本无法编码的问题。2) 新闻标题中的许多词汇都来源于新闻，而使用默认模型生成的摘要只包括藏文词表中的词汇。但藏语词表中不存在人名、地名等专有名词，评价结果较差。添加指针机制后，文本可以根据注意力从原始文本中复制出来，生成的摘要更接近新闻标题。那么Seq2Seq+attention+point模型得到的ROUGE-1分数比Seq2Seq+attention模型高5%。而且，得到的ROUGE-2评分提高了2%，证明了指针机制能够更好地摘要提高的质量。

表6给出了使用带指针机制的双层Bi-GRU网络从新闻中提取摘要的例子。用双层Bi-GRU神经网络选出四个句子。本文可以看到像“ཨོང་གི་མ།”这样的多余句子和短语被删除了。理解式模型生成的摘要可以粗略地表达标题中包含的信息，证明了模型的有效性。单词“ལུའ་ཏ་ཞི་”不出现在词汇表中，但指针网络可以在原始文本中指向该单词并复制它以生成摘要。证明了指针机制可以解决OOV问题，增加了摘要的新颖性。此外，本文发现“སྒོན་པ་”一词出现了两次。这说明生成的摘要存在重复性问题，这与传统的中英文文本摘要模式相似。

6 总结

本文提出了一个藏文新闻摘要生成的统一模型。在该模型中，结合了抽取式摘要和理解式摘要的优点，解决了神经网络无法对太长的藏文新闻进行编码的问题。在藏文摘要的生成过程

中，采用了指针机制和注意机制来解决与OOV相关的问题。然而，仍有许多困难有待解决。首先，作为参考摘要的标题不能包含原文的重要信息。其次，生成的摘要存在语义重复问题。今后，本文将使用K-Means聚类方法生成参考摘要，以提高原始信息覆盖的准确性。然后，本文将使用覆盖机制来解决语义重复问题。

参考文献

Anjian Cairang. 2010. Research on Automatic Summarization of web pages in Tibetan search engine system. *Microprocessors*, 31(5):77–80. *In Chinese.*

Araly Barrera and Rakesh Verma. 2011. Automated Extractive Single Document Summarization: Beating the Baselines with a New Approach. Proceedings of the 2011 ACM Symposium on Applied Computing, TaiChung, Taiwan, China, 268–269.

Ronald Brandow, Karl Mitze, and Lisa F. Rau. 1995. Automatic condensation of electronic publications by sentence selection. *Information Processing and Management*, 31(5):675–685.

Jianpeng Cheng and Mirella Lapata. 2016. Neural Summarization by Extracting Sentences and Words. *Association for Computational Linguistics.* Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 484–494.

Sumit Chopra, Michael Auli, Alexander M. Rush. 2016. Abstractive Sentence Summarization with Attentive Recurrent Neural Networks. *Association for Computational Linguistics.* Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, California, 93–98.

Mahak Gambhir and Vishal Gupta. 2017. Recent automatic text summarization techniques: a survey. *Artificial Intelligence Review*, 47:1–66.

Bohan Li, Huidan Liu, Congjun Long and Jian Wu. 2018. Tibetan word segmentation based on deep learning. *Computer Engineering and Design*, 39(01):194–198. *In Chinese.*

Yachao Li, Jing Jiang and Jiayangji. 2015. Tip-las: an open source tagging system for Tibetan word segmentation. *Journal of Chinese Information Processing*, 29(6):203-207. *In Chinese.*

Chin Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. *Association for Computational Linguistics.* Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, Barcelona, Spain, 74–81.

H. P. Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165.

Inderjeet Mani and Mark T. Maybury. 1999. Adavnces in Automatic Text Summarization. *Computational Linguistics*, 26(2):280–281.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing Order into Text. *Association for Computational Linguistics.* Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, Barcelona, Spain, 404–411.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. SummaRuNNer: a recurrent neural network based sequence model for extractive summarization of documents. *AAAI'17.* Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, 3075–3081.

Ramesh Nallapati, Bing Xiang, and Bowen Zhou. 2016. Sequence-to-Sequence RNNs for Text Summarization. *ArXiv.* abs/1602.06023.

Nankui Niangruo and Anjian Cairang. 2016. Research on Extraction of Tibetan text Abstract Based on sensitive information. *Network Security Technology and Application*, 4:58–59. *In Chinese.*

Courtney Napoles, Matthew Gormley and Benjamin Van Durme. 2012. Annotated Gigaword. *Association for Computational Linguistics.* Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction, Montreal, Canada, 95–100.

Kishore Papineni, Salim Roukos, Todd Ward, Weijing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. *Association for Computational Linguistics*. Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, 311–318.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A Neural Attention Model for Abstractive Sentence Summarization. *Association for Computational Linguistics*. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 379–389.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get To The Point: Summarization with Pointer-Generator Networks. *ArXiv*. abs/1704.04368.

Guochao Sun. 2017. Research and implementation of Web Text Summarization System Based on LDA topic model. Shandong University of science and technology. *In Chinese*.

Yan Xie. 2011. Research on Automatic Summarization System Based on LSA and paragraph clustering. Liaoning University of science and technology . *In Chinese*.

Wenpeng Yin and Yulong Pei. 2015. Optimizing sentence modeling and selection for document summarization. *IJCAI'15*. Proceedings of the 24th International Conference on Artificial Intelligence, 1383–1389.

蒙古文拼写形式多样化现象研究

白双成
内蒙古社会科学院/呼和浩特市
baishuangcheng@qq.com

呼斯勒
内蒙古社会科学院/呼和浩特市
husela@163.com

摘要

蒙古文文本中存在一个有别于多数其他文字的特别现象——看到的单词字形正确但其内码序列不正确，或者说单词“变形显现字形”序列正确但“名义字符”序列不正确的现象，我们称其为蒙古文的拼写形式多样化现象。本文先定义该现象及相关概念，再通过简单图示、例词拼写形式穷举、新闻语料统计分析和基于整篇文章标注统计等多方式、多角度论证这一现象的事实性和严重性，分析导致这一现象的深层原因并指出拼写形式多样化对蒙古文信息处理和应用方面的严重影响，最后提出通过推广普及录入规范和标准提高用户意识、使用智能输入法避免误录、使用校对纠错工具后纠正、基于生语料的统计学习方法为补充等多途径解决方法。本文对蒙古文标准编码的推广普及具有较好的参考价值。

1 引言

蒙古文编码国家标准将蒙古文字符分为“名义字符”（Nominal Character）集和“变形显现字形”（Presentation Form/Character）集，并规定前者用于信息存储、传输和计算，有明确的码位，后者（即蒙古文各类丰繁复杂的字母变形形式）仅用于信息的输出（显示和打印），不需要码位 (Unicode, 2020)。把一种文字的字符分为“名义字符”和“变形显现字形”进行编码，并把信息传输与信息输出截然分开来处理，是一个完全不同于英法德等西方文字和汉日韩等东方文字通用方式的较特殊的编码结构与编码方式。原本在其他文字中特别简单的字符C_i到字形T_i的“一对一”“映射关系”，变成了字符序列$C_1C_2.....C_n$到字形序列$T_1T_2.....T_m$的“多对多”的复杂“转换过程”。由于蒙古文的这一转换过程远复杂于阿拉伯文，仅仅依靠词首、词中、词尾等词内位置进行调形的阿拉伯文成功经验已无法满足蒙古文复杂变形需求。幸好目前已有多数操作系统、浏览器及通用基础软件环境具备了这一编码方式的实现条件，技术层面上基本攻克了这一复杂转换过程的技术难题。只待各大厂商严格遵照“用户协定”和转换规则 (确精扎布, 2014)即有希望实现对于任意一个名义字符序列$C_1C_2.....C_n$转换为唯一一个变形显现字形序列$T_1T_2.....T_m$，从而真正实现编码统一。为解决同形词（Homograph）完全有可能存在另一个名义字符序列C $_1$'C_2'$.....C_k$'（k与n不一定相等）同样被转换为$T_1T_2.....T_m$，Unicode标准Core Specification中13.4节给出了如表1的一个案例。

字符序列C			词形	词意		字符序列C			词形	词意
1	1824	u		长度		6	1823	o		宫殿
1	1837	r				6	1837	r		
1	1832	t				6	1833	d		
1	1824	u				6	1824	u		

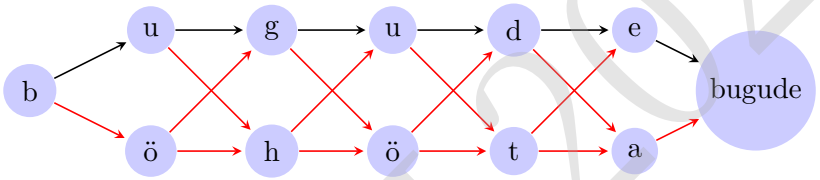
Table 1: Unicode标准提供的蒙古文词形趋同案例

但是，C'是否是一个完全不合法的字符序列或根本不是用户想要的字符序列呢？这种我们所不希望的事情发生概率如何呢？非常不幸的是，由于蒙古文的“一字一音”、“一字多形”和“一

形多字”等自然属性，这种可被映射为相同T的C的组合非常庞大且非常常见（见下文分析）。蒙古文文本中存在的，这种单词输出（屏幕显示或打印）的“变形显现字形”所表现的字形正确，但其存储的“名义字符”序列（文本内码序列）不正确现象，我们称为蒙古文拼写形式多样化现象（Mongolian Spelling Diversity Phenomena），简称为拼写多样化。也有文章称其为同形异码词 (敖敏等, 2011)，是对同一事实的不同层次命名。

正由于蒙古文的拼写多样化现象，蒙古文拼写错误（spelling error） (Islam et al., 2009)有别于其他文种，可细分为“读音错误”和“词形错误”。依据字符序列C转换结果T的唯一性特性可知，当一个单词的词形T错误时对应的C也一定错误，所以词形错误时读音一定错误（详见2.5）。相反，词形正确却无法保证其读音正确。所以拼写多样化也可以叫“字形正确，读音错误”现象。从这个意义上讲，蒙古文的读音错误才是其他文字中所述拼写错误，但从其他文字经验和直觉而言，很容易理解为字形错误是拼写错误。所以我们也可以说蒙古文的拼写错误有“字形正确但读音错误”和“字形读音都错误”两个层次。本文中“词”是指单词名义字符序列，而“词形”是指其变形显现字形序列，所以同形词即指同形异码词。本文生语料统计计算中没有进行字形纠错和归并。

另需要说明的是，MIT编码（因蒙科立成果转化而用户习惯称其为蒙科立编码）） (白双成等, 2013)作为一种“全字符编码”方式，本身就是基于标准编码框架的一种变形显现字形方案。所以，不管是按名义字符形式保存的标准编码还是按变形显现字形形式保存的蒙科立编码，只要是基于“音”的编码形式就必然存在拼写形式多样化现象，本文分析结果通用于所有此类编码方式的文本。文章 (敖敏等, 2011)提到蒙科立编码文本语料库中存在同形异码词，但此文集中于用同形异码字符替换和符合字符拆分、组合方式归并同形词上，与本文目标具有较大差异。蒙古文自动校对 (斯·劳格劳, 2009) (苏传捷等, 2013)等也提及蒙古文同形异码，但都没有对此进行特别深入分析。



2 拼写多样化情况

2.1 简单的拼写多样化案例

我们先看一个简单且容易理解的拼写形式多样化现象。因ö/ü、x/g、d/t、a/e四对字母在相同词内位置和相同阴阳性（ ）条件下经常表现为同形，人们很容易就“发现”录入常用词 (bugude)时可随意替换这几个字母而获得所需词形，并至少可列举出图1所示32种拼写方法（其中只有实线路径是正确拼写）。原本在书面教学和日常使用中就很容易混淆的这些同形异音字母在一个单词中如此高频度组合出现，录入中混淆不足为奇。实际上，这个词形的可拼写形式远不止这32种。

2.2 基于例词的拼写多样化穷举

我们再以常用词 （undusuten）为例，试图穷举其所有可能的拼写形式。这次我们不仅要考虑上例所用同形异音字母之间的替换输入，还要考虑字形之间内含关系（ 被拆分为 ）和切分歧义 可能是 ，也可能是 ），甚至考虑古文用法字形等编码范围内的所有可能录入方式。据此我们可以理论上穷举出如表2的多种拼写方式。

其中音标后上标1和2表示那个字母的变体，就是变体选择符（Mongolian Free Variation Selector）的缩写形式。例如 a2表示字母a的词中变体，也就是古文用的单齿a。 n1表示词首不带点的n辅音（ Consonant）古文写法。 ö1表示ö的词中变体等等。

从表1可知，这个单词词形有多达(2+2*3+3*4*3+3*2)*3*2*4*1*4*2*3*3=86400种拼写形式。考虑到这个数字已经非常大，足以说明问题，也为了防止表格过于繁杂，我们没有再列举w辅音的 形等更为偏激的异常拼写方式。如果要考虑连续、交替、重复使用多个控制字符，可以稍微夸张地说每个单词有无穷多种拼写形式。

表 2 穷举ündüsüten 一词拼写形式表

字母									
拼写形式	ö ü		a e n	d t	o u ö ü		o u ö ü	d t	a e n	a e n
	o u	i j y[1]								
	e a[2] n[1]	o u ö ü								
	ö[1] ü[1]									
组合数	2+2*3+3*4*3+3*2		3	2	4	1	4	2	3	3

2.3 真实语料统计下的拼写多样化

穷举分析毕竟是理论推导，而且有的拼写方式过于复杂，实际出现概率极低。那么实际应用情况是什么样呢？为此我们抓取三个新闻网站的文字性新闻报道页面，并经过行序恢复、HMTL标签剔除、编码转换等一系列预处理后形成了测试集，本文称其为MGLNews。其数据情况如表3所示。

在可获取众多数字资源中，尤其是各种网络资源中选择这三个新闻网站数据作为实验数据的原因主要有

Website	Docs	Sentences	Tokens	Types
中国蒙古语新闻网MNN	86189	4008244	49243265	312902
中国蒙古语广播网CNR	43857	885970	11493090	84458
央视网CNTV	8195	180570	2039115	57216
合计	**138241**	**1214513**	**62775470**	**454576**

Table 2: MGLNews数据情况汇总

- **可获取性**：选择网络资源的首要原因是它有便利的可获取性，便于其他研究人员也可以获取对照。
- **可靠性**：正规新闻媒体机构主办，稿件经过编辑、审核等多道编审流程发布。具有内容相对可靠、术语相对规范统一、干扰因素相对低等优势。
- **时效性**：作为新闻类网站，具有较强时效性，可基本反映新词术语和蒙古文使用现状。当然，时政类新闻为主的新闻内容词汇量必然比不上文学作品，文风也相对拘谨。
- **代表性**：虽然是正规新闻稿件，但依然存在较为严重的读音错误，也不乏字形拼写错误，具有普遍代表性。
- **可验证性**：因工作便利，可对此三个网站爬取内容进行正确性验证，确保网页爬取、网页模板分析、格式转换、行序回复等工作正确。
- **结构性**：可额外获得结构化数据（Structured Data），便于进行按文档种类各自分类训练，便于进行关键字抽取（Keyword Extraction）、摘要生成（Summary Generation）等有监督学习（Supervised Learning）的后续研究工作。搜索引擎中结构化搜索（Structural Search）就是基于MNN的结构化数据（实际用TREC标记标示）进行了充分训练和验证。
- **可延续性**：这三个新闻网站每日稳定更新。通过前期试验，语料搜集工具成熟后，可以定期更新扩充语料。

表 4 拼写形式统计节选数据

拼写种类	所属词形频度	所属词形数	平均频度	所属词例（top 5）
273	90485	1	90485	
179	5954	1	5954	
118	11472	2	5736	
……				
50	122088	29	4209	
49	181698	31	5861	
……				
1	310317	141006	2	

表 5 词 的部分字形相似词

单词	类型	原因	
	原词	名词“要求”	
	形近	动词“要求”	
	拼错	替换	辅音“ ”替换为辅音“ ”
	拼错	替换	辅音“ ”替换为“ ”
	拼错	替换	辅音“ ”替换为元音“ ”
	拼错	替换	辅音“ ”替换为辅音“ ”
	拼错	插入	多输入一个齿“ ”
	拼错	插入	重复输入辅音“ ”
	拼错	插入	多输入一个齿“ ”
	拼错	插入	多输入一个齿“ ”
	拼错	插入	多输入一个齿“ ”
	拼错	插入	多输入一个齿“ ”
	拼错	删除	少输入一个结尾元音 A
	拼错	删除	少输入一个辅音 R
	拼错	删除	首音节少输入一个元音 A

将MGLNews语料所有单词按字形进行归类后获得了所有词形统计数据，表4节选展示了部分典型数据。

表4第一行表示，词形 在语料中共出现90485次，有273种不同拼写方式，是拼写形式最多的词形。第三行表示 和 这两个词形各有118种拼写方式，共出现11472次，每个词形平均出现频度为5736次。最后一行表示141006个词形只出现了一种拼写形式。如前所述，语料中不仅存在读音错误，也有字形错误，表3统计过程中我们并没有进行错误字形纠错。例如，表5展示了MGLNews中单词 的部分词形相似词，其中只有一个是形近字，其余都是拼写错误。

实际数据表明，蒙古文拼写形式混乱程度远超我们之前的估计[3]。这种混乱不仅有理论依据和发生概率，实际使用中人们也确实经常犯这种错误，而且很多用户根本没有意识到自己的错误所在。更何况还有表5所示拼写错误也非常多，这给我们的数据利用形成了非常大的阻碍。

从统计数据分析可知，拼写多样化基本与单词组成字母的同形字母个数及字形组合可能性而定。显然单词越长，拼写形式多样化概率越大。常用词拼写形式个数略显多一些的原因仅仅是因为常用词频度较高，其各种拼写形式出现概率高了而已，相信随着语料量的增大，“大数法则”凸显后，这一现象渐渐平稳并趋于稳定。

2.4 基于完整文章标注统计的拼写多样化

内蒙古日报2014年6月6日第一版的题为  的一篇文章[0] 共计411词。其中词形正确但读音错误的拼写现象共出现182次，竟达45.25% [1] 。内蒙古日报社编辑们作为专业文字工作者，非常清楚自己要写的词如何拼读，只是他们没有意识到拼写正确的必要性，受到传统纸质媒体出版的多年影响，一般认为传递的是蒙古文字形，没有必要一定要读音正确。或者说，他们将基于“音”的标准编码还是当成基于“形”的编码来用。这种现象绝不是内蒙古日报独有，它是整个蒙古文信息处理和应用中普遍存在的，也是我们需要认真面对和解决的重要问题。

经认真分析，我们可以看出这篇文件的录入者有几个录入习惯：①不用o 、ö，只用u、ü；②不管读音是t，还是d，字形  用t输入，字形 都用d输入；③阴性词里的x、g都用g录入；④词末的 不管阳性、阴性都一律用u录入；⑤不用分写词缀( )前必须用的202F等等。

经我们持续观察，这种录入习惯具有普遍代表性，仅仅是不同人对不同键位具有一定偏好而已，但总体思路基本相似。

3 拼写形式多样化的原因浅析

蒙古文拼写形式多样化原因很多，主要有：

3.1 同形字母混用

如前所述，同形异音字母混用是拼写形式多样化的主要原因。其中尤其以词首o/u、词首ö/ü、词中/词尾的o/u/ö/ü、阴性词中的x/g、所有d/t、词中a/e、词尾a/e/n混用占绝大多数。这种混用更大是源于认识问题，还有一部分是源于确实分不清楚如何正确拼读。

3.2 分写词缀误录

蒙古文分写词缀视觉上与前导词分写，容易被误解为独立单词。所以让人们普遍理解、接受并规范输入需要一个过程。目前，录入分写附加成分的错误主要集中在不用控制字符202F和使用错误字母两点。例如“所有格”词缀 yin可以有-iin –ii -iie –ǰyn –ǰy -ǰye-yi -yie 和jin.....jie ji 等十几种错误拼写方。

3.3 控制字符误用

因控制字符是不可见字符，目前操作系统和编辑器又缺少控制字符查重（Duplication Checking）或过滤（Filtering）功能，乱用、误用情况在所难免。尤其是生僻且写法特殊的外来词，录入者可能反复交替试用几个控制字符后最终获得所需字形，但有可能录入了多余控制符而浑然不知。即使是常用词也有可能中间插入多余控制字符而表面上看不出来。例如 后放置FVS1（U+180B）后变成 ，之后可加入随意多个控制字符而不变形。所以理论上 可以有A+FVS1和A+FVS1+FVS2等无限多个拼写方式。

3.4 异常同形词使用

某些人会利用特定字体中形似字符来代替输入。例如，有“方正书版”系统用户利用“方正白体”的特点，将“属格”分写附加成分 -u直接用词首形辅音 b/来输入，在一般出版印刷中难以辨别，只有当切换到手写体或进行编码转换时才有可能暴露。而这些用户还会为自己的创新的简易录入而沾沾自喜。

3.5 字体瑕疵的利用

虽然目前各机构都声明自己的OpenType/AAT字体是符合标准，但部分字体显然并未完全执行国家标准、用户协定和转换规则最新版本，导致不同字体转换结果存在差异，破坏了字符序列到字形序列的“唯一性”约定。例如，因对用户协定理解不同或疏忽，有的机构OpenType/AAT字库规则中“辅音+w+辅音”条件下w（u+1838）被转换为 形，而有的转换为 形。所以， 中E被误录为w时，有些字体中显示了期望字形，而有些字体可能显示

[0]原文已转载至http://www.mgyxw.net/mdls/am/amview.aspx?iid=138925&mid=7273 。

[1]本案例数据源自确精扎布教授与作者联名提交的报告，由六月副研究员完成校对审核，因篇幅所限未付标注数据。

为 了。因为非专业人士不可能甄别出这么细致的错误和差异而将错就错地使用，导致使用真正正确的字体时反倒字形出错了。这种字体转换规则在国家标准[8]、用户协定和转换规则执行上的差异所带来的编码混乱危害性不比原来的编码不统一所带来的危害性小。由于这种错误具有一定的隐蔽性和不确定性，只有随着应用深入和数据累计才能暴露，而那时用户将更加迷茫和无措，所以其危害性更大。

4 拼写形式多样化泛滥导致的严重后果

如此泛滥的拼写形式多样化现象具有什么样的后果呢?

4.1 无法检索

无法对这样的文本直接进行各种检索（Search），哪怕是文本编辑器中的“查找、替换”等简单功能都是很不确定的事情。例如上1.4述这篇文章中一词共出现6次，一致拼错为ündüsüden，利用此篇文章的人除非恰巧有相同拼错习惯，否则就无法搜索到这个单词。依据本文穷举演算，一词至少有86400种拼写形式，MGLNews中实际出现273种拼写形式，所以不进行任何处理情况下，依靠内码匹配搜索几乎是不可能的事情。

4.2 无法排序

拼写形式多样化泛滥使得无法排序（Sort）。这里指的排序是指用蒙古文字母表中的字母顺序排序。如果我们将MGLNews不做任何处理而直接排序，总共出现90485次的一词将分别位于273个不同位置。

4.3 无法统计

“检索”和“排序”是计算机最常用、最基本，也是最有实际效能的功能。不能检索和排序的直接后果是不能进行任何形式的统计，哪怕是最基本的字数、词数统计都成为大问题。泛滥的拼写多样化阻碍了成千上万篇文章当做动态资源直接收入“语料库”做进一步研究。这使得语言监测、网络语象等需要大量语料统计支撑的研究工作受阻。

更何况每个语言的统计计算都会有些个性需求。例如。蒙古文统计中分写词缀与前导词算作一个词更符合语言学要求，NNBSP（U+202F narrow no-break space）就是用于连接分写词缀和前导词。由于部分独立单词字形与分写词缀同形，自动纠错难度较大，从而，统计准确率很难得到保障。工作量统计中分写词缀算作一个单词，也许有利于编辑人员稿费统计而故意为之。

检索、排序和统计等最基本的应用需求都难于得到满足，我们该怎么办呢?

5 拼写形式多样化问题的解决方案

既然有这么多问题，我们该如何解决呢？大致有两种解决思路。一种是要完全正确录入或同时对被搜索内容和搜索关键字进行拼写纠错，区分同形异音字母和词，达到精准搜索、精准排序、精准统计。另一种是通过额外算法解决同形异音字母的字形模糊匹配，达到模糊搜索和模糊统计，但这是否违反了标准编码制定初衷，纵容用户拼写错误了呢？对此，我们建议：

5.1 推广普及录入规范和标准，提高用户意识

由蒙古文的自然属性、编码特性及拼写多样现象可知，让用户接受蒙古文标准编码的录入规范，形成良好习惯并不是简单的书写媒介更换问题，可以说是一次重大的变革。把握好了，我们可以借此机会将标准音与信息化同时推进。把握不好，将严重阻碍蒙古文信息化进程。为此我们认为，必须同时注重现有使用者培训和未来使用者培养，且后者更重要。未来使用者培养方面，将蒙古文标准编码的录入规范纳入中小学课程，结合标准音推广工作，从入学儿童开始培养起良好的习惯，待他们毕业走上工作岗位时，将按标准录入视为很正常的必然事件。现有使用者培训方面，我们先重点培训编辑、记者、相关蒙古文工作人员，再逐步扩大到普通用户步骤。单从报纸、期刊和图书出版角度考虑，要想让长期习惯于纸质出版做法的相关人员按规范录入具有一定难度。兼顾字形和读音会额外耗费精力，这明显有悖于他们追求绩效考核的初衷。那么如何让他们意识到这样做的好处并愿意付出这份努力也许是个不简单的系统工程问题。将出版资源升级为“语料库”，作为商品授权给研究机构及商业公司，获取经济效益等都不

一定能凑效。如果不能形成长效机制，很难长久维持。所以从他们自身使用便利角度考虑，让他们意识到这样做了可以便利利用以往资料等可能是更合理的方式。

5.2 使用智能输入法避免误录

在标准编码OpenType/AAT字库实现中，一般都会附带一个键盘映射（Keyboard Mapping）输入法。因复杂文本引擎和字体规则担负了名义字符到变形显现字形的映射转换，输入法本身一般只需从键盘字母映射为蒙古文字母，一般不用做额外处理。因蒙古文是个同形异音字符较多的文字，这种键盘映射输入法没有避免同形字符输入错误避免措施，从输出的字形又不易察觉错误，所以，即使用户了解标准编码框架、懂得输入规范，也意识到规范录入的必要性，但指望所有用户都能按标准录入是不现实的。为标准编码的推广普及，不让终端用户陷入迷茫的一个有效途径就是推广智能化程度较高的输入法做预防性处理。鼓励使用完全符合规范和标准的智能输入法，从录入源头避免错误[9]。智能化输入法确保录入字形和读音正确的同时给用户简单易用的用户体验，让用户不再感觉遵循规范是个负担。此处所述输入法不仅局限于全键盘录入，也包括智能终端的虚拟全键盘、数字键盘及OCR识别录入、语音识别录入等所有输入方式。这些输入方式上必须加以监督机制，尽量避免用户录入错误。不管输入法做到什么程度，总是无法避免OOV，而这部分的正确录入只能依靠用户自律或通过网络协同等方式作为弥补。

5.3 使用校对纠错工具后纠正

虽然通过前两项可以解决今后的问题，但对于历史数据或字形扫描、手写录入等场合，我们需要依赖自动校对和自动纠错。目前“词典+规则”是实现蒙古文文本校对常用方法[6]。不管使用不确定有限状态自动机（NFSA）数据结构获取较高计算效率、使用词干/词缀和生成规则来节省存储空间或是使用最一般的字符串匹配的库结构，其本质无非都是依赖词库，词库中有的词认为是正确词，词典中没有词（OOV）就认为是错别词。再进一步用搭配库或规则对部分同形异音词（例如  的  录入为 的  ）进行甄别。从公开资料来看，未登录词、同形多音词处理还不够成熟，句法和语义层面错误基本未能触及,甚至词法层面的形态变化分析[10]还有待提高，校对和纠错效果基本取决于词典词汇量。字形拼写错误的纠错也不尽如人意，所以有待进一步完善和改进校对和纠错。

5.4 探索基于生语料的统计学习方法

我们有了确保单词读音正确的熟语料，可以顺利开展一些统计建模的科研研究[11]，但目前我们所能获得量还难以支撑实际应用需求，更无法满足需要大数据支撑的个别模型。虽说可以采取各种手段缓解数据稀疏（Data Sparse），但归根结底还得需要足够量数据支撑统计建模[12]。很显然，深度机器学习（Deep Machine Learning）使用的词向量表示（Word Vector Presentation）来说，语料量越大，低维空间（Low Dimensional Space）上的词向量越趋于精准[13]。更何况熟语料没有读音错误只是一种假设，而我们日常产生的原始数据又有如此严重的拼写多样化，我们所能采取的防范措施又不能解决所有问题，所以我们不能单纯等待和依赖加工足够量的熟语料后再开展相关研究工作。另一方面，新词术语研究、语言动态监测、舆情分析等工作总不能还要依赖加工的熟语料。直接利用生语料的研究工作是熟语料建设的必要补充和回旋途径，相铺相成，互为补充，也是解决拼写多样化的一个重要途径。

6 总结

综上所述，蒙古文文本中大量存在拼写多样化现象，严重影响着蒙古文文本的日常应用及科研研究。各种解决方式都无法独自满足需求，需加以综合利用。各项工作的开展势必依赖语料库建设、知识库建设及相应大数据、机器学习等方面的突破，所以我们的研究工作还任重道远。

参考文献

Aminul Islam , Diana Inkpen. 2009. Real-Word Spelling Correction using GoogleWeb 1T 3-grams. *JProceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pages ,Singapore, 6-7 August 2009. c2009 ACL and AFNLP*,1241–1249.

Deniz Yuret, Ergun Bicici. 2009. Modeling Morphologically Rich Languages Using Split Words and Unstructured Dependencies. *ACL-IJCNLP 2009 Conference*.

Daniel Jurafsky,James Martin. 2010. Speech and Language Processing. 人民邮电出版社.

Jacob Devlin , Rabih Zbib. 2014. Fast and Robust Neural Network Joint Models for Statistical Machine Translation. *ACL2014*1370—1380.

The Unicode Consortium[EB]. http://www.Unicode.org.

敖敏,熊子瑜,呼和. 2011. 基于蒙科立输入法的蒙古语同形异码词研究. 第十一届全国人机语音通讯学术会议.

白双成，张劲松，呼斯勒. 2013. 蒙古文输入法输入码方案研究. 中文信息学报2013(06):169-174.

确精扎布. 2014. 确精扎布蒙古文信息处理专辑. 内蒙古教育出版社.

国家质量监督检验检疫总局，国家标准化管理委员会. 2011. GB 25914-2010.信息技术传统蒙古文名义字符、变形显现字符和控制字符使用规则. 中国标准出版社.

斯·劳格劳. 2013. 基于不确定有限自动机的蒙古文校对算法. 中文信息学报, 2009,(06).

苏传捷,侯宏旭,杨萍等. 2013. 基于统计翻译框架的蒙古文自动拼写校对方法. 中文信息学报, 2013,(06).

S·苏雅拉图2001. 蒙古文整词计算机生成理论研究. 中文信息学报,2001(04):59-65..

赵伟,侯宏旭,从伟,宋美娜. 2010. 基于条件随机场的蒙古语词切分研究. 中文信息学报. 2010, 24(5).

面向司法领域的高质量开源藏汉平行语料库构建

沙九[1,4] 周鹭琴[2,4] 冯冲[1,4*] 李洪政[1,4] 张天夫[1,4] 慧慧[3,5]
1.北京理工大学计算机学院,北京市海量语言信息处理与云计算应用工程技术研究中心
2.北京理工大学信息与电子学院
3.甘肃省迭部县初级中学
4.北京市海淀区中关村南大街5号
5.甘肃省迭部县兴迭东街38 号
{shajiu,zhouluqin,fengchong,lihongzheng,zhangtainfu}@bit.edu.cn,huihui@163.com

摘要

面向司法领域的藏汉机器翻译面临严重的数据稀疏问题。本文将从两个方面展开研究: 第一，相比于通用领域，司法领域的藏语需要有更严谨的逻辑表达和更多的专业术语。然而，目前藏语资源在司法领域内缺乏对应的语料，稀缺专业术语词以及句法结构。第二，藏语的特殊词汇表达方式和特定句法结构使得通用语料构建方法难以构建藏汉平行语料库。为此，本文提出一种针对司法领域藏汉平行语料的轻量级构建方法。首先，我们采取人工标注获取一个中等规模的司法领域藏汉专业术语表作为先验知识库，以避免领域越界而产生的语料逻辑表达问题和领域术语缺失问题；其次，我们从全国的地方法庭官网采集实例语料数据，例如裁判文书。我们优先寻找藏文实例数据，其次是汉语，以避免后续构造藏语句子而丢失特殊的词汇表达和句式结构。我们基于以上原则采集藏汉语料构建高质量的藏汉平行语料库，具体方法包括：爬虫获取语料，规则断章对齐检测，语句边界识别，语料库自动清洗。最终，我们构建了16万级规模的藏汉司法领域语料库，并通过多种翻译模型和交叉实验验证了构建的语料库的高质量特点和鲁棒性。另外，此语料库会开源以便于相关研究人员用于科研工作。

关键词: 司法邻域 ; 藏汉平行语料 ; 稀缺资源

A High-quality Open Source Tibetan-Chinese Parallel Corpus Construction of Judicial Domain

Jiu Sha[1,4] Luqin Zhou[2,4] Chong Feng[1,4*] Hongzheng Li [1,4] Tianfu Zhang[1,4] Hui Hui[3,5]
1.Beijing Engineering Research Center of High Volume Lange Information Processing and Cloud Compution Applications,School of Computer Science & Technology,Beijing Institute of Technology
2.School of Information and Electronics Beijing Institute of Technology
3.Diebu County Junior High School, Gansu Province
4.No. 5, Zhongguancun South Street, Haidian District, Beijing
5.No. 38, Xingdie East Street, Diebu County, Gansu Province
{shajiu,zhouluqin,fengchong,lihongzheng,zhangtainfu}@bit.edu.cn,huihui@163.com

Abstract

To date, the Tibetan-Chinese (Ti-Zh) Machine Translation in the judicial domain confronts a data-sparse severe problem. In this work, we tackle the problem from two aspects: 1) judicial Tibetan needs more rigorous logical expression and professional terminology vocabulary than the public domain. However, there hardly exists the high-quality Ti-Zh corpus in the judicial domain, which contains professional terminology and syntactic structure. 2) It is challenging to construct a Ti-Zh parallel corpus due to the unique lexical expression and specific syntactic structure. To this end, we propose a lightweight Ti-Zh parallel corpus construction method for the judicial domain. First, we construct a medium-scale Tibetan-Chinese terminology glossary of the judicial domain to be our prior knowledge, which can avoid the logical expression

and domain terminology missing problems caused by the out-of-domain phenomenon. Secondly, we collect the instance data, such as judgment documents, from the official websites of Chinese courts in various places. To avoid losing the Tibetan lexical expressions and syntactic structures, we firstly search for Tibetan case data, followed by Chinese. Based on the above principles, we build a high-quality Tibetan-Chinese parallel corpus，which consists of the following methods: crawling corpus, document segmentation alignment detection, sentence boundary recognition, automatic corpus cleaning. Lastly, we construct a 160,000 Ti-Zh parallel corpus of the judicial domain, and we evaluate the quality and robustness of the constructed corpus by performing a variety of translation models and cross-validation experiments. Besides, this corpus will be an open-source to provide to other researchers for related research.

Keywords: Judicial Domain , Tibetan-Chinese Parallel Corpus , Data-sparse

1 引言

最近神经机器翻译(Neural Machine Translation, NMT)的进步已经证明，在某些特定领域内，翻译质量基本上可以达到专业的人工翻译，这些翻译模型通常受益于大量的平行语料库，它们的效果在神经机器翻译模型中最为明显。然而，平行语料库的获取和构建需要大量的时间和精力，而且并非所有域或语言对都可以使用相同的数据集，为此，有不少研究者通过数据增强来提升翻译质量，数据增强是一种具有显著价值的技术，它既可用于缓解数据量不足的问题，同时还用于提升模型的稳健性。在图像分类和文本分类等应用中，使用的几乎所有表现最好的机器学习以及深度学习等模型都会用到数据增强技术。例如：启发式的数据增强方案往往需要依靠具有丰富领域知识的人类专家进行人工调整，但这可能导致所得到的增强方案是次优的。词汇替换，这种方法试图在不改变句子主旨的情况下替换文本中的单词，包括基于词典的替换(Zhang X, Zhao J, LeCun Y., 2015)，基于词向量的替换(Jiao et al., 2019)，基于TF-IDF的词替换。反向翻译，利用NMT来解释文本，同时重新训练含义。文本表面转换, 使用正则表达式简单的模式匹配转换。随机噪声注入，其思想为在文本中加入噪声，使所训练的模型对扰动具有鲁棒性。语法树操作将解析和生成原始句子的依赖关系树，使用规则对其进行转换，并生成改写后的句子。这些增强方法对性能的影响仅针对某些特定用例进行了研究。系统地比较这些方法并且分析它们对许多任务性能的影响将是一项有趣的研究。然而与计算机视觉(Computer Vision, CV)中使用图像进行数据增强不同，在自然语言处理(Natural Language Processing, NLP)中文本数据增强是非常罕见, 其主要原因之一是图像的一些简单操作，如将图像旋转或将其转换为灰度，并不会改变其语义。语义不变变换的存在使其增强成为CV研究中的一个重要工具。常规数据增强方法的局限性表明这一领域还存在很大的研究进步空间。常规数据增强技术依赖相关领域的专家，耗时耗力成本高昂，因此研究者开始探索自动化数据增强技术。自动化数据增强领域具有挑战性的难题，从人工设计到自动搜索算法可以发现：1.不同于执行次优的人工搜索，我们要如何设计可学习的算法来寻找优于人类设计的启发式方法的增强策略？2.从实践到理论理解：虽然在实际应用中增强技术的设计研发进展飞速发展，可是由于缺乏分析工具，仍然难以挖掘这类技术的优点。该如何从理论上理解实践中使用的各种数据增强技术？3.从粗粒度到细粒度的模型质量保证：现有的大多数数据增强方法的关注重点都是提升模型的整体性能，通常还需在更细的粒度上关注数据的关键子集。当模型在数据的重要子集上的预测结果不一致时，又该如何利用数据增强来缩减在相关指标上的表现差距？然而真正从根源上解决问题的并不多，为此本文探究最根本最实质性的构建高质量特定领域的平行语料库。本研究中，我们带着以上自动化数据增强技术面临的挑战问题，研究NMT中通过半自动方式构建高质量特定领域的数据增强技术。我们研究针对稀缺资源司法领域的藏汉平行语料在神经机器翻译中的构建，通过半自动化数据增强技术获取了大量司法领域中的藏汉料库，进一步构建了在司法领域中具有裁判文书以及法庭判决书等子领域的庞大藏汉平行语料。此外，我们还表明，使用CWMT2018的通用数据训练基线模型，并使用我们构建的数据集对模型

进行微调，将比基线模型显著提高翻译质量。我们的贡献如下：我们在稀缺资源司法领域公开了160K大小的高质量藏汉平行语料库。这是本文的主要贡献。我们比较了几种识别句子边界以及句对齐的方法，用于构建NMT的平行数据集。我们的实验表明，利用不同的句子边界识别技术的消融策略可以获得更可靠的可用数据。我们发现，对藏汉互译的140K或160K个句子对进行微调以及预训练，可以将大幅度的提升译文质量，在较大的数据集上翻译质量继续提升。

2 相关工作

数据增强是一种普遍存在的技术，通过利用保留类标签的特定于任务的数据转换来增加带标记的训练集大小。为了解决这一难题，Ratner 等人(2017)提出了一种方法来自动化这个过程，通过学习生成序列模型在用户指定的转换函数使用生成对抗的方法。具体是采用了对抗式方法训练变换函数序列生成器，以得到与训练数据相比足够真实的增强数据。2019年谷歌大脑提出了一种自动数据增强方法(AutoAugment) (Cubuk et al. , 2019)，该方法创建一个数据增强策略的搜索空间，利用搜索算法选取适合特定数据集的数据增强策略。此外，从一个数据集中学到的策略能够很好地迁移到其它相似的数据集上。发表在ICLR 2019上的(Wei et al. , 2019)文章中介绍了几种NLP数据增强技术，具体提出了四种简单的操作来进行数据增强，以防止过拟合并提高模型的泛化能力。

在NMT中，通过上下文软连接的方式来处理NMT中数据增强问题(Gao et al., 2019)，这篇文章跟以往在句子中随机删除或替换单词的增强方法有所不同，将一个单词的一种表示替换为一个分布(由语言模型提供)，将该词的嵌入替换为多个语义相似的词的加权组合。由于这些词的权重依赖于被替换词的上下文信息，因此新生成的句子比以前的增强方法捕捉到更加丰富的信息。在超越反向翻译这篇文章(Li et al., 2019) 中作者通过增强数据来提高和扩展神经机器翻译的鲁棒性。他们以扩展有限的噪声数据，进一步提高NMT对噪声的鲁棒性，同时保持较小的模型。探索合并双语词典的方法(Nag et al., 2020)，以实现半监督神经机器翻译，通过一种简单的数据增强技术来解决在反向翻译中对低资源环境下合成句子产生不利影响的问题。结合了广泛使用的双语词典，解码时以逐次生成词从而合成句子达到翻译的效果。在无监督机器翻译中通过学习双语单词嵌入来提升数据增强(Nishikawa et al., 2020)，利用无监督机器翻译模型生成的伪平行语料库，以促进两个嵌入空间的结构相似性，提高映射方法中双语词嵌入的质量。在NEJM-enzh (Liu B and Huang L, 2020)中提出了一个在生物医学领域内构建英汉平行数据集。(Han L, Jones G J F, Smeaton A F., 2020)中介绍了多语语料库的构建方法，包括德语-英语和汉语-英语的平行语料库的提取方式。本文将半自动化数据增强技术应用到NMT中，通过分析比较离散而实际应用的可获取数据，并利用一些技术和方法针对司法领域资源稀缺的藏汉互译任务构建了庞大的平行语料，取得了突破性的进展。

3 语料库的构建

本节描述了我们构建句子对齐语料库的步骤。

3.1 构建平行语料库的基本流程

通过多语言网站获取语料库，从语料库中抽取语句构建平行语料包括以下几个步骤：(1)从多语言网站中抓取所需语言的文档；(2)从爬取的文档中提取纯文本并规范化；(3)两种语言的文档根据其内容进行匹配；(4)在每个文件中，段落被分解成单独的句子；(5)句子随后被排列成句子对；(6)对对齐的句子对进行过滤，去掉重复的和低质量的句子对。其前两步基本是工程任务，而后四步正在不断的探索之中。对于第(3)步，在WMT16中使用了一个用于双语文档对齐的共享任务(Gomes and Lopes, 2016)，其中最佳词条依赖于匹配不同的双语短语对(Read et al., 2012)。对于步骤(4)而言，Read等(2012)系统地评估了9种现有的句子边界检测工具。第(5)步的句子对齐可能是目前最具挑战性的。与文档对齐相比，句子对齐使用更少的文本，但有更多的排列。第(6)步也属于工程化任务，相比第(5)步简单容易实现。

3.2 法律领域内的语料库来源

本文的语料库主要来源于中国裁判文书网站，中国民族语文翻译局，中国知网以及一些官方微信公众平台。中国裁判文书网站提供了一种民族语言文书，其中含有藏文和中文的刑事案件；民事案件；行政案件；赔偿案件；执行案件以及其他案件。中国民族语文翻译局每年会定

期发布每季度的新词术语，例如：“带头攻坚克难、敢于担当”。在中国知网上可以获取法律领域相关的论文，进一步获取具有藏汉双语摘要部分。还有一些官方微信公众号如《藏汉双语法律平台》、《刚察藏汉双语普法平台》以及《TBL酥油灯青年法客》等，推送的相关法律立案以及法庭判决书等数据。以上四种种数据的历史可追溯至2015年。

3.3 获取语料库并断章对齐

我们使用Selenium (2014)抓取所有可用的藏文和相应的中文文章，首先，在爬取期间，为了易于检索内容，获取的文章都按时间顺序排列。其次，对应的文档对通过超链接连接，消除了文档对齐的需要。最后，把藏文和汉文两个对应的PDF或者Word文章按段落标识符分段对齐，研究文章由相关的统计人员校对。

3.4 检测并识别语句边界

我们比较了以下三个方法并取交集: 首先，在汉文中，我们通过统计并发现，汉文的引号出现在断句之前，这使得很容易发现句子的边界。与欧洲语言不同，“.”在汉文中不能兼用作小数点或其他语言标记。所以我们利用“！ 、？ 、。”来检测识别汉文句子的边界。在藏文中，我们同样做了统计实验，另外人工分析并归纳，我们使用跟汉文相似的方法，通过识别“.<ja>、,、.<jb>、.<jc>、.<jd>”(转写为拉丁)来判断藏文句子的边界。其次，我们针对藏文和汉文统一使用如下两个工具进行识别句子边界，分别为Read (2012)等人开源的无监督句子标记器Punkt和Ziemski (2016)等人开源的Eserix系统。最终，我们发现取以上三种方法的交集误差最小，因此取三种方法的交集。

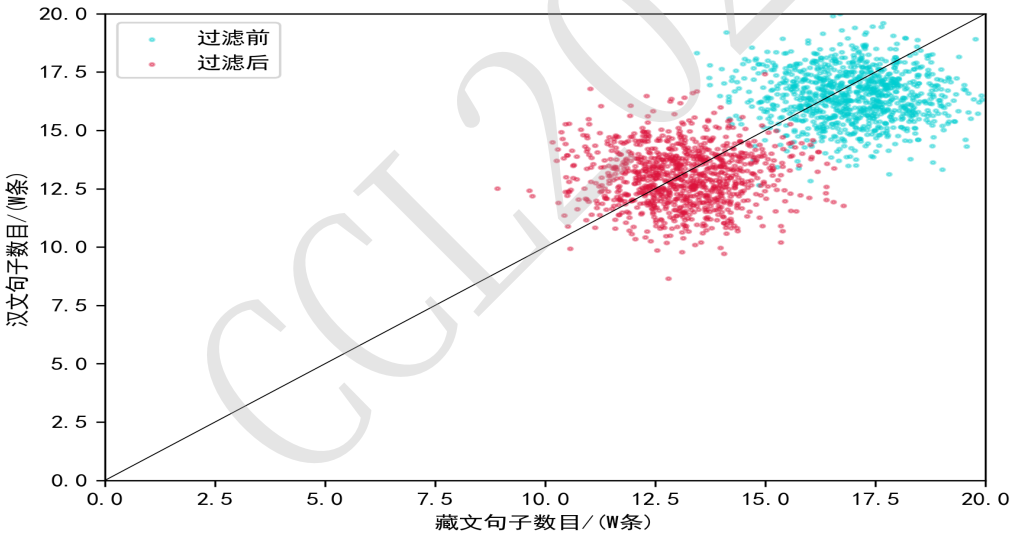


Figure 1: 藏文和汉文在句子数目过滤前后的对比，通过过滤之后藏文的句子数目越来越接近汉文的句子数目。

3.5 清洗和过滤语料库

对于这两种语言，我们过滤掉如下内容：(1)数字以及数字说明；(2)表格和图片及对应的说明；(3)短语以及短句。图1对比了过滤前后不同来源下藏文和汉文句子的数量。在过滤之前，大量文章中藏文句子数远远超过了汉文句子数，用蓝色点表示。是因为在藏语中短语往往构成短句，例如“on kyang |”(转写为拉丁)，而在汉文中很少出现“但是。”，一般是“但是，”。经过滤之后，每篇文章中藏文和汉文句子的数量变得更接近。

3.6 构建双语句对齐

虽然已经提出了一些句子对齐的方法(Simard M and Plamondon P, 1998; Repar et al.,, 2019)，但这些方法在稀缺资源司法领域的语言上表现缺乏共识。我们比较了以下三种方法: 基

藏-汉	数据大小(M)	句子对数目(条)	约占法律领域的数据(%)
中国裁判文书网站	8.68	80000	99.63%
中国民族语文翻译局	5.43	50000	89.56%
中国知网	1.63	15000	75.32%
各官方微信公众号	2.17	20000	93.23%

Table 1: 不同来源下获取的最终平行语料大小。

于长度对齐(Gale-Church)(Gale and Church, 1993)，它是通过假设源句和目标句的长度相似来寻找句子对；基于词典对齐(Microsoft Aligner)(Moore, 2002)，是把单词对应与句子长度结合搜索句子对；基于翻译对齐(Bleualign)(Sennrich and Volk, 2010)，将原始文本和翻译文本进行比较搜索锚定句，然后使用Gale-Church算法对其余的文本进行对齐。 为了比较这些方法，

藏-汉	Count	Percent
1-0	202	4.04%
0-1	204	4.08%
1-1	4132	82.64%
1-2	164	3.28%
2-1	298	5.96%

Table 2: Microsoft Aligner在5000句测试集上的对齐测试结果，其中大多数是1-1对齐。

我们人工构建了两种语言不同来源的5000句测试集。如表2显示了Microsoft Aligner在5000句测试集上对齐类型的分布。将近82.64%的对齐是一对一的。 因为大多数句子对都是一对一对齐

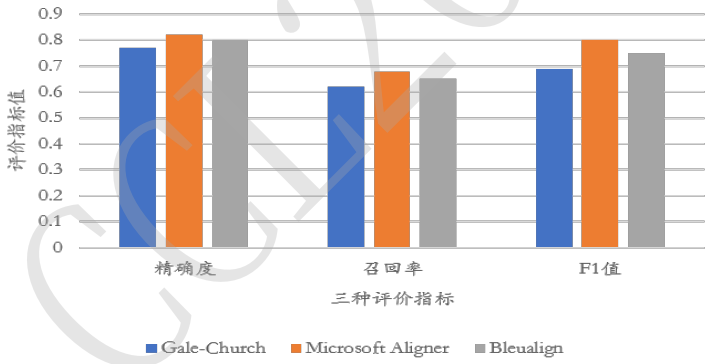


Figure 2: 三种句子对准器在语料库上通过双向对齐即藏对汉和汉对藏的最终结果。

的，对于一对多对齐，所有算法的性能都会显著下降，因此在本研究中我们主要针对一对一对齐。精确度，召回率以及F1值的分数如图2所示。其中微软的Aligner获得最佳F1分，所以我们选择Microsoft Aligner用于构建句对对齐。随后所有的句子对由专业的翻译人员逐句校正，为了句子的流畅性，偶尔会进行句子的衔接和分割，即把一个藏文句子分成两个句子或者更多的汉文句子，反之亦然。最终的数据由专门的编辑小组成员校和统计如表1所示，便用于NMT的训练。

4 实验设置

4.1 模型架构

我们使用基于PyTorch的OpenNMT (Klein et al.,, 2017)框架，使用Transformer-base模型训练，本次实验中所有的网络参数跟论文(Vaswani et al.,, 2017)中的参数设置保持一致。模型有6层编码器和解码器，每个输出大小为512个隐藏单元(Ziemski et al.,, 2016)，使用8个注意头和正弦位置嵌入。最后隐藏的前馈层大小为2048。模型总共训练了100,000步，训练耗时约

为1.5天。使用Adam优化器(Klein et al.,, 2017)，其中β_1=0.9，β_2=0.98，ϵ=10^{-9}，P_{drop}=0.1。我们在藏译汉和汉译藏上使用相同的参数训练，使用CWMT2018官方评测工具衡量译文的质量，具体以BLEU4值为评测指标。我们在8台Nvidia TitanX GPU上训练模型。

4.2 训练语料

我们利用公开数据*CWMT*2018提供的藏汉数据集，该数据都属于新闻领域。以2017*Dev*作为开发集，2018*Test*作为测试集；另外我们把本文构建的数据*OurCorpus*分割为训练集*OurTraining*、开发集*OurDev*以及测试集*OurTest*。具体实验所用的数据如表3所示。在本文中，汉文统一使用Jieba(Sun J, 2012)分词，随后处理为子词(Byte-Pair-

藏-汉	训练集(句对)	开发集(句对)	测试集(句对)
CWMT2018	147434	650	1000
OurCorpus	163000	1000	1000

Table 3: 两种不同训练数据的大小

Encoding，BPE)(Sennrich et al.,, 2015)。藏文先使用西北民族大学开源的TIP-LAS(李亚超, 江静等, 2015)分词，随后按照(沙九，冯冲等, 2020)中音词融合的方式处理，使用80K的源端和目标端词汇表。最终我们实验所用的所有数据的汉文以BPE为粒度单位，藏文以音词融合为粒度单位。我们主要设置了五种方案进行实验：(1)单独用*CWMT*2018和*OurCorpus*数据分别训练模型作为基线系统(*topic*1&*topic*2)；(2)先用*CWMT*2018数据进行训练，其次用*OurCorpus*数据进行微调(*topic*3)；(3)先用*OurCorpus*数据进行训练，其次用*CWMT*2018数据进行微调(*topic*4)；(4)把*CWMT*2018和*OurCorpus*数据合成再训练(*topic*5)。在(2)和(3)中具体微调方式我们参照了(Guo et al.,, 2019)。另外我们还做了一些预训练的实验加以验证我们所构建的数据是可靠的。预训练我们使用BERT (Devlin et al., , 2018)和XLM (Lample and Conneau, 2019)，具体方式我们参考了(Weng et al.,, 2019)文章。

5 结果

5.1 主要结果

实验结果如表4所示，我们的基线系统为*CWMT*2018和*OurCorpus*数据集上训练的模型，我们不难发现，单独在数据集*CWMT*2018和*OurCorpus*上训练时，不管在藏译汉还是汉译藏上，2017*Dev*和2018*Test*在*OurCorpus*数据上的BLEU值偏低于*CWMT*2018数据上的值，相反*OurDev*和*OurTest*在*OurCorpus*数据上的BLEU值胜于*CWMT*2018上的值，至少提升了3.21个BLEU值。首先，我们发现同一领域内的数据具有较强的相似之处，所以针对特定领域的测试集用跟它相同领域的训练集训练是至关重要的，为此，我们构建特定的法律领域数据是很有必要的；其次，本文所构建的数据集在新闻领域的测试集上虽然偏低，但是跟用新闻领域的数据集训练的结果相比，相差最多也不到1.31个BLEU值，相反在法律领域的测试集上远远超越了用新闻领域训练的结果。从此，我们发现本文所构建的平行数据集具有较高的质量。

用*CWMT*2018数据进行训练，其次用*OurCorpus*数据进行微调和用*OurCorpus*数据进行训练，其次用*CWMT*2018数据进行微调在本文中也提升了翻译的质量。当*CWMT*2018数据进行训练，其次用*OurCorpus*数据进行微调时，在2017*Dev*和2018*Test*测试集上的效果优于*OurDev*和*OurTest*测试集上的值；当*OurCorpus*数据进行训练，其次用*CWMT*2018数据进行微调时，在*OurDev*和*OurTest*测试集上的效果优于2017*Dev*和2018*Test*测试集上的值。为此，我们发现虽然通过微调能提升一定的翻译效果，但是不如用该领域内的数据直接训练的效果好。为此我们肯定本文所构建的数据集是很有价值的。从整体实验结果可以发现，所有测试集的值在藏译汉上的评分值都高于汉译藏上的评分值。为此我们认为，目标端的分词质量以及切分粒度相当重要。因为，当藏译汉时，目标端为汉文，而汉文具有很多开源的分词工具并比较成熟，但是藏语几乎没有统一成熟的开源工具，导致每个机构或者高校在藏语相关的分词任务上各不相同，并且很大程度上具有不同的分词粒度。这直接影响了下游的工作。所以我们判断目标端的分词甚至比源端的分词更重要。在表4的最后一行*topic*5可以看出，本次实验通过*CWMT*2018和*OurCorpus*数据合并后的训练模型上译文质量最佳，相比之前单独实验和其

System	汉⇒ 藏		汉⇒藏		藏⇒ 汉		藏⇒汉	
	2017*Dev*	2018*Test*	*OurDev*	*OurTest*	2017*Dev*	2018*Test*	*OurDev*	*OurTest*
*topic*1	47.29	35.75	19.33	20.24	51.74	38.07	23.56	24.67
*topic*2	45.98	34.48	22.54	23.49	50.49	36.78	26.87	27.98
*topic*3	48.22	36.72	23.81	24.76	52.73	39.02	28.14	29.25
*topic*4	50.22	38.32	24.81	25.98	54.67	40.64	28.68	30.13
*topic*5	52.04	40.10	27.12	28.82	56.24	42.08	30.86	32.52

Table 4: 本文主要的实验结果，“*topic*1”为单独用*CWMT*2018数据训练的模型；“*topic*2”为单独用*OurCorpus*数据训练的模型；“ *topic*3”先用*CWMT*2018数据进行训练，其次用*OurCorpus*数据进行微调；“*topic*4”先用*OurCorpus*数据进行训练，其次用*CWMT*2018数据进行微调；“*topic*5”把*CWMT*2018和*OurCorpus*数据合成再训练的实验结果。

System	汉⇒ 藏		汉⇒藏		藏⇒ 汉		藏⇒汉	
	2017*Dev*	2018*Test*	*OurDev*	*OurTest*	2017*Dev*	2018*Test*	*OurDev*	*OurTest*
*topic*6	49.32	37.32	24.21	25.36	53.37	40.20	29.44	30.45
*topic*7	52.04	38.72	26.33	27.56	54.95	41.94	30.48	31.53

Table 5: 用BERT初始化编码器，用GPT初始化解码器，分别用*CWMT*2018作为初始化参数语料*OurCorpus*作为后期NMT训练语(*topic*6)；用*OurCorpus*作为初始化参数语料*CWMT*2018作为后期NMT训练语(*topic*7)。

他的微调都要好，为此，我们证明只有高质量且大规模的平行语料训练模型，NMT才能获得最佳结果。

5.2 消融实验

本节我们按照(Weng et al.,, 2019)中的预训练方法训练，因为GPT是单向语言模型，而BERT屏蔽语言模型可以获得更多的上下文信息。GPT可以对顺序信息进行建模。为此，本文用BERT初始化编码器，用GPT初始化解码器。在表5中的*topic*6行中*OurCorpus*作为初始化参数语料，*CWMT*2018作为后期的NMT训练语料。在表5中的*topic*7行中*CWMT*2018作为初始化参数语料，*OurCorpus*作为后期的NMT的训练语料。当编码器由BERT初始化并且解码器由GPT初始化时，BLEU分数在四个测试集上都有所提升。并且本次实验结果都优于表4中的微调方法。通过这样的预训练方法比微调方法更有效地从预训练模型中获取更多知识。我们比较了两种不同语料作为初始化参数的方案中实验结果有所不同，在*topic*7中四个测试集上的实验结果总比*topic*6中的实验结果强，并且在新闻领域的测试集上明显提升了大幅度的BLEU值。我们认为，通过利用领域内的数据进行预训练并初始化，其次用不同领域的数据训练，这样不仅保留了原领域内的信息特征，同时更多的层次融合了外部领域内的知识，让模型获得了更好的性能。为此说明，预训练不仅能提升译文质量，而且本文所构建的数据质量是值得信赖。只有高质量的平行语料训练NMT，才能从输入句子中获取语义，获得更多的上下文信息从而提升增益。当*CWMT*2018作为初始化参数语料，*OurCorpus*作为后期NMT训练语料时，在藏译汉的*OurDev*测试集上相比*OurCorpus*作为初始化参数语料，*CWMT*2018作为后期的NMT训练语料提升了1.58个BLEU值，在汉译藏上提升了2.12个BLEU值。我们的数据不管在*CWMT*2018数据的上进行微调还是用*CWMT*2018数据初始化都取得不错的效果。

5.3 译文分析

为了更好的看到本文构建的平行语料库的效果，我们手动检查了*topic*1至*topic*7中的输出，并在表3中展示了一些示例。我们的*topic*7在四种测试集上都能较好的翻译出源文所含有的所有词。在表3中不难发现，当*CWMT*2018数据和*OurCorpus*数据合并后的训练模型最好，如表3中的第*topic*5行能够正常的把“毁损、灭失、承担、损害、赔偿以及责任”都能准确的翻译。

其次为预训练方法，用*CWMT*2018数据初始化参数并用*OurCorpus*数据进行训练，如表3中的第*topic*7行能够准确翻译“毁损、灭失、损害、赔偿”。随着数据集的增长，翻译质量不断提高。此外，使用领域外数据进行预培训有助于提高翻译质量，甚至在全集级别上也是如此。

藏文-汉文	句子
源文	དེན་ཚན་ སྒམ་བཀུ་ དང་ གསྒམ་པར་ སྐྱེལ་ འདེན་ཡིད་ རིང་འགྱུལ་པ་ ལ་ ཁེར་བ་ ཞེ་ ཅ་དངོས་ ཚག་སྐྱོན་ ཕྱུང་བ་ དང་ རྩ་འཕྲག་ ཏུ་ སོང་བ །ཚ་ལ་ ཐུམ་འཛོལ་ རོར་བ་ ཡེན་ན་ ཁྱོད་ཀྱིས་ རྒྱུ་གསལ་ ཀྱི་ འགན་འཛུར་ དགོས །
参考文	第 三百零三 条 运输 过程 中 旅客 自带 物品 毁损 、 灭失 ， 承运人 有 过错 的 ， 应当 承担 损害 赔偿 责任 。
TOPIC1	条 三百 和 第三 送 春风 中旅客 我 带 的 加快 损坏 发生 和 灭失 中 去 人夫 我 过错 丧失 的 损失 垮 补 的 负责 要
TOPIC2	第 三百零三 条 搬运 路途 中 旅客 自拿 东西 毁坏 、 灭失 ， 他人 有 错过 的 ， 应当 承担 损害 赔偿 责任 。
TOPIC3	第 三百零三 条 搬运 路途 中 旅客 自带 东西 毁坏 、 灭失 ， 运者 有 犯错 的 ， 相应 承担 损害 赔偿 责任 。
TOPIC4	第 三百零三 条 搬运 路途 中 旅客 自拿 东西 毁坏 、 灭失 ， 承运者 有 错误 的 ， 需要 承担 损害 赔偿 责任 。
TOPIC5	第 三百零三 条 运输 过程 中 旅客 自带 物品 毁损 、 灭失 ， 承运人 要 过错 的 ， 相应 承担 损害 赔偿 责任 。
TOPIC6	第 三百零三 条 运输 过程 中 旅客 自带 物品 毁损 、 灭失 ， 承运人 要 过错 的 ， 需要 承担 损失 补偿 义务 。
TOPIC7	第 三百零三 条 运输 过程 中 旅客 自带 物品 摧毁 、 灭失 ， 承运人 要 过错 的 ， 相应 承担 损害 补偿 责任 。

Figure 3: 藏译汉上同一条测试句在7个不同训练方法中的实验结果。

6 总结及未来工作

在本文中，我们已经证明，用CWMT2017藏汉平行语料库训练的基准模型在稀缺资源司法领域上可泛化性是有限。为此，我们针对稀缺资源司法领域的藏汉平行语料库，构建了一个高质量的藏汉平行数据集。我们利用本文所构建的数据集训练司法领域的NMT，极大地提高了翻译质量，同时我们发现随着数据集的增长，翻译质量也将不断提高。我们的数据集大小为160K个句子对，这也弥补了到目前为止公开的只有新闻领域CWMT数据的局限性。我们的数据集将会公开便于研究者使用，使得让研究少数民族语言信息处理快速发展。同时具有一个公开透明的可比性。在未来，我们计划针对藏汉语料的翻译模式进行一些语言知识调查。我们将稀缺资源司法领域语料库扩展到其他领域上，包括政治和教育等领域。我们通过一种领域来构造另外一种领域内的平行语料库。

参考文献

Zhang, Xiang and Zhao, Junbo and LeCun, Yann. 2015. *Character-level convolutional networks for text classification.* Advances in neural information processing systems.

Jiao, Xiaoqi and Yin, Yichun and Shang, Lifeng and Jiang, Xin and Chen, Xiao and Li, Linlin and Wang, Fang and Liu, Qun. 2019. *Tinybert: Distilling bert for natural language understanding* arXiv preprint arXiv:1909.10351

Gao, Fei and Zhu, Jinhua and Wu, Lijun and Xia, Yingce and Qin, Tao and Cheng, Xueqi and Zhou, Wengang and Liu, Tie-Yan. 2019. *Soft Contextual Data Augmentation for Neural Machine Translation* Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics

Li, Zhenhao and Specia, Lucia. 2019. *Improving Neural Machine Translation Robustness via Data Augmentation: Beyond Back Translation* arXiv preprint arXiv:1910.03009

Nag, Sreyashi and Kale, Mihir and Lakshminarasimhan, Varun and Singhavi, Swapnil. 2020. *Incorporating Bilingual Dictionaries for Low Resource Semi-Supervised Neural Machine Translation.* arXiv preprint arXiv:2004.02071.

Nishikawa, Sosuke and Ri, Ryokan and Tsuruoka, Yoshimasa. 2020. *Data Augmentation for Learning Bilingual Word Embeddings with Unsupervised Machine Translation*. arXiv preprint arXiv:2006.00262.

Alexander J. Ratner, Henry R. Ehrenberg, Zeshan Hussain, Jared Dunnmon, Christopher Ré 2017. *Learning to Compose Domain-Specific Transformations for Data Augmentation*

Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, Quoc V. Le. 2019. *AutoAugment: Learning Augmentation Strategies From Data*. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 113-123.

Wei, Jason W and Zou, Kai. 2019. *Eda: Easy data augmentation techniques for boosting performance on text classification tasks*. arXiv preprint arXiv:1901.11196.

Christian Buck and Philipp Koehn. 2016. *Findings of the wmt 2016 bilingual document alignment shared task*. In Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers, pages 554–563.

Lu ıs Gomes and Gabriel Pereira Lopes. 2016. *First steps towards coverage-based document alignment*. In Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers, pages 697–702.

Jonathon Read, Rebecca Dridan, Stephan Oepen, and Lars Jørgen Solberg. 2012. *Sentence boundary detection: A long solved problem?*. In Proceedings of COLING 2012: Posters, pages 985–994.

Sagar Shivaji Salunke. 2014. *Selenium Webdriver in Python: Learn with Examples*. CreateSpace Independent Publishing Platform.

Kingma, Diederik P and Ba, Jimmy. 2014. *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980。

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. *OpenNMT: Opensource toolkit for neural machine translation*. In Proceedings of ACL 2017, System Demonstrations, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. *The united nations parallel corpus v1. 0*. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pages 3530–3534.

Liu B, Huang L. 2020. *NEJM-enzh: A Parallel Corpus for English-Chinese Translation in the Biomedical Domain*. arXiv preprint arXiv:2005.09133, 2020.

William A Gale and Kenneth W Church. 1992. *A program for aligning sentences in bilingual corpora*. Computational linguistics, 19(1):75–102.

Robert C Moore 2002. *Fast and accurate sentence alignment of bilingual corpora*. In Conference of the Association for Machine Translation in the Americas, pages 135–144. Springer.

Rico Sennrich and Martin Volk. 2010. *Mt-based sentence alignment for ocr-generated parallel texts*. In The Ninth Conference of the Association for Machine Translation in the Americas (AMTA 2010).

Simard M, Plamondon P. 1998. *Bilingual sentence alignment: Balancing robustness and accuracy*. Machine Translation, 1998, 13(1): 59-80.

Repar A, Podpecan V, Vavpetič A, et al. 2019. *TermEnsembler: An ensemble learning approach to bilingual term extraction and alignment*. Terminology. International Journal of Theoretical and Applied Issues in Specialized Communication, 2019, 25(1): 93-120.

沙九; 冯冲; 张天夫; 郭宇航; 刘芳2020. 多策略切分粒度的藏汉双向神经机器翻译研究. 厦门大学学报(自然科学版)

Sun J 2012. *Jieba chinese word segmentation tool*. Accessed: Jun, 2012, 25: 2018.

李亚超, 江静, 加羊吉, 等. 2015. *TIP-LAS:* 一个开源的藏文分词词性标注系统. 中文信息学报, 2015, 29(6): 203-207.

Sennrich R, Haddow B, Birch A. 2015. *Neural machine translation of rare words with subword units*. arXiv preprint arXiv:1508.07909, 2015.

Vaswani A, Shazeer N, Parmar N, et al. 2017. *Attention is all you need*. Advances in neural information processing systems. 2017: 5998-6008.

Guo J, Tan X, Xu L, et al. 2019. *Fine-Tuning by Curriculum Learning for Non-Autoregressive Neural Machine Translation*. arXiv preprint arXiv:1911.08717, 2019.

Weng R, Yu H, Huang S, et al. 2019. *Acquiring Knowledge from Pre-trained Model to Neural Machine Translation*. arXiv preprint arXiv:1912.01774, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. *Bert: Pre-training of deep bidirectional transformers for language understanding*. arXiv preprint arXiv:1810.04805, 2018.

Lample G, Conneau A. 2019. *Cross-lingual Language Model Pretraining*. arXiv preprint arXiv:1901.07291, 2019.

Han L, Jones G J F, Smeaton A F. 2020. *MultiMWE: Building a Multi-lingual Multi-Word Expression (MWE) Parallel Corpora*. arXiv preprint arXiv:2005.10583, 2020.

一种基于相似度的藏文词同现网络构建及特征分析

加羊东周[1,3,4] **才智杰**[1,2,3,4] **才让卓玛**[1,2,3,4] **三毛措**[1,3,4]

1. 青海师范大学计算机学院，青海西宁 810016；

2. 西南民族大学计算机科学与技术学院，四川成都 610041；

3. 藏文信息处理教育部重点实验室，青海西宁 810008；

4. 青海省藏文信息处理与机器翻译重点实验室，青海西宁 810008

358521688@qq.com Czjqhsd@163.com cr-zhuoma@163.com 2627996852@qq.com

摘要

语言文字是人类智慧和文明的结晶，是经过漫长演化形成的复杂系统。语言同现网络采用复杂网络技术研究语言的特征，揭示语言文字的内部结构关系。文章通过分析相似性同现网络构建模块结构，提出一种基于相似度的藏文词同现网络构建方法，该方法以词为网络节点，以相似词间连边构造词同现网络。基于相似度藏文词同现网络构建方法，在大、中、小三类文档上建立了词同现网络，并分析了它们的统计特征，实验数据表明建立的藏文词同现网络都具有小世界效应和无标度特征。

关键词： 自然语言处理； 藏文； 词向量； 相似度； 同现网络

A Research on Construction and Feature Analysis of Similarity-based Tibetan Word Co-occurrence Networks

Jia Yang-dongzhou[1,3,4]**Cai Zhi-jie**[1,2,3,4]**Cai Rang-zhuoma**[1,2,3,4]**San Mao-cuo**[1,3,4]

1.College of Computer Science and Technology, Qinghai Normal University, Qinghai Xining 810016;

2.School of Computer Science and Technology,

Southwest Minzu University, Sichuan Chengdu 610041,China;

3.Tibetan Information Processing and Machine Translation Key Laboratory of Qinghai Province, Qinghai Xining 810008;

4.Key Laboratory of Tibetan Information Processing, Ministry of Education, Qinghai Xining 810008

358521688@qq.com Czjqhsd@163.com cr-zhuoma@163.com 2627996852@qq.com

Abstract

As the crystallization of human wisdom and civilization, Language is a complex system formed after a long evolution. Language concurrency network utilizes complex network techniques to study the linguistic features and reveal the internal structure of languages. In this work, we analyze the modular structure co-occurrence network and proposes a similarity-based method for constructing Tibetan word co-occurrence networks, in which Tibetan words serve as the nodes, and the similarity metrics among words serve

as the edges. We established of similarity-based word co-occurrence network on three types of documents in terms of size, namely, large, medium and small. and analyzed their statistical features. The experimental data indicated that the Tibetan word co-occurrence networks have small-world effects and scale-free features.

Keywords: Natural Language Processing , Tibetan , Word Embedding , Similarity , Co-occurrence Network

1 引言

语言文字是人类智慧和文明的结晶，是经过漫长演化形成的复杂系 (Steels, 2000)。语言同现网络采用复杂网络技术定量考察和分析语言的特性, 验证语言同现网络的小世界效应和无标度特征, 揭示语言的内部结构关系。词同现网络是语言同现网络的一种表现形式 (孙文俊等, 2010), 揭示词与词之间的内部结构关系。词同现网络的定义不同，其构建词方法也各不相同。词同现网络构建方法主要有 n 阶 Markov 同现模型和相似性同现模型 (才智杰, 2018) 等两种。由于 n 阶 Markov 同现模型理论相对成熟且操作便捷，成为构建词同现网络的常用方法。

近年随着神经网络技术的飞速发展，词向量表示性能得到了显著提升 (才智杰, 2020)，方便了词相似度的计算，从而为构建相似性词同现网络奠定了理论基础。为了研究相似性同现网络模型的构建技术及验证相似性同现模型下藏文词同现网络的小世界效应和无标度特征，揭示藏文词同现网络的内部结构，本文在已有藏文词向量表示的基础上，研究了相似性模型的藏文词同现网络构建技术，提出了一种基于相似度的藏文词同现网络构建方法，并分析了它们的统计特征，实验数据表明建立的藏文词同现网络都具有小世界效应和无标度特征。

2 研究现状

自 Cancho 和 Solé (2001) 首次将复杂网络的方法引入语言研究中，学者们开始针对不同的语言从不同层面研究语言网络。基于 n 阶 Markov 同现模型依据句子中两个词的 n 阶 Markov 链建立语言同现网络，通过上下文的顺序制约关系揭示词间的关系；相似性同现模型通过词之间的相似性建立网络，通过节点的相似度和上下文语义关系揭示词间的关系。Barabasi (2002) 采用 n 阶 Markov 同现模型建立了英文词的同现网络，梁伟 (2012)、林枫 (2012) 和刘知远 (2007) 等采用 n 阶 Markov 同现模型建立了汉文字/词的同现网络。梁伟等 (Wei et al., 2012; Liang et al., 2015; Liang et al., 2009) 从文学的视角，通过词同现网络对中国、英、美作品做了一系列的比较研究工作；耿志杰 (2010) 等构建了图书情报领域关键词同现网络，并进行了结构研究；余传明 (2010) 等利用网站评论数据构建情感词汇同现网络，并挖掘情感词汇之间的关系及内部规律；Liu (2011) 从地理系统科学数据中提取关键词进行词同现网络分析和可视化；He (2016) 通过构建 6000 首华语流行歌曲歌名的词同现网络，揭示流行歌曲独特的词同现网络特征；李亚星 (2016) 等根据微博语料的特点，通过词同现网络获取关联性强和具有潜在传播效应的词语；Atsushi Tsuya (2014) 用词同现网络分析癌症病人日常发布与疾病相关的推荐信息获得到对病人需求的深层理解。以上文献都以 n 阶 Markov 同现模型构建词同现网络，迄今为止未见有关采

用相似性同现模型构建词同现网络的文献报道。刘知远 (2008) 等采用相似性同现模型建立了汉语依存句法网络，Cancho 和 Motter (Cancho and Sole, 2001; Cancho et al., 2004; Motter et al., 2002) 采用相似性同现模型分别建立了英语句法网络和概念网。

少数民族语言同现网络的研究相对较少，才智杰 (2018) 等采用 n 阶 Markov 同现模型在诗歌、散文、政治、佛教、教材、口语等不同类型的语料上构建了 97 个藏文字同现网络，分析了藏文字同现网络的最短路径长度、聚类系数和度分布，实验数据显示 97 个藏文字同现网络都具有小世界效应和无标度特性，表明藏文字同现网络都具有小世界效应和无标度特性。藏文词同现网络的研究未见文献报到。本文在已有藏文词向量表示的基础上，研究了相似性模型的藏文词同现网络构建技术，提出了一种基于相似度的藏文词同现网络构建方法，该方法以词为网络节点，以相似的词间连边构造词同现网络。在大、中、小三类文档上建立了基于相似度藏文词同现网络，并分析了它们的统计特征，实验数据表明建立的藏文词同现网络都具有小世界效应和无标度特征。

3 基于相似度的藏文词同现网络构建

3.1 基于相似度的藏文词同现网络构建模块结构

在语言学领域，词与词之间的关系具有很强的规律，词同现网络的文本表示可以捕捉文本结构信息，揭示其内在的组织原则与语言学规律。近年来随着深度神经网络的飞速发展，词向量在自然语言处理领域得到了广泛应用。词向量作为处理下游任务的输入特征，使下游任务的性能得到了显著改进和提升。

相似性同现模型是通过词之间的相似性建立网络，网络中的节点为词，同一文档中最相似两词对应的节点间连接一条边。即同现网络 $G = \{V, E\}$，$V = \{v_i | v_i \in T\}$，$E = \{e_i | e_i = \mathbf{max_j sim(v_i, v_j)}\}$，其中 T 表示藏文词的集合。基于相似度的藏文词同现网络构建结构如图 1 所示。

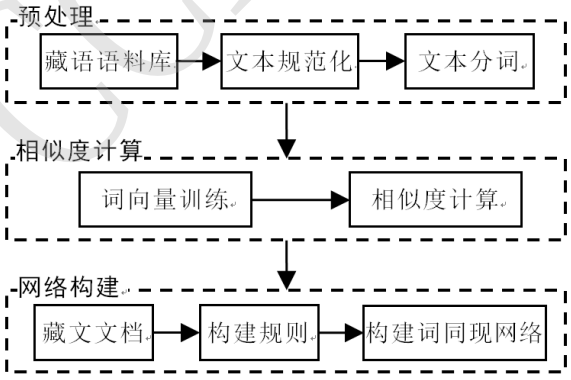


图 1: 基于相似度的藏文词同现网络构建结构

基于相似度的藏文词同现网络构建结构包含预处理模块、相似度计算及网络构建模块。预处理模块将语料库中的藏文本进行规范化和分词处理，通过规范化处理得到纯净的藏文文本数据，使用分词软件对藏文文本进行分词。相似度计算模块先训练词向量，将藏文词表示为向量，通过词向量计算出词之间的相似度。通常使用余弦相似度、欧氏距离计算词之间的相似度。词同现网络构建模块根据网络构建规则建立揭示词之间关系的网络，网络的节点为文档中的词，满足相似条件的两个词间连一条边。

3.2 基于相似度的藏文词同现网络构建规则

相似度计算模块和网络构建模块是基于相似度的藏文词同现网络构建的要素。词相似度的计算性能取决于词向量的效果，训练词向量的语料越大词向量效果越好，因此选用一个大语料训练词向量较合适 (我们用全集语料训练了词向量)。网络构建模块通过节点的选取和节点之间的连边规则构建网络，节点应该从当前文档中选取。连边规则决定词同现网络边的选择，揭示词之间的关系，是整个词同现网络构建的核心，称为词同现网络构建规则。

构建基于相似度词同现网络的规则中，需要考虑两个问题，其一是对于给定的节点词 A，如何选取该节点词的相邻节点词 X，其二是对于给定的节点词 A 已经选择了相邻节点词 B 的情况下，又选到相邻节点词 B 时该如何处理。问题一中相邻节点词 X 的选择可根据具体情况取与节点词 A 相似的前 n 个词。由于词的相似性具有传递性，当 $n=1$ 时具有 N 个节点词的词同现网络由 $\frac{N}{2}$ 个连通子图组成 (如图 2 所示)，不符合实际需求，因此在实际构建词同现网络时 $n \geq 2$ 较合适。

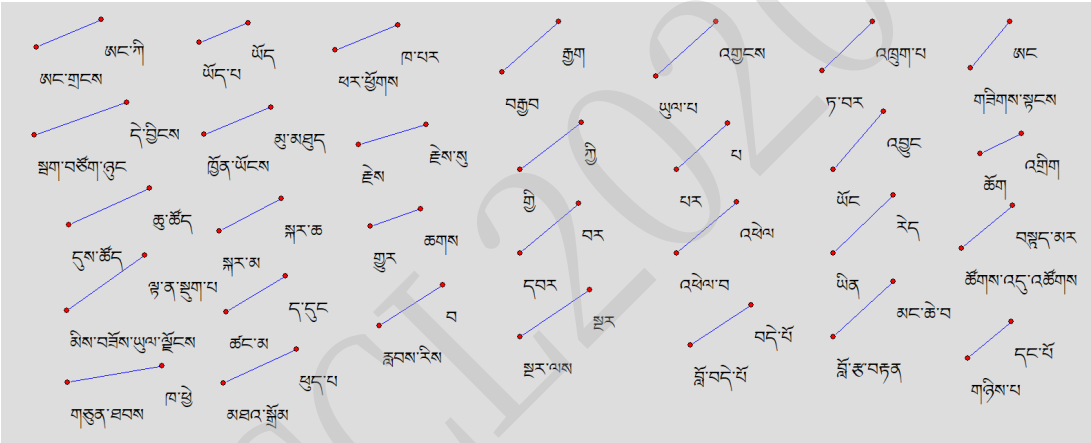


图 2: $n=1$ 时的词同现网络示意图

对于给定的节点词 A 已经选择了相邻节点词 B 的情况下，又选到相邻节点词 B 时有两种处理方法：第一种处理方法是忽略相邻节点词 B，节点词 A 和与它相邻的 $n-1$ 个相邻节点相连；第二种方法是忽略相邻节点词 B，增加与节点词 A 第 $n+1$ 个相似的词 C 为相邻节点，从而保持每个与节点 A 相邻的节点数为 n。在构建基于相似度词同现网络时选择第二种方法较合适。事实上，通过实验观察到采用第一种处理方法会使后继节点词的邻接节点越来越少，从而使词同现网络变为边稀疏网络，不能更好的反映词之间的内部结构关系。通过以上讨论可得以下基于相似度的藏文词同现网络构建规则和算法。

基于相似度的藏文词同现网络构建规则: 选用一种比较好的词向量训练方法和大的语料训练得到一个性能好的词向量表用于计算词相似度，给定文档中的每个词为网络节点 v_i，求出与节点 v_i 最相似度的 n 个节点 $u_j(j=1,2,...,n)$。若这 n 个节点 u_j 都与 v_i 不相邻，则连接 v_i，u_j，即 $(v_i\ u_j)$ 加入 E; 否则用节点 v_i 的第 n 个相似节点之后且与 v_i 未连接的点 u_k 替换节点 u_j。具体算法如下:

算法 1 基于相似度的藏文词同现网络构建算法

输入: 词向量矩阵 M^{N*D}，选定的藏文文本 T;

输出: 基于相似度的藏文词同现网络;

for $i = 1 \rightarrow |V|$ **do**
  for $j = 1 \rightarrow |V|$ **do**
    if $v_i == v_j$ **then**
      $similarity\_list[j-1] = 0$ v_i 和 v_j 为同一个词时, 相似度置为 0
    else
      $x \leftarrow B_{v_iM}$ 取 x 为 v_i 的词向量，B_{v_i} 为 v_i 的 one-hot 向量
      $y \leftarrow B_{v_jM}$ 取 y 为 v_j 的词向量，B_{v_j} 为 v_i 的 one-hot 向量
      $similarity\_list[j-1] \leftarrow similarity(x, y)$ 计算相似度并保存到数组中
    end if
    设置 n
    while $j < n$ **do**
      $index \leftarrow argmax(similarity\_list)$ 取出与 v_i 相似度最高值的下标
      $u_j \leftarrow V[index]$
      $similarity\_list[insex] = 0$
      if $(v_i, u_j) \in Eor(u_j, v_i) \in E$ **then**
        $n = n + 1$
        $j = j + 1$
      else
        $E \leftarrow (v_i, u_j)$
        $j = j + 1$
      end if
    end while
    $G \leftarrow (V, E)$
  end for
end for

3.3 基于相似度的藏文词同现网络构建

在构建藏文词同现网络时我们从青海师范大学建立的藏语语料库中选取了 18.07M 大小的语料，对其进行了预处理 (才智杰, 2018; 才智杰等, 2011)，将其作为词向量训练语料。词向量训练语料（下文称全集语料）信息见表 1。

	文学	政论	藏医	共计
大小	6.64M	8.53M	2.90M	18.07M
词条数	485815	542230	230935	1258980

表 1: 词向量训练语料信息表

由于藏文语料库相对较小, 因而我们参考文献 (才智杰等, 2019) 选用了在小规模语料库中表

现较好的 CBOW 模型训练词向量。CBOW 模型参数见表 2。

Dimsize	Window	alpha	Iter	hs	TWordSim215
300	5	0.025	500	0	47.67

表 2: CBOW 模型参数表

词向量的相似度通常由余弦相似度、欧氏距离表示，余弦相似度更能刻画向量间的相似程度。我们在计算相似度时采用词向量的夹角余弦值来评估词间的相似度，余弦相似度计算公式如下:

$$\mathbf{CosineSimilarity(u,v)} = \frac{\mathbf{u}\cdot\mathbf{v}}{\|\mathbf{u}\|_2\|\mathbf{v}\|_2} \tag{1}$$

其中 $u \cdot v$ 是两个向量的点积，$\|u\|_2$ 是向量 u 的范数。余弦相似度的取值在 $[0,1]$，当余弦相似度的值越大表示两个向量越相似。

为了观察词同现网络的效果及特征分析，我们选用了大、中、小三个文档采用 2.2 节的词同现网络构建规则构建了藏文词同现网络，其中 n 取 2。大文档指用于训练词向量的大小为 18.07M 全集语料，中文档指从大文档中文学、政论、藏医等三类中各取选了 50% 而得到的大小为 8.54M 的语料，小文档指从大文档中任意抽取的大小为 2.08M 的语料。构建同现网络文档信息见表 3，构建的词同现网络如图 3 所示。

	大文档	中文档	小文档
大小 (M)	18.07	8.54	2.08
词条数	1258980	599407	165740

表 3: 构建词同现网络语料信息表

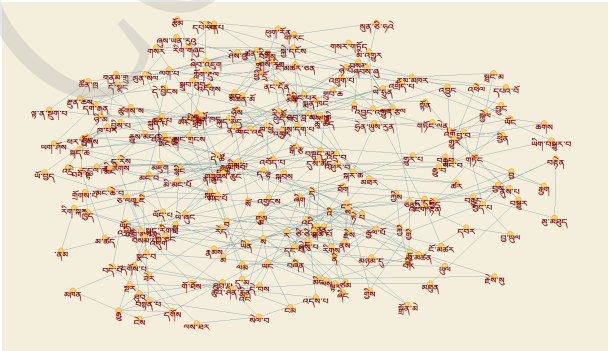


图 3: 藏文词同现网络示意图

4 藏文词同现网络的特征分析

4.1 小世界效应

为了从多方位考察藏文词同现网络的特征，我们对大、中、小三类文档 (文档信息见表 3，网络构建规则中 $n = 2$) 建立的基于相似度的藏文词同现网络利用 Pajek 网络分析工具进行了特征

统计分析, 同现网络基本数据对比表如表 4 所示，其中藏文字同现网络的统计参数来自文献 (才智杰, 2018)，汉文字同现网络的统计参数来自文献 (梁伟等, 2012)，汉文词同现网络的统计参数来自文献 (刘知远等, 2007)。

类型		N	E	D	$< k >$	L	Lr	$C(\%)$	$Cr(\%)$
藏文字 (均值)		3194	41943	7	26.2636	2.5644	2.4690	11.5398	0.8225
汉文字 (均值)		4520	96512	9	42.7000	2.4900	2.2400	38.0700	0.9500
汉文词		157000	8300000	–	64.35	2.63	2.99	0.619	0.00025
藏文词	大文档	38636	77272	15	4	7.4910	7.6204	0.2075	0.000048
	中文档	24047	48094	13	4	7.1841	7.2784	0.1691	0.000055
	小文档	12371	24742	13	4	6.8142	6.7987	0.1280	0.000017

表 4: 同现网络基本数据对比表

表中 N 表示词同现网络的顶点数、E 表示边数、D 表示直径、$< k >$ 表示平均度、L 表示平均最短路径长度、Lr 表示平均最短路径长度参照系数、C 表示平均聚类系数、Cr 表示平均聚类系数参照系数。

以上的实验数据体现了基于相似度的藏文词同现网络的以下特征:

(1) 在藏文词同现网络的统计参数中所有统计参数比较稳定，只是随语料大小的变化有小的波动，并不随语料大小的变化而有较大的变化，说明选取语料规模的大小对基于相似度的藏文词同现网络的统计参数没有太大的影响。

(2) 直径 D 的值在小语料集和大语料集上几乎相同，比汉文字/词、藏文字的大；藏文词的平均度 $< k >$ 都为 4，远远小于汉文字/词、藏文字的平均度。说明基于相似度的藏文词同现网络是一种边稀疏网络，相似词之间的关联度较弱。

(3)3 个藏文词同现网络都具有小的平均最短路径 L，且 $L \approx Lr$，$C \gg Cr$，说明基于相似度的藏文词同现网络具有小世界效应。

4.2 无标度特性

我们分析了构建的 3 个藏文词同现网络的度分布情况，与其他语言网络的度分布类似，网络中少数的节点往往拥有大量的连接，而大部分节点却很少，这些大多数点对无标度网络的运行起着主导作用，呈现“胖尾”现象。说明藏文词同现网络的度分布服从幂律分布，显示了无标度特性。双对数坐标下三类文档的词同现网络度的分布见图 4。

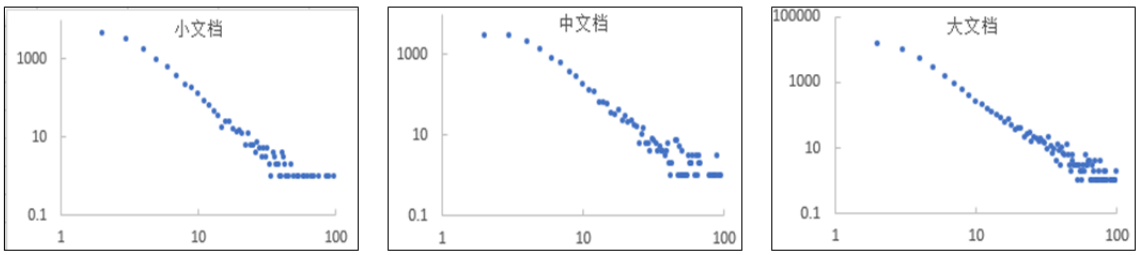


图 4: 双对数坐标下三类文档的度分布图

5 结论

语言同现网络通过复杂网络的方法研究语言网络的特征，有助于揭示语言文字的内部结构关系。词同现网络是语言同现网络的一种表现形式，其构建方法主要有 n 阶 Markov 同现模型和相似性同现模型等两种。学者们已从不同角度研究了基于 n 阶 Markov 同现模型的词同现网络构建方法，并对英汉词同现网络的特征进行了分析。近年随着神经网络技术的飞速发展，词向量表示性能得到了显著提升，方便了词相似度的计算，为构建基于相似性词同现网络奠定了理论基础。

为了研究相似性词同现网络技术及揭示藏语词同现网络的小世界效应和无标度特性，我们研究了藏文词同现网络构建方法，提出了一种基于相似度的藏文词同现网络构建方法，该方法以词为网络节点, 以相似的词间连边构造词同现网络。在大、中、小三类文档上建立了词同现网络，并分析了它们的统计特征，实验数据表明建立的藏文词同现网络都具有小世界效应和无标度特征。今后在该研究成果的基础上进一步研究藏文构件、字、词的超复杂网络构建技术及统计特征分析。

致谢

本项工作得到了国家自然科学基金资助项目（61866032,61966031), 青海省科技厅资助项目(2019-SF-129), “长江学者和创新团队发展计划”创新团队资助项目（IRT1068），青海省重点实验室项目（2013-Z-Y17、2014-Z-Y32、2015-Z-Y03），藏文信息处理与机器翻译重点实验室（2013-Y-17）资助。

参考文献

Steels L. 2000. *Language as a complex adaptive system.* Proceedings of Lecture Notes in Computer Science. Berlin: Springer-Verlag.

孙文俊，杜娟. 2010. 基于词同现网络与支持向量机的论文甄别. 现代情报, 2010(07):89-94.

才智杰, 孙茂松, 才让卓玛. 2018. 藏文字同现网络的小世界效应和无标度特性. 中文信息报,32(10):45-52.

才智杰,才让卓玛,孙茂松. 2020. 一种多基元联合训练的藏文词向量表示方法. 中文信息报,2020,34(5):44-49.

Ramon Ferrer i Cancho and Ricard V. Solé. 2001. *The Small World of Human Language.* Proceedings Biological Sciences, 268(1482):2261-2265.

Barabasi A L. 2002. *The New Science of Neworks.* Massachusetts,Persus Publishing.

梁伟，史玉明. 2012. 不同时期汉语散文的字同现网络之研究. 中国科学: 信息科学,42(7):831–842.

林枫，刘云，江钟立. 2012. 汉字网络的历时性模式探析. 复杂系统与复杂性科学,9(3):50-61.

刘知远，孙茂松. 2007. 汉语词同现网络的小世界效应和无标度特性. 中文信息学报,21(6):52-58.

Wei and Liang and YuMing. 2012. *Study on co-occurrence character networks from Chinese essays in different periods.* Science China Information Sciences,55(11):2417-2427.

Liang W and Wang Y and Shi Y. 2015. *Co-occurrence network analysis of Chinese and English poems.* Physic A: Statistical Mechanics and its Applications, 420:315-323.

Liang W and Shi Y and Tse C K. . *Comparison of co-occurrence networks of the Chinese and English languages.* Physic A: Statal Mechanics and its Applications, 388(23):4901-4909.

耿志杰, 王文鼐. 2012. 关键词同现网络结构研究. 情报杂志, 29(2):14-16.

余传明，周丹. 情感词汇共现网络的复杂网络特性分析. . 情报学报,29(5):906-914.

Liu R and Zhao H. 2014. *2011 International Conference on Management and Service Science-Word Co-Occurrence Network Analysis of Scientific Data Using NWB Tool.* IEEE 2011 International Conference on Management and Service Science (MASS 2011).

He B and Xu D. 2016. *An exploration on the word co-occurrence network of Chinese popular song titles..* International Conference on Natural Computation & Fuzzy Systems & Knowledge Discovery.

李亚星，王兆凯，冯旭鹏. 2016. 基于实时词共现网络的微博话题发现. 计算机应用,309(05):130-134.

Tsuya A and Sugawara Y and Tanaka A. 2014. *Do Cancer Patients Tweet? Examining the Twitter Use of Cancer Patients in Japan.* Journal of Medical Internet Research,16(5):e137.

刘知远，郑亚斌，孙茂松. 2008. 汉语依存句法网络的复杂网络性质. 复杂系统与复杂性科学,5(2):37-45.

Cancho R F I and Sole R V. 2001. *The Small World of Human Language.* Proceedings of the Royal Society of London Series B-Biological Sciences,268(1482):2261-2265.

Cancho R F I and Sole R V and Kohler R. 2004. *Patterns in Syntactic Dependency Networks.* Phys Rev E, 69(5):1915.

Motter A E and de Moura A P S and Lai Y C. 2002. *Topology of the Conceptual Network of Language.* Phys Rev E, 65(6):102.

才智杰，才让卓玛. 2011. 藏文自动分词系统的设计. 计算机工程与科学,33(5):151-154.

才智杰, 孙茂松, 才让卓玛. 2019. 藏文词向量相似度和相关性评测集构建. 中文信息学报,33(7):81-87,100.

《动词句法语义信息词典》知识内容说明书

袁毓林
北京大学中文系/ 北京
中国语言学研究中心/ 北京
计算语言学教育部重点实验室/ 北京
yuanyl@pku.edu.cn

曹宏
北京大学考古文博学院/ 北京
赛克勒考古与艺术博物馆/ 北京
caohong@pku.edu.cn

摘要

本文首先介绍《实词信息词典》的研制目标与结构内容，重点介绍其中的《动词信息词典》的体系结构与理论背景；然后，介绍《动词信息词典》所区分的8种动词小类及其定义，其为动词所设置的22种语义角色及其定义，由这些语义角色的不同配置而造成的20来种句法格式及其例句，其所考察的动词的9种主要的语法功能及其对于该词类的隶属度；最后，给出《动词信息词典》中检索系统的界面截图，交代其相应的纸质版本的情况。

关键词：《实词信息词典》；《动词信息词典》；动词小类；语义角色；句法格式；语法功能

An Introduction to the Syntactic-Semantic Knowledge-Base of Chinese Verbs

Yuan Yulin
Department of Chinese Language and Literature,
Peking University / Peking
Center for Chinese Linguistics, PKU / Peking
Ministry of Education Key Laboratory of
Computational Linguistics, PKU / Peking
yuanyl@pku.edu.cn

Cao Hong
School of Archaeology
and Museology,
Peking University / Peking
Arthur M. Sackler Museum
of Art and Archeology,
Peking University / Peking
caohong@pku.edu.cn

Abstract

The Knowledge-Base of Content Words, an integrated knowledge system of Chinese linguistic resources, is composed of three sub-systems: the Knowledge-Base of Adjectives (KB@adjecitve), the Knowledge-Base of Verbs (KB@verb) and the Knowledge-Base of Nouns (KB@noun). This paper focuses on KB@verb. First, the structural system and theoretical foundation of KB@verb are introduced. Secondly, KB@verb classifies verbs into eight sub-classes and defines 22 semantic roles, which are configured into decades of syntactic formats. These syntactic formats and their examples taken from real-world texts are also included. In addition, KB@verb identifies nine major grammatical functions of verbs and their degrees of membership. Finally, the retrieval system and the paper version of KB@verb are briefly illustrated or introduced.

Keywords: Knowledge-Base of Content Words , Knowledge-Base of Verbs (KB@verb) , sub-classes of verbs , semantic role , syntactic format , grammatical functions

1 《实词信息词典》的研制目标与结构内容

由袁毓林教授主持的《北京大学现代汉语实词句法语义功能信息词典》，从1998年开始研制，现在已经初步完成。下面简单介绍该词典的知识内容、组织结构及其理论背景。

《北京大学现代汉语实词句法语义功能信息词典》（简称《实词信息词典》）是一个电子化的语言知识资源，知识内容主要是现代汉语常用形容词、动词和名词的句法功能、语义角色及其组配方式、主要句型及其典型例句；并且，配备完善与方便的检索系统。《实词信息词典》主要是为汉语自动语义分析和文本生成、汉语国际教育与研究而研制的，可以为汉语的理论研究、教学应用和信息处理工程提供语言知识资源。

《实词信息词典》主要根据袁毓林教授30多年来，在认知科学的背景上，对于现代汉语的下列研究成果：（1）汉语词类的模糊划分与隶属度计算，（2）汉语配价语法研究、汉语动词与形容词的论元结构研究，（3）汉语生成词库论研究、汉语名词的物性结构研究。该词典特别重视词语之间的搭配关系和选择限制，通过大量的例子来展示目标词的意义和用法，并且提供目标词的搭配习惯和基本句型。这样做的一个理由是，正如英国语言学家J. R. Firth（1890-1960）所说：You shall know a word by the company it keeps.（欲知其词，先观其伴）Each word when used in a new context is a new word.（每一个单词出现在不同的上下文中就是一个新的单词）。的确，要了解一个词的意义和用法，最好的办法莫过于观察它跟什么样的词语搭配。并且，同一个词在不同的语境中不同程度的意义变化，也只能通过它的搭配环境来显示。这直接为汉语的国际教学提供了汉语常用实词的情境意义和搭配组合的范例，也为基于词语共现（co-occurrence）的机器学习方法，提供了一种精炼的语料库和训练样本。

《实词信息词典》分为下面三个既相对独立、又相互链接的子系统：

1. 《汉语形容词句法语义功能信息词典暨检索系统》（简称《形容词信息词典》），收入常用形容词3千多个，4千多个义项条目；信息内容包括：词条、拼音、次类（形容词、状态词）、释义（包括：语体色彩、同义词和反义词及其链接关系等）、语义角色集合及其定义、由目标形容词和其论旨角色组配成的句法格式和相应例句、各种主要的语法功能及其词类隶属度，等等。

2. 《汉语动词句法语义功能信息词典暨检索系统》（简称《动词信息词典》），收入常用动词1万2千多个，1万6千来个义项条目；信息内容包括：词条、拼音、次类（8种动词小类）、释义（包括：语体色彩、同义词和反义词及其链接关系等）、语义角色集合及其定义、由目标动词和其论旨角色组配成的句法格式和相应例句、各种主要的语法功能及其词类隶属度，等等。（检索系统的界面图见文末）

3. 《汉语名词句法语义功能信息词典暨检索系统》（简称《名词信息词典》），收入常用名词2万多个，2.5万来个义项条目；信息内容包括：词条、拼音、释义（包括：语体色彩、同义词和反义词及其链接关系等）、物性角色集合及其词例（及其跟相关动词和形容词的链接关系等）、由目标名词和其物性角色组配成的句法格式和相应例句、各种主要的语法功能及其词类隶属度，等等。

下面主要介绍《动词信息词典》的知识内容和体系结构。

2 《动词信息词典》的内容结构和理论背景

《汉语动词句法语义功能信息词典暨检索系统》（简称《动词信息词典》）主要描写动词跟其他词语的搭配方式和语义上的句型构造。通过语料调查，我们可以发现，动词在句子中主要做谓语或谓语核心，表示人或事物的某种动作、行为、状态或关系；跟它搭配的词语表示这种动作、行为、状态或关系的主体，或者受到这种动作、行为影响的事物，以及工具、方式、处所等其他相关的事物。简单地说，动词的有关搭配成分是论元；论元又可以根据它们在动词所表示的事件中的作用（扮演的角色），分为不同的语义角色。

这样，了解一个动词跟什么样的语义角色搭配、它们怎样搭配、它们组合起来以后描述了一种什么样的情景，也就基本掌握了一个动词的意义和用法。难怪美国著名的认知心理学

本课题的研究得到教育部人文社会科学重点研究基地重大研究项目“汉语意合语法框架下的词汇语义知识表示及其计算系统研究”（项目编号：18JJD740003）和国家社科基金重大项目《基于“互联网+”的国际汉语教学资源与智慧教育平台研究》（项目编号：18ZDA295）的资助，谨此致以诚挚的谢意。

家Steven Pinker要说：“一个动词不仅仅是一个用于指称动作或状态的词，它实际上是句子的‘底盘’。它为该句子的其他成分提供了一个多槽的框架，在这里，无论主语、宾语还是各种介词宾语和从句等成分都可以各就各位，各司其职”。事实的确如此，请看例子：

（1）吐鲁番的葡萄熟了。——（主体+动词）
（2）萧何月下追韩信。——（施事+处所+动词+受事）
（3）我们向幸福进发。——（施事+目标+动词）
（4）黄鼠狼给鸡拜年。——（施事+对象+动词）
（5）我的婚姻我做主。——（范围+施事+动词）
（6）我家住在黄土高坡。——（主体+动词+处所）
（7）赵普用半部《论语》治天下。——（施事+工具+动词+受事）
（8）爹爹送我两尺红头绳。——（施事+动词+与事+受事）

从括号中对这些句子的简单的语义分析可见，意义和用法不同的动词能够跟不同的语义角色、通过不同的配置方式，造成不同的句式；从而，表达了不同的意义、描绘了不同的场景、讲述了不同的故事、报道了不同的事件。因此，可以直观地把动词想象成各种带钩子的原子，不同的动词所带的钩子的数量和形状不尽相同；于是，它们所能勾住的其他成分的数量和性质也就有所不同。如果借用化学上原子化合和配价的说法，那么动词就是语句组合的核心，像施事、主体、受事、范围、处所、工具等伴随成分就是配价成分。不同的动词有不同的配价功能，支配不同数量和不同性质的配价成分，构成不同形式的短语和句子。

《动词信息词典》就是从这样一种“情境语义学”（situation grammar）和“配价语法”（valence grammar）的角度，通过对大规模真实文本语料的调查和分析，来全面、准确、简明地描写动词在情境意义和搭配用法上的关键性特点，使读者“观其伴，会其意；明其价，知其用”；即让读者在查阅一个动词的条目以后，可以了解该动词通常跟哪些伴随成分一起出现，从而从搭配关系上理解该动词的意义、明白其配价组合方面的特点、并掌握其基本的常用句式；进而根据这些句式、模仿相关实例，可以依葫芦画瓢，万无一失地造出合语法、有意义的句子来。我们希望本词典对于人们理解相关句子的骨架意义具有指导性，从而为语言研究者、中文信息处理研究者，提供汉语句子的语义关系的基本数据；并对于中小学生和学习汉语的留学生的阅读理解和遣词造句，提供切实有效的帮助。

下面，介绍这部词典的动词小类、语义角色、句法格式和语法功能的描写体系。

3 《动词信息词典》的动词小类及其定义

这里的动词是根据其用法（即语法功能）来定义的，指具有下面这些语法功能的词：（1）可以作谓语或谓语核心，（2）可以受否定词“不、没有”修饰，（3）可以带真宾语或者不能受“很”等程度副词修饰。为了反映动词内部不同的词之间在有关用法上的差别，我们又把动词分为八种小类：<不及物动词>、<准及物动词>、<体宾动词>、<助动词>、<形式动词>、<名动词>、<强谓宾动词>、<弱谓宾动词>。

在本词典中，对于每一个动词词条，我们首先根据它们能否带真宾语而区分为“及物动词vs.不及物动词”两大类：可以带真宾语的是及物动词，不能带真宾语、并且只有一个必有性配价成分的是不及物动词，虽然不能带真宾语、但是另有一个需要介词引导的必有性配价成分的是准及物动词。如果是不及物动词，那么直接标上<不及物动词>；如果是准及物动词，那么直接标上<准及物动词>。

如果是及物动词，我们根据它们能否带谓词性宾语而区分为“谓宾动词vs.体宾动词”两大类：可以带谓词性成分作宾语的是谓宾动词，不能带谓词性成分作宾语的是体宾动词。如果是体宾动词，那么直接标上<体宾动词>。如果是谓宾动词，那么还要细分为下列五种小类：“强谓宾动词vs. 弱谓宾动词vs. 助动词vs. 形式动词vs. 名动词”。粗略地说：只能带谓词性成分作宾语、表示情态意义的是助动词，直接标上<助动词>。只能带名动词、事件名词作宾语的是形式动词，直接标上<形式动词>。可以兼作名词的是名动词，直接标上<名动词>。剩下的谓宾动词，只能带谓词性成分作宾语的是强谓宾动词，标上<强谓宾动词>；既能带谓词性成分作宾语、又能带体词性成分作宾语的是弱谓宾动词，标上<弱谓宾动词>。

各种动词小类的具体定义和判定标准如下：[-1]

[-1]参考朱德熙（1982）第56-61页。

0. 动词：可以作谓语或谓语核心、可以受“不、没有”修饰、可以带真宾语或者不能受“很”等程度副词修饰的词。分为如下八种小类：

1. 不及物动词：不能带真宾语、可以带准宾语的动词。

准宾语包括下列三种：

（1）量度宾语，即表示时量、动量或程度的宾语。例如：

[休息了]一会儿 | [醒了]好几次 | [绣了]一点儿

（2）处所宾语，即表示运动的终点的宾语。例如：

[来]北京 | [去]贵州 | [飞]乌鲁木齐 | [到]亲戚家

（3）存现宾语，即表示存在、出现或消失的宾语。例如：

[来了]两个新队员 | [走了]几个客人 | [新到了]一批冻猪肉

2. 准及物动词：不能带真宾语、但是可以通过介词来引导与事、对象等必有性配价成分的动词。[0]

准及物动词的配价成分，常见的包括下列五种：

（1）与事，即表示动作行为的协作者。例如：

[跟]李红[结婚] | [和]对手[较量] | [与]浙江[接壤]

（2）对象，即表示动作行为所针对的对象。例如：

[向]朋友[求助] | [给]师傅[拜年] | [为]儿子[说情] | [怪罪] [于]他人

（3）施事，即表示动作行为的发出者。例如：

[由]王平[接手] | [由]老板娘[主事] | [由]我爸爸[做东]

（4）处所，即表示运动的起点或终点等地点。例如：

[起源] [于]唐朝 | [出生] [在]苏州 | [毙命] [于]海外 | [在]西安[落户]

（5）依凭，即表示动作行为所依靠或凭借的事物。例如：

[以]音乐[见长] | [以]功臣[自居] | [以]失败[告终]

3. 体宾动词：不能带谓词性宾语、可以带由体词性成分充当的真宾语的动词。

谓词包括动词、性质形容词、状态形容词和谓词性代词等。

体词包括名词、时间词、处所词、合成方位词、数量词、体词性代词等。例如：

骑（马） | 买（菜） | 捆（行李） | 喝（一盅） | 驾驶（汽车） | 修理（拖拉机） | 招待（他们俩）

4. 强谓宾动词：只能带谓词性宾语、不能带体词性宾语的动词。例如：

打算（买房子） | 企图（越狱逃跑） | 希望（大家都高兴） | 觉得（这样做不妥当） | 以为（没有人知道） | 认为（老子天下第一）

5. 弱谓宾动词：既可以带谓词性宾语、又可以带体词性宾语的动词。例如：

看（下棋/立体电影） | 喜欢（出去旅游/中式服装） | 赞成（走出国门/他的意见） | 反对（盲目地扩大产能/厂长提出的融资方案） | 考虑（买一台电脑/这种办法）

6. 助动词：只能带谓词性宾语、不能带体词性宾语的表示情态意义的动词。又叫能愿动词，情态动词。例如：“能、能够、会、可以、可能、得、要、敢、想、应该、应当、该、愿意、情愿、乐意、肯、许、（不）配、值得”等。它跟不表示情态意义的其他谓宾动词的不同的语法特点是：（1）不能重叠，（2）不能带时体助词“着、了、过”，（3）可以放在“不 不”格式里。

7. 形式动词：只能带名动词、事件名词或以它们为核心的体词性偏正结构为宾语的动词。又叫傀儡动词或准谓宾动词。下面是“形式动词+名动词”的例子：

有（准备/保障） | 作（调查/分析） | 进行（处理/研究） | 加以（鉴别/管束） | 给以（打击/报复） | 受到（批评/监控） | 予以（表扬/追究）

下面是“形式动词+事件名词”的例子：作（手术） | 进行（手术/战争）

8. 名动词：可以作形式动词的宾语，可以修饰名词或受名词修饰的动词。又可以称为名动兼类词。例如：

[有]准备/保障 | [作]调查/分析 | [加以]鉴别/管束 | [给以]打击/报复 | [受到]批评/监控 | [予以]表扬/追究

[0]参考袁毓林（2010a）第281-284页。

4 《动词信息词典》的语义角色系统及其定义

动词的各种伴随成分（即配价成分，又叫论元），根据它们跟动词在意义上的不同的关系，可以区分为不同的语义角色。在本词典中，动词的配价成分首先分为必有论元和非必有论元两种，前者是构成意思相对完整的句子所不可缺少的，后者则用以扩充句子的意思，帮助形成意思相对复杂的句子。必有论元可以分为主体论元和客体论元两种，前者主要作主语，后者主要作宾语。主体论元可以细化为施事、感事、经事、致事、主事等语义角色，客体论元可以细化为受事、与事、结果、对象、系事等语义角色。非必有论元可以从语义上分为依凭论元、环境论元和关涉论元三种，它们主要作状语。其中，依凭论元可以细化为工具、材料、方式、原因、目的等语义角色，环境论元可以细化为时间、处所、源点、终点、路径等语义角色，关涉论元可以细化为量幅、范围等语义角色。

上述22种语义角色的定义可以大致规定如下：

（1）施事（agent，简写为A）：动词所表示的自主性动作行为的施行者。例如：

鸟儿飞了　医生来了　奶奶杀了一只鸡　爸爸买了一本书

在上面的例子中，“鸟儿”是发出“飞”这种动作的施事，“奶奶”是进行“杀（鸡）”这种行为的施事。

（2）感事（sentient，简写为SE）：动词所表示的心理感觉、精神体验等事态的感知性的主体。例如：

我们看电影｜爸爸认识李小龙｜同事们赞成这个方案｜大家讨厌这个人

在上面的例子中，“我们”是“看”这种感知行为的感事，“大家”是体验了“讨厌”这种心理行为的感事。

（3）经事（experiencer，简写为EX）：动词所表示的某种变化的具有感知性的主体。例如：

同学们都毕业了｜爷爷去世了｜我碰到几个熟人｜布什当选为美国总统

在上面的例子中，“同学们”是经历“毕业”这种变化性事件的经事，“我”是经历了“碰到”这种行为的经事。

（4）致事（causer，简写为CAU）：某种致使性事件的引起者，即造成某种后果的致使性因素。例如：

大雨阻断了山区的交通｜这一举措改变了公司的形象｜罢餐事件引起了校方的注意|工厂倒闭的消息把大伙儿吓坏了

在上面的例子中，致事“大雨”是造成“山区的交通”发生“阻断”的引起因素，致事“罢餐事件”是“引起”“校方的注意”的致使性因素。

（5）主事（theme，简写为TH）：动词所表示的性质、状态、关系或变化等事态的非感知性的主体。例如：

锅里的水开了｜村后的石桥塌了｜小孩掉水沟里了｜我爸爸长了一个疖子

在上面的例子中，“锅里的水”是“开”这种状态的主事，“小孩”是“掉”这种变化的主事。

（6）受事（patient，简写为P）：因施事或致事的行为而受到影响的事物。例如：

小猫逮耗子　爷爷喝葡萄酒　老师批评了王平　弟弟打碎了那面镜子

在上面的例子中，“耗子”是受到“逮”这种动作影响的受事，“王平”是遭受“批评”这种行为影响的受事。

（7）与事（dative，简写为D）：动词所表示的动作、行为的非主动的参与者。例如：

老板对雇员发火｜我们向当事人打听了一下｜舅舅给了小明一本词典｜同学们请教李老师一个地理方面的问题

在上面的例子中，与事“雇员”是“发火”这种事件的被动的参与者，与事“李老师”是“请教”这种事件的非主动的参与者。

（8）结果（result，简称R）：由施事或致事的动作、行为造成的结果。例如：

爸爸又挖了一个菜窖｜妈妈织了一件毛衣｜猫咪在桌子上踩了一串脚印｜弟弟把窗户纸捅了一个大窟窿

在上面的例子中，“一个菜窖”是“挖”这种动作的结果，“一串脚印”是“踩”这种行为的结果。

（9）对象（target，简写为TA）：动词所表示的心理感觉、精神体验等感知性行为的对象和目标，有时可以用介词“对、对于”等引导。例如：

我爸爸认识吴校长 | 妹妹喜欢芭蕾舞 | 我奶奶居然相信通灵术 | 陈一平很熟悉广告业务~陈一平对广告业务很熟悉 | 我哥很热爱本职工作 我哥对本职工作很热爱

在上面的例子中，感事“我爸爸”所“认识”的对象是“吴校长”，感事“妹妹”所“喜欢”的对象是“芭蕾舞”。

（10）系事（relative，简写为RE）：在动词所表示的状态、关系等事态里，跟主事、与事等相对的事物，一般表示相应于主事、与事等的属性、类型等。例如：

李四光是地质学家 | 徐先生有两个女儿 | 大家叫鞠萍知心姐姐 | 通州属于北京

在上面的例子中，系事“地质学家”表示主事“李四光”的性质，系事“知心姐姐”表示与事“鞠萍”的称谓，系事“北京”表示主事“通州”的归属关系。

（11）工具（instrument，简称I）：动词所表示的动作、行为所凭借的器具，有时可以用介词“用、以”等引导。例如：

妈妈用水果刀切黄瓜 | 王老师用显微镜看切片 | 刘大夫用中药治风湿 | 建筑师用计算机设计智能大楼

在上面的例子中，“水果刀”是“切”这种动作的工具，“计算机”是“设计”这种行为的工具。

（12）材料（material，简称MA）动词所表示的动作、行为所用的材料，有时可以用介词“用、以”等引导。例如：

妈妈用毛线结了一双手套 | 孙大爷用米泔水浇月季花 | 小海娃用红蚯蚓引诱鱼群 | 设计师用灯光装饰展厅

在上面的例子中，“毛线”是“结（手套）”这种行为所凭借的材料，“红蚯蚓”是“引诱（鱼群）”这种行为所用的材料。

（13）方式（manner，简称M）：动词表示的动作、行为所采取的方式、方法，有时可以用介词“用、以、经过、在（. . . . . 下）”等引导。例如：

余子真用低音唱了一首民歌 | 曼联队以点球取得了决赛胜利 | 他们经过三年的奋斗编纂了一部词典 | 电影节在社会各界的支持下顺利结束

在上面的例子中，“低音”是“唱（民歌）”这种行为的方式，“三年的奋斗”是“编纂（词典）”这种行为所用的方式。

（14）原因（reason，简写为RN）：动词所表示的动作、行为、事件等发生的原因，一般用介词“因、因为、由于、为、为了”等引导。例如：

鱼苗由于缺痒而死亡 | 雷曼银行因金融危机而倒闭 | 小明因考试失败而哭泣 | 小芳为找不到工作而叹气

在上面的例子中，“缺痒”是造成主事“鱼苗”“死亡”的原因，“找不到工作”是造成施事“小芳”“叹气”的原因。在表示原因的介词结构和动词之间经常用连词“而”来连接。

（15）目的（aim，简写为AI）：施事发出动词所表示的动作、行为、事件等的目的，一般用介词“为、为了”等引导。例如：

为了早日完工，工人们经常加班 | 为了出国旅行，她兑换了一些美元 | 为了加薪，小刚向厂长求情 | 为了养家糊口，大伙儿拼命地工作

在上面的例子中，“早日完工”是驱使施事“工人们”“加班”的目的，“养家糊口”是促使施事“大伙儿”“（拼命地）工作”的目的。

（16）时间（time，简写为T）：动词所表示的动作、行为、事件等发生的时间，有时可以用介词“在、于”等引导。例如：

工人们在节假日也不休息 | 我们明天去北京 | 在上大学期间他们俩就相识了 | 1996年春天孩子出生了 | 这件事发生在深夜

在上面的例子中，“节假日”是“（不）休息”这种行为发生的时间，“1996年春天”是“（孩子）出生”这种行为发生的时间。

（17）处所（location，简称L）：动词所表示的动作、行为、事件等发生的处所，有时可以用介词“在、于”等引导。例如：

同学们在教室里写毛笔字 | 我们火车站会合 | 在南操场上他们训练了一个上午 | 大礼堂里正在放映了一部美国电影 | 抢劫案发生在银行外面

在上面的例子中，“教室里”是“写（毛笔字）”这种行为发生的处所，“银行外面”是“（抢劫案）发生”的处所。

（18）源点（source，简称SO）：动词所表示的动作、行为开始的地点或时间，有时可以用介词“自、从、在、于”等引导。例如：

赛车从北京出发｜我们从窗口看风景｜我们从七月初放假｜这些学员来自沿海地区｜科举制度起源于隋朝｜监狱里跑了一个犯人

在上面的例子中，“北京”是“出发”这种行为发生的源点，“七月初”是“（开始）放假”的源点。

（19）终点（goal，简称GO）：动词所表示的动作、行为结束的地点、时间或状态，有时可以用介词“朝、向、往、到、至、在、于”等引导。例如：

骏马向北方飞驰｜我们朝窗外看星星｜在南墙上他们挂了一幅山水画｜黑板上老师又写了几行字｜我们将工作到八月底｜这些学员要分配到边远地区｜科举制度延续至晚清｜我们家里来了几个客人

在上面的例子中，“北方”是“飞驰”这种行为的终点，“八月底”是“工作”的终点。

（20）路径（path，简写为PA）：动词所表示的动作、行为、事件等中途经过的处所或时间，有时可以用介词“经过、通过、沿着、在、于”等引导。例如：

坦克从石拱桥上经过｜火车经过天津抵达北京｜士兵们沿着大马路巡逻｜我从窗户通过黑两座楼房的空档仰望天空｜我们将走四号线｜他们通过了榆树林

在上面的例子中，“石拱桥”是“经过”这种行为的路径，“四号线”是“走”这种行为的路径。

（21）量幅（extent，简写为EXT）：动词所表示的动作、行为、事件等所涉及的数量、频率、幅度、时间长度等相关事项。例如：

小明去了两趟｜鲁智深打了郑屠夫三拳｜一个西瓜卖八块钱｜渔船偏离了主航道几百米｜小王迟到了一刻钟｜我们结婚二十多年了

在上面的例子中，“两趟”是“去”这种行为的量幅，“一刻钟”是“迟到”这种行为的量幅。

（22）范围（range，简写为RA）：动词所表示的动作、行为、事件等所涉及的具体方面，或者是主事所依附的主体；它是主体、客体、环境、依凭和量幅之外的事物，有时可以用介词“在（……上/方面）、关于、对于”等引导。例如：

在立法方面，全国人大做了许多工作— 关于加快金融改革，刘建生教授提出了一个方案｜陈冬青在学业上取得了丰硕的成果｜大楼施工，我们顺利地完成了第一阶段的工程｜我的手表，发条断了｜这本书，爸爸只读了第一章

在上面的例子中，范围“立法方面”是施事“全国人大”“做了许多工作”这种行为的具体方面，范围“我的手表”是主事“发条”所依附的主体。

5 《动词信息词典》的句法格式系统及其例句

在本词典中，动词跟其配价成分构成的句法格式，主要有三大类：第一类，是主体论元作主语、客体论元作宾语的“主语+动词(+宾语)”类基础句式；第二类，是非主体论元（包括：客体论元、环境论元、依凭论元、关涉论元）作话题的“话题+主语+动词(+宾语)”类派生句式；第三类，是用介词“把”引导客体论元、或用介词“被”引导主体论元的“把”字句、或“被”字句等有标记的派生句式。例如：

（1）a. A + (用I +) __+ P：妈妈正在洗衣服。｜爸爸用大刀砍树枝。
b. A + (用MA +) __+ R：奶奶炒了两个菜。｜爷爷用柳条编了一个筐。
c. A + (在L +) __+ P/R：哥哥在路旁种了槐树。｜弟弟在墙上画了一幅画。
d. TH +__+ RE (+ EXT)：我姑姑是知识分子。｜轮船偏离了主航道20多米。

（2）a. P + A + (用I +) __：衣服妈妈已经洗了。｜那棵树爸爸用大刀砍了。
b. I + (A +) __+ P：这块香皂洗衣服。｜这把大刀爸爸砍树枝。
c. MA + (A +) __+ R：这些香椿炒鸡蛋。｜这些柳条我编一个箱子。
d. L + (A +) __+ P/R：路旁种了槐树。｜墙上弟弟画了一个太阳。

（3）a. A + (用I +) 把P +__：妈妈把衣服都洗了。｜爸爸用刀把那根树枝砍了。
b. P + 被A + (用I +) __：衣服被妈妈都洗了。｜那棵树被爸爸用大刀砍了。
c. A + 把MA + __+ R：奶奶把肉末炒了两个菜。｜爷爷把柳条编了一个筐。
d. MA +被A + __+ R：肉末被奶奶炒了两个菜。｜柳条被爷爷编了一个筐。

（1）是一般性的“主语+介词结构+动词(+宾语)”类基础句式。其中，施事、主事等主体论元作主语，受事、结果等客体论元作宾语，工具、材料、处所等环境或依凭论元通过介词

引导作状语。把这种状语放在圆括号中，表示没有它们也不影响句子的结构和意义的完整。（2）是特殊性的“话题+主语+动词(+宾语)”类派生句式。其中，受事、结果等客体论元前置到句首作话题，其余部分作说明（用以对话题作出评论）；施事、主事等主体论元作说明部分的主语，从而形成没有宾语的主谓谓语句。或者，工具、材料、处所等环境或依凭论元前置到句首作话题，其余部分作说明（用以对话题作出评论）；施事、主事等主体论元作说明部分的主语，受事、结果等客体论元作宾语，从而形成保留宾语的主谓谓语句。（3）是用介词“把”或“被”作标志的主谓句。在介词“把”的引导下，受事等客体论元可以前置到动词之前；在介词“被”的引导下，施事等主体论元可以降级为状语，同时受事等客体论元可以前置到到句首作主语。

上面的句式表示，不用主语、宾语等句法成分概念，也不用名词（性成分）、动词（性成分）等句法范畴概念，而是用施事、受事、与事、工具、材料、时间、处所、范围、量幅等语义角色概念；目的是要反映动词的几个惯常性的伴随成分在语句中的出现位置，以及它们之间的相互共现和选择限制关系，从而帮助人们在纷繁多变的语句形式上发现语句的意义骨架的有限性。如果我们能够把不同的上下文抽象成有限的几种语义格式（即由语义角色作为分布框架的句法构式），那么庶几可以在语言学习中以有限统御无限了。

6 《动词信息词典》中关于动词的语法功能信息

在本词典中，我们根据袁毓林等（2009）和袁毓林（2010b），设定动词的主要语法功能及其对于动词这一词类的隶属度：

（1）可以受否定副词“不”或“没有”修饰。符合得10分，不符合得0分。例如：

不走、不发展、没有前进、没有洗澡。

（2）可以后附或中间插入时体助词“着、了、过”，或者可以进入“...了没有”格式。符合得10分，不符合得0分。例如：

躺着、买了（票）、去过（美国），洗着澡、理了发、出过力、上过学，钉子锈了没有、樱花谢了没有、尸体腐烂了没有、东京地震了没有。

（3）可以带真宾语，或者通过“和、为、对、向、拿、于”等介词引导其必有论元。符合得20分，不符合得0分。例如：

看电影、踢足球、想心事、学习本领、研究太空，和朋友见面、为子孙造福、对孩子发火、向观众挥手、拿次货充数、昆曲起源于昆山。

（4）或者不能受程度副词“很”修饰，或者能同时受“很”修饰和带宾语。符合得10分，不符合得-10分。例如：

*很哭、*很做、*很改造、*很探索，很想家、很怕考试、很感谢你们、很担心她的身体。

（5）可以有“VV、V一V、V了V、V不V、V了没有”等重叠和正反重叠形式。其中，V代表动词。符合得10分，不符合得0分。例如：

坐坐、琢磨琢磨、瞧一瞧、试验一试验、说了说、调查了调查、吃不吃、考虑不考虑、馊了没有。

（6）可以作谓语或谓语核心（，因而一般可以受状语或补语修饰）。符合得10分，不符合得-10分。例如：

咱们走、钢丝断了、你们好好儿地玩、我们马上出发，修好、洗干净、整理出来、笑得前仰后合、跳得非常高。

（7）不能作状语直接修饰动词性成分。符合得10分，不符合得0分。例如：

*站吃～站着吃、*笑说～笑着说。

（8）可以跟在“怎么、怎样”之后，对动作的方式进行提问；或者可以跟在“这么、这样、那么、那样”之后，用以作出相应的回答。符合得10分，不符合得0分。例如：

怎么剪？～这么剪！、怎样开？～这样开！、怎么调查？～那么调查！、怎样发展？～那样发展！、双手是怎么颤的？～这么颤的！、植物是怎么发育的？～植物是这么发育的！。

（9）不能跟在“多”之后，对性质的程度进行提问；并且不能跟在“多么”之后，表示感叹。符合得10分，不符合得-10分。

Figure 1: 《动词信息词典》检索系统界面截图

8 关于纸质版《汉语动词造句词典》

我们把《动词信息词典》进行了简化处理，特别是删去了其中的"语法功能"这一板块，形成了纸质版本的《汉语动词造句词典》（将由商务印书馆出版），以方便广大读者携带和使用。

参考文献

袁毓林2010a. 汉语配价语法研究. 商务印书馆, 北京.

袁毓林2010b. 汉语词类的认知研究和模糊划分. 上海教育出版社, 上海.

袁毓林,马辉,周韧,曹宏. 2009. 汉语词类划分手册. 北京语言大学出版社, 北京.

朱德熙. 1982. 语法讲义. 商务印书馆, 北京.

Pinker Steven. 2007. *The Stuff of Thought: Language as a Window into Human Nature*. Viking Press, Penguin Groups, NY.

面向中文AMR标注体系的兼语语料库构建及识别研究

侯文惠[1], 曲维光[1,2], 魏庭新[2,3], 李斌[2], 顾彦慧[1], 周俊生[1]
(1.南京师范大学 计算机科学与技术学院，江苏省 南京市 210023;
2.南京师范大学 文学院，江苏省 南京市 210097;
3.南京师范大学 国际文化教育学院，江苏省 南京市 210097)

摘要

兼语结构是汉语中常见的一种动词结构，由述宾短语与主谓短语共享兼语，结构复杂，给句法分析造成困难，因此兼语语料库构建及识别工作对于语义解析及下游任务都具有重要意义。但现存兼语语料库较少，面向中文AMR标注体系的兼语语料库构建仍处于空白阶段。针对这一现状，本文总结了一套兼语语料库标注规范，并构建了一定数量面向中文AMR标注体系的兼语语料库。基于构建的语料库，采用基于字符的神经网络模型识别兼语结构，并对识别结果以及未来的改进方向进行分析总结。

关键词： 中文AMR ； 兼语结构 ； 识别

Research on the Construction and Recognition of Concurrent corpus for Chinese AMR Annotation System

HOU Wenhui[1], QU Weiguang[1,2], WEI Tingxin[2,3], LI Bin[2], GU Yanhui[1], ZHOU Junsheng[1]
(1.School of Computer Science and Technology, Nanjing Normal University, Nanjing , Jiangsu 210023, China; 2.School of Chinese Language and Literature, Nanjing Normal University, Nanjing , Jiangsu 210097, China; 3.International College for Chinese Studies, Nanjing Normal University, Nanjing , Jiangsu 210097, China)

Abstract

The concurrent structure is a common verb structure in Chinese. The predicate phrase and the subject-predicate phrase share the concurrent structure. The structure is complex and difficult to analyze. Therefore, the construction and recognition of the concurrent corpus is of great significance for semantic analysis and downstream tasks. However, there are few existing concurrent corpora, and the construction of concurrent corpora for the Chinese AMR labeling system is still in the blank stage. In response to this situation, this paper summarizes a set of concurrent corpus annotation specifications, and builds a number of concurrent corpora for Chinese AMR annotation systems. Based on the constructed corpus, this article uses an character-based neural network model to recognize the concurrent structure, and analyzes and summarizes the recognition results and future improvements.

Keywords: Chinese AMR , Concurrent structure , Recognition

收稿日期: **定稿日期**:
基金项目：国家自然科学基金“汉语抽象意义表示关键技术研究”(61772278)；江苏省高校哲学社会科学基金“面向机器学习的汉语复句语料建设研究”(2019JSA0220);国家社会科学基金“中文抽象语义库的构建及自动分析研究”（18BYY127）。

1 引言

兼语结构是由述宾短语与主谓短语套接而成的一种动词结构，述宾短语的宾语同时做主谓短语的主语，其结构通常表示为NP1+V1+NP2+V2(周鸣, 2018)。如“老师让大家补选一名劳动委员。”是一个典型的含有兼语结构的兼语句，该句中的“大家”既充当“让”的宾语又充当“补选”的主语。兼语结构与连动结构以及主谓短语做宾语结构相似，且存在共享省略成分，使得兼语句的识别与解析十分困难。据李斌(2017) 统计，兼语结构普遍存在于汉语语料中。因此，正确识别兼语结构，对句子的语义解析及其他下游任务具有重要意义。

语言学领域对兼语结构做了大量的研究，其工作集中在兼语句分类、语义研究、偏误分析等方面。然而在自然语言处理领域，针对兼语结构识别及相应的语料资源构建的研究较少。部分现有语料(周强, 2004; 郭丽娟, 2019) 中包含兼语结构的标注，但其不针对兼语构建，规模较小，规范不统一，无法用于兼语结构的识别及深入研究。现有的兼语结构识别工作依赖分词以及词性标注的效果，对未经人工校对的语料识别效果较差，对于低频兼语动词的识别能力有限。

抽象语义表示(abstract meaning representation, AMR)是一种新型的语义表示方法，它从语义角度出发，通过补充句子中的隐含或省略成分，更全面地描述句子的语义(曲维光 et al., 2017)，在语义解析任务中更具优势。AMR 在对语义标注时需要对兼语缺省的论元进行补充，因此，自动识别出兼语结构，将其转化为AMR图，可以辅助中文AMR语料的构建及解析，为语义解析及下游任务提供帮助。然而，中文AMR语料中兼语句较少，不足以用于训练，因此需要构建一定规模的兼语语料。

针对这一现状，我们构建了一个面向中文AMR标注体系的兼语语料库，并对语料库进行了统计分析。基于该语料，我们使用添加词典信息的字符神经网络模型识别兼语结构的边界，并对识别结果进行分析总结，讨论了未来可以改进的方向。

本文后续组织如下：第一节对以往的相关研究工作进行总结；第二节介绍面向中文AMR标注体系的兼语语料库构建工作，其中包括兼语结构界定、语料库构建规范以及统计分析；第三节主要介绍兼语结构识别的问题定义及模型；第四节介绍了兼语结构识别实验的结果及分析；最后一节总结全文并对未来的改进研究提出方向。

2 相关工作

语言学领域对于兼语句的理论及应用研究十分深入，也为自然语言处理领域的研究奠定了基础，但是兼语语料资源的匮乏限制了兼语结构识别的研究。

2.1 兼语语料库构建研究现状

兼语结构广泛存在于汉语中，语言学领域关于兼语结构的研究层出不穷。其研究工作主要集中在兼语句分类、语义研究、偏误分析等方面。胡裕树(1962/1979)将兼语句分为使令、促成类和有无类两类。邢福义、汪国胜(2010)则主张将兼语句分为使令式、爱恨式、有无式三类,并提到了连动兼语混用的情况。李婷玉(2017)从V1出发将兼语句式分为八个大类，并在语义分类和描述框架的基础上对八个大类进行细分。马德全、王利民(2010)对V1的二价动词和三价动词的应用进行了考察。司玉英(2010)对双宾兼语句各成分之间的语义关系进行了分析。

然而，针对兼语语料资源构建的工作较少，只有少数综合语料中包含兼语结构标注。周强(2004) 构建TCT(Tsinghua Chinese Treebank)的句法标记集采用功能分类的方法对汉语短语进行描述。其中，兼语结构是一类特殊的动词短语，使用“vp-JY”作为标识，兼语动词使用“vJY”标注。TCT对兼语结构这类特殊的动词短语有明确的边界标注。但该语料库对于兼语结构以及主谓短语做宾语结构的界定模糊，且未对兼语结构中的兼语以及V2进行标注。中文依存句法树库中也包含对兼语的标注，HIT-CDT(Harbin Institute of Technology Chinese Dependency Treebank)中使用“DBL”这个依存关系类型标注V1以及兼语中心词的依存关系，SU-CDT(SUDA Chinese Dependency Treebank)(郭丽娟, 2019) 在此标注系统的基础上增加一个“pred”依存关系类型，该关系类型用来标注兼语指向V2的关系，使得兼语与V2的语义关系更加紧密。李斌(2017)等构建的中文AMR语料将一个句子抽象为一个AMR 图，通过补充句子中的隐含或省略成分，完善句子的语义信息。兼语结构是典型的包含省略成分的结构，AMR会将共享的兼语进行补充。但以上的语料并不针对兼语构建，规模较小，且各语料对于兼语的定义不统一，无法直接用于兼语结构的识别工作。

2.2 兼语结构识别研究现状

现有的兼语结构识别研究主要分为两类，一类是基于规则的识别方法，另一类是基于机器学习的识别方法。傅成宏(2007) 统计分析了1998年《人民日报》1月份语料，通过建立兼语动词词表识别V1，并在此基础上使用规则方法识别兼语边界，进而确定兼语结构的存在。但兼语动词词表的建立需要依赖语料，难以建立适合所有语料的兼语动词词表，无法识别新产生的兼语动词。且该方法只能识别符合语法规则的简单兼语结构，无法处理不符合语法规则的句子以及包含连动、复句等其他结构的复杂兼语结构，也无法对完整的兼语结构进行识别，难以达到应用层面。陈静(2012) 等将兼语结构边界识别问题转化为序列化标注问题，使用CRF模型识别兼语结构的边界。但是，该工作基于人工校对的语料进行，依赖分词以及词性标注的效果，对于大量未加工的语料识别效果较差。兼语中存在大量低频兼语动词，且其“使令”含义不强，CRF模型的识别效果有限。近年来，神经网络模型的出现有效提高了序列化标注任务的效果。词性标注以及命名实体识别通常被建模为序列化标注任务解决，神经网络因其具有更好的泛化性和不依赖手工选择特征等特点而被广泛应用于词性标注以及命名实体识别。Pinheiro和Collobert(2014)等首次将CNN模型与CRF模型结合，并用于命名实体识别任务。Chiu和Nichols(2016)结合CNN模型与LSTM模型，然后与CRF模型拼接，进一步提高了命名实体识别的效果。近年来，基于字符的神经网络模型被广泛应用于命名实体识别任务，Zhang(2018)等提出的Lattice LSTM模型在中文命名实体识别中取得了较好的结果。目前为止，还没有使用神经网络模型识别兼语结构的研究。

相关语料的缺乏限制了兼语结构边界识别任务的解决和提升。为了对兼语结构进行语义解析，需要构建更为细致的语料，因此本文构建了一个面向中文AMR标注体系的兼语语料库，并使用添加词典信息的字符神经网络模型识别兼语结构，缓解了分词系统造成的错误传播，有效提高了兼语结构的识别效果。

3 兼语语料库构建

本文首先对兼语结构进行界定，筛选出兼语句，然后根据标注规范构建语料库，最后对构建的语料库进行了统计分析。

3.1 兼语结构的界定

兼语结构是一种套接的动词结构，通常将结构表示为NP1+V1+NP2+V2(周鸣, 2018)，其中V1和V2的关系为递系式(张志公, 1957)，NP2为兼语，V1一般是具有“使令”含义的动词，V1和V2共享NP2，NP2分别做V1和V2的受事宾语和施事主语。AMR在标注兼语结构时，将NP2标注为V1的arg1，V2的arg0。具体示例如图1所示。

但是汉语句式复杂多变，仅凭结构难以识别，其中连动结构以及主谓短语做宾语结构与兼语结构尤为相似，需要结合结构和语义进行判定。具体界定过程一般分为两步。(1)筛选具有NP1+V1+NP2+V2结构，且NP2充当V1宾语，V2主语的句子。(2)判断V1宾语涉及的范围是NP2还是整个主谓结构构成的短语或从句，如果只涉及NP2则判定为兼语句，否则，判定为非兼语句。

TCT中对兼语结构和主谓短语做宾语的界定模糊。“建议纪委介入调查”是一个典型的主谓短语做宾语句，“建议”的内容是“纪委介入调查”，涉及的范围是其后的整个从句，但TCT将其标注为兼语结构。本文在构建语料库时，综合考虑了以上两个界定步骤，有效避免了两类结构界定模糊的问题。

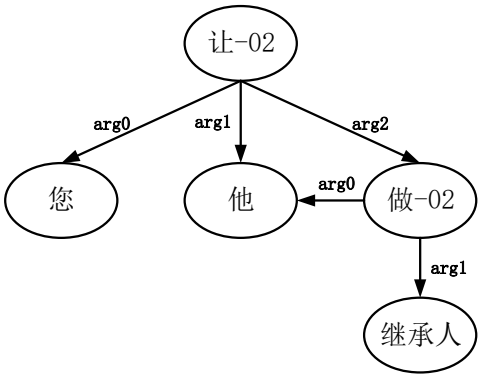
3.2 兼语语料库构建规范

本文构建的兼语语料库主要对兼语结构的边界、兼语的中心词以及V1、V2 进行标注。

3.2.1 兼语结构的前边界

本文语料库将兼语结构的前边界标注在V1前，并用“【【”标注。如果兼语结构的V1存在于连动结构中，则将兼语结构的前边界标注到连动结构的第一个动词前。

例1：**他 【【号_V1 召_V1 和 动 员 全 体 指_JY 战_JY 员_JY 节_V2 衣_V2 缩_V2 食_V2】】 。**



句子：您让他做继承人
x2/让
:arg0 x1/您
:arg1 x3/他
:arg2 x4/做
:arg0 x3/他
:arg1 x5/继承人

Figure 1: 兼语结构AMR图

3.2.2 兼语结构的后边界

本文语料库将兼语结构的后边界标注在V2所在的动词短语后，用“】】”标注，如果兼语结构的V2存在于连动结构中，则将兼语结构的后边界标注到连动结构中最后一个动词所在的动词短语之后。

例2：**它 能 【【帮_V1 助_V1 人_JY 类_JY 开_V2 拓_V2 未 知 的 领 域 和 获 得 新 的 知 识】】 。**

说明：例2中的“开拓”和“获得”的主语都是“人类”，且可以将其拆分成两个兼语结构，一个是“帮助人类开拓未知的领域”，一个是“帮助人类获得新的知识”。后边界标注到连动结构的最后一个动词所在的动词短语之后。

3.2.3 V1的标注

本文语料库使用“_V1”标注V1，如果兼语结构的V1存在于连动结构中，则只标注连动结构中的第一个兼语动词。具体情况如例1所示。

3.2.4 兼语标注

本文语料库使用“_JY”标注兼语。汉语中的兼语通常为一个名词、代词、名词短语或一个主谓宾结构，本文在标注兼语时只标注其中心词。针对各类复杂情况，本文对兼语标注规范进行以下细化规定。

（1）如果兼语为名词短语，则标注名词短语的中心词。

例3：**奏 鸣 曲 【【让_V1 专 修 音 乐 的 妹_JY 妹_JY 大_V2 吃_V2 一_V2 惊_V2】】 。**

说明：该例句中“专修音乐的妹妹”构成的名词短语充当兼语，“妹妹”为该名词短语的中心词，故只对“妹妹”进行标注。

（2）如果兼语是由多个名词或名词短语并列组成，则对其中的每一个名词或名词短语的中心词进行标注。

例4：**能 够 【【让_V1 灾 区 的 孩_JY 子_JY 、 学_JY 生_JY 得_V2 到_V2 相 应 的 关 怀】】 就 够 了 。**

说明：该例句中“灾区的孩子、学生”两个并列的名词短语充当兼语，“孩子”和“学生”分别为两个名词短语的中心词，故对这两个词进行标注。

（3）如果兼语由一个完整的主谓宾结构构成，则标注该结构的中心谓词。

例5：**【【使_V1 高 速 度 大 容 量 异 种 机 传_JY 输_JY 信 息 成_V2 为_V2 可 能】】 。**

说明：该例句中“高速度大容量异种机传输信息”为一个完整的主谓宾结构，其中的谓词“传输”为该结构的中心词，故对其进行标注。

3.2.5 V2的标注

本文语料库使用“_V2”标注V2，针对兼语句中包含连动、复句、以及其他修饰成分等复杂情况，对V2 的标注规范进行以下细化规定。

（1）如果兼语结构主谓词组的谓词存在于连动结构中，则将V2标注为连动结构中的第一个动词。该类型是包含连动的复杂兼语结构，具体例子如下。

例6：**把 读 书 当 成 【【使_V1 人_JY 信_V2 教_V2 修 行】】 的 一 种 手 段 。**

说明：例6中的“信教”和“修行”构成连动结构，我们将V2标注为连动结构的第一个动词“信教”。

（2）如果句中包含“去吃饭”、“来做客”这类连动结构，AMR会将“去”和“来”这类无实际含义的词省略，本文在标注V2时标注第一个动词。

例7：**他 们 【【邀_V1 请_V1 全 国 1 8 家 甲 级 城 市 规 划 设 计 院 的 专_JY 家_JY 来_V2 考 察 论 证】】 。**

（3）如果主谓词组为情态动词加动词的结构，则将V2 标注为情态动词。

例8：**要 重 视 理 论 队 的 建 设 ， 【【使_V1 确 有 成 就 的 青 年 理 论 人_JY 才_JY 能_V2 脱 颖 而 出】】 。**

（4）如果存在一个动词作为另一动词的“方式”的句子，则将V2标注为兼语之后的第一个动词。

例9：**【【让_V1 乡_JY 亲_JY 们_JY 集_V2 中_V2 到 一 个 碾 子 上 碾 米】】 。**

说明：AMR标注体系会将“碾”作为“乡亲们”的谓语，而将“集中到一个碾子上”作为“碾米”的方式，为了与前面的标注标准一致，故将V2 标注为兼语后的第一个动词“集中”。

（5）如果兼语结构中存在主谓词组后有补语的情况，则将V2标注为兼语后的第一个动词。

例10：**我 们 也 尽 可 能 【【让_V1 她_JY 过_V2 得 充 实 如 意】】 。**

（6）如果兼语结构中含有复句，对于并列以及递进等没有主次关系的复句，将V2标注为复句第一部分的谓词，对于其他带有主次关系的复句，将V2标注为主要子句中的谓词。

例11：**老 师 【【让_V1 她_JY 一 边 听_V2 语 音 一 边 记 笔 记】】 。**

例12：**干 吗 【【让_V1 人_JY 家_JY 一 进 门 就 赶_V2 上_V2 一 顿 熊】】 呢**？

3.3 兼语语料库的统计分析

本文选取了来自文学、新闻、微博等不同领域的67419 个句子作为语料构建的原始语料，从中筛选得到了4760 个兼语句以及5248个兼语结构，并按照本文设计的兼语结构标注规范完成了兼语语料库的构建。我们对兼语结构中V1出现的频率进行统计，其中出现频率最高的六个词如图2所示。根据图2可以看出兼语结构中的兼语动词多集中在“让”、“使”、“令”、“请”、“叫”、“要求”等词，这六个词构成的兼语结构占所有兼语结构的70.8%。

本文对低频兼语动词也进行了统计，其中出现频次低于5次的兼语动词数量如表1所示，根据表1可以发现兼语语料库中包含大量低频兼语动词，其中出现频率为1次的有128个，出现频率为2次的有51个。低频动词多为高频动词的近义词，使用规则以及统计的方法难以识别此类动词，低频兼语动词的大量存在使得兼语结构识别工作十分困难，因此有效处理低频兼语动词对兼语结构的识别具有重要意义。

4 兼语结构识别研究

基于构建的兼语语料库，我们使用神经网络模型自动识别兼语结构的边界，辅助中文AMR语料的构建及解析。由于兼语结构的语义关系复杂，句式变化丰富，因此兼语结构的识别任务具有一定的挑战性。

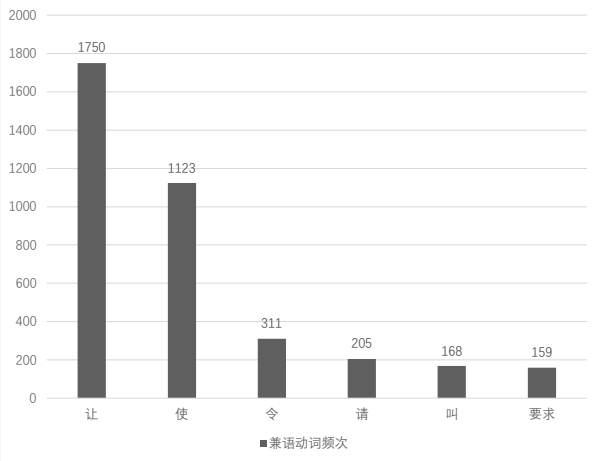


Figure 2: 兼语动词频次图

频次	数量	示例
1	128	胁迫、恳请、指派、催促
2	51	吁请、责令、诚邀
3	18	打发、煽动
4	8	任命、扶持
5	10	督促、选派、提请

Table 1: 低频兼语动词表

4.1 任务定义以及数据划分

我们将兼语结构的识别任务建模为序列化标注任务。给定输入的句子序列$X = \{x_1, x_2, ..., x_n\}$，模型需要预测出对应输入句子序列的标签序列$Y = \{y_1, y_2, ..., y_n\}$,其中$y_i \in \{B, M, E, O\}$。B标签对应兼语结构的起始字，E 标签对应兼语结构的结尾字，M标签对应兼语结构除以上成分的其他字，O标签对应句子的非兼语结构，句子对应标签序列示例如表2所示。我们将标注好的语料导出为序列化标注格式的文件，并随机打乱顺序，选取其中的10%作为测试集，然后从剩余的语料中选取90% 作为训练集，10% 作为开发集。

文本	林	老	师	让	大	家	补	选	一	名	劳	动	委	员	。
标签	O	O	O	B	M	M	M	M	M	M	M	M	M	E	O

Table 2: 兼语句标注示例表

4.2 模型

兼语结构中通常包含两个动词，因此使用规则或者机器学习的方法对兼语结构进行识别时，通常将词性作为重要的特征进行统计分析。然而现存的语料大部分没有分词以及词性标注，使用自动分词以及词性标注工具处理语料容易造成错误传播。单独使用字符信息对兼语结构进行识别容易丢失词语本身携带的信息，因此我们对字向量及其对应的词典信息进行拼接，获得句子完整的向量表示。将该向量传入表示层，获得包含上下文信息的句子表示。常用的表示层模型有BiLSTM、CNN、Transformer等，本文选用BiLSTM模型作为表示层获取句子的上下文信息。兼语结构的标签具有很强的依赖性，比如M标签必须在B之后，而不能在O之后，如果对每个标签进行独立决策，则无法考虑其间的依赖性。因此我们在BiLSTM模型之后拼接了CRF(李航, 2012) 模型，实现对含有约束关系的序列标签解码。最终构成的LexcionAugemented-BiLSTM-CRF(Peng et al., 2020)模型(LA-BiLSTM-CRF) 可以完成纯文本的兼语结构边界识别任务。

4.2.1 LA-BiLSTM-CRF模型

使用添加词典信息的字向量表示句子，既有效运用了文本中包含的词语信息，又避免了分词工具带来的错误传播。本文模型的输入为不包含任何分词以及词性信息的文本内容，然后使用公式(1)(2)(3)(4)获取每个字对应的向量表示。

$$x_i = [x_i^c; e^s(B, M, E, S)] \tag{1}$$

$$e^s(B, M, E, S) = [v^s(B) \bigoplus v^s(M) \bigoplus v^s(E) \bigoplus v^s(S)] \tag{2}$$

$$v^s(S) = \frac{1}{Z} \sum_{w \in S} Z(w + c) e^w(w) \tag{3}$$

$$Z = \sum_{w \in B \bigcup M \bigcup E \bigcup S} Z(w) + c \tag{4}$$

其中x_i表示当前句子中第i个字的向量表示，该向量由字向量与词语向量拼接构成，字向量通过查找字表获得对应的向量表示。$e^s(B, M, E, S)$表示包含当前字的所有词向量信息，$v^s(B)$表示以当前字为开始的所有词的向量表示，$v^s(M)$表示当前字在词的中间组成部分的所有词向量表示，$v^s(E)$表示以当前字为结尾的所有词的向量表示，$v^s(S)$表示当前字独立构成的词的向量表示。S表示某一种词集合，Z表示词语出现的频率，$e^w(w)$表示w的词向量。将词语的频率作为该词的权重，对集合内的所有词向量进行加权求和得到词集合的向量表示。

根据上述公式获得句子的向量表示$(x_1, x_2, \ldots, x_n)$，其中，n表示句子包含词的数量。将句子的向量表示传入BiLSTM模型(Yang et al., 2018)，获取包含上下文信息的句子表示$(h_1, h_2, \ldots, h_n)$。然后将其传入CRF 模型中，对于一个预测序列$y = (y_1, y_2, \ldots, y_n)$，将该序列的得分定义为公式(5)所示。

$$s(X, y) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} P_{i, y_i} \tag{5}$$

其中，P定义为BiLSTM网络输出的分数矩阵，是一个规模为n×k的矩阵，其中k是标签的数量，P_{ij}对应的是在一个句子中第i个词语对应第j个标签的得分。A是一个转移得分矩阵，A_{ij}表示从第i个标签转移到第j个标签的得分。y_0和y_n是句子标签的开始和结束标志，我们将开始和结束标志也加入到候选标签集合，所以A是一个$(k + 2) \times (k + 2)$维的矩阵。

通过softmax函数对所有可能的标记序列进行概率计算，使用公式(6) 计算序列y的概率。在训练过程中，使用交叉熵损失函数来更正标签序列的预测，具体如公式(7)所示。

$$P(y|X) = \frac{e^{s(X,y)}}{\sum_{\tilde{y} \in Y_X} e^{s(X,\tilde{y})}} \tag{6}$$

$$\log(P(y|X)) = s(X, y) - \log(\sum_{\tilde{y} \in Y_X} e^{s(X,\tilde{y})}) \tag{7}$$

其中，Y_X代表对应于句子X的所有可能的标签序列集合。使用Viterbi 算法解码，得到最优输出序列(Graves and Schmidhuber, 2005)。解码过程中，预测得到的输出序列的最大得分由公式(8)计算得到。整体的模型结构图如图3所示。

$$y^* = \arg\max_{\tilde{y} \in Y_X} s(X, \tilde{y}) \tag{8}$$

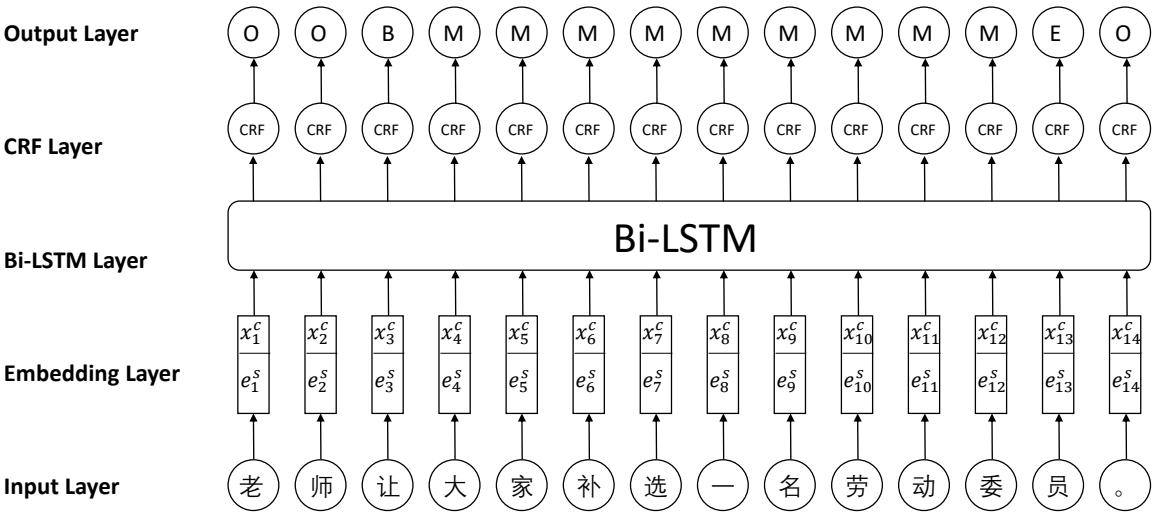


Figure 3: LA-BiLSTM-CRF模型结构图

5 兼语结构识别研究

实验中使用pytorch编写LA-BiLSTM-CRF模型。实验使用的语料是本文构建的面向中文AMR标注体系的兼语结构标注语料。

5.1 实验参数设置

LA-BiLSTM-CRF模型的参数设置如表3所示。本文的词向量使用预训练的CTB6.0(Xue et al., 2005)50 维词向量，字向量以及使用word2vec(Mikolov et al., 2013)训练的Giga-Word 50维字向量，通过BiLSTM模型获得隐藏层为300维的含有上下文信息的句子向量表示。然后将此向量表示传入CRF模型，获得对应输入序列的输出标签预测序列。使用Adam(Kingma et al., 2014)优化函数训练模型，学习率为0.0015，学习衰减率为0.05。对所有语料循环训练30 次，保留在开发集上预测结果最佳的模型，使用该模型对测试集数据进行预测。

超参数名称	参数值
epoch	30
Learning rate	0.0015
Learning rate decay	0.005
Word embedding size	50
Char embedding size	50
Hidden size	300

Table 3: 超参数设置表

5.2 实验结果及分析

由于没有使用神经网络模型识别兼语结构边界的相关研究工作，因此本文对神经网络模型的构建进行了探索。我们分别使用了CNN、Transformer以及BiLSTM(Lample et al., 2016)这三个基础模型作为表示层提取句子特征，其识别结果如表4 所示。

根据表4的实验结果可以发现，BiLSTM模型在兼语结构边界识别任务中表现最好，其F1值达到了86.06%。CNN模型的识别效果最差，其F1值为67.54%，CNN模型对于文本局部特征的捕捉能力较强，但是难以捕捉长兼语结构的特征，因此其识别效果较差。Transformer模型采用注意力机制提取文本特征，解决了文本的长距离依赖问题，其识别效果优于CNN模型。但Transformer模型难以捕捉句子中字词的位置方向信息，对于兼语结构中包含的连动以及宾

模型	P	R	F
LA-CNN-CRF	65.04	70.25	67.54
LA-Transformer-CRF	73.48	66.14	69.62
LA-BiLSTM-CRF	**86.25**	**85.91**	**86.06**

Table 4: 神经网络模型对比实验结果表

语从句这种与位置方向有关的结构学习能力较差。该模型通常只捕捉兼语结构中的一个V2及该V2对应的宾语，对于包含连动以及宾语从句的兼语结构的后边界识别效果较差。BiLSTM 模型既可以捕捉句子中较长的上下文信息，又不会丢失句子中字词的位置方向信息，其对于长兼语结构以及包含连动或宾语从句的兼语结构识别效果优于以上两个模型，在兼语结构边界识别任务中表现突出，其精确率、召回率以及F1值分别为86.25%、85.91% 和86.06%。实验结果证明，BiLSTM模型更适合兼语结构边界识别任务。

为了证明基于字符的神经网络模型以及词典信息的有效性，我们做了相关的消融实验，其实验结果如表5所示，其中BiLSTM-CRF_W是基于词和词性信息的神经网络模型，BiLSTM-CRF_C 是基于字符的神经网络模型，LA-BiLSTM-CRF是本文的添加词典信息的字符神经网络模型。

模型	P	R	F
BiLSTM-CRF_W	71.72	75.87	73.73
BiLSTM-CRF_C	85.52	84.34	84.93
LA-BiLSTM-CRF	**86.25**	**85.91**	**86.06**

Table 5: 消融实验结果表

根据表5可以发现BiLSTM-CRF_C模型的精确率、召回率以及F1值比BiLSTM-CRF_W模型分别高13.80%、8.47%和11.20%，这证明基于字符的神经网络模型缓解了分词以及词性标注的错误传播问题，有效提高了兼语结构边界识别任务的效果。但该模型丢失了句子中包含的词语信息，本文在此模型的基础上添加了词典信息，使用LA-BiLSTM-CRF 模型识别兼语结构边界，使得识别结果的精确率、召回率以及F1值分别提高了0.73%、1.57% 和1.13%，实验结果证明添加词典信息可以有效提高基于字符的神经网络模型对兼语结构边界识别的效果。

目前为止，兼语结构边界识别的研究工作较少，只有陈静(2012)采用基于特征模板的条件随机场模型对兼语结构边界进行了识别研究，因此我们使用陈静(2012)的模型以及特征模板对本文构建的语料进行识别，并将其结果与本文模型的结果进行对比，实验结果如表6所示。我们还对两个模型的所有标签识别效果进行了研究，具体结果如表7所示。

模型	P	R	F
CRF	**87.12**	82.24	84.61
LA-LSTM-CRF	86.25	**85.91**	**86.06**

Table 6: 对比实验结果表

根据表6的实验结果可以发现，LA-BiLSTM-CRF模型识别兼语结构边界的F1值比CRF模型提高了1.45%。CRF模型识别的精确率为87.12%，略高于LA-BiLSTM-CRF模型，而LA-BiLSTM-CRF模型识别的召回率为85.91%，比CRF模型的召回率高3.67%。就表7的各标签识别效果而言，两个模型对兼语结构前边界的识别效果最好，其F1值分别为93.55%和94.51%，后边界识别效果最差，其F1值分别为85.00% 和86.67%。CRF模型对前边界与后边界识别的精确率为96.32%和87.53%，分别高于本文模型2.00%和1.03%，但本文模型对前边界与后边界识别的召回率较高，比CRF模型分别高3.77%和4.21%，且F1值比CRF模型分别高0.96%和1.67%。实

标签	CRF			LA-BiLSTM-CRF		
	P	**R**	**F**	**P**	**R**	**F**
B	96.32	90.93	93.55	94.32	94.70	**94.51**
M	88.58	91.43	89.98	94.03	89.15	**91.52**
E	87.53	82.63	85.00	86.50	86.84	**86.67**

Table 7: 各标签识别结果表

验结果证明，CRF模型基于特征模板进行训练，识别结果较为精确，但其难以识别包含低频兼语动词以及兼语动词存在分词错误的兼语结构。LA-BiLSTM-CRF使用向量对句子进行表示，有效提高了兼语结构前边界识别的召回率以及F1值。兼语结构本身较为复杂，其内部常包含许多修饰成分，且前边界识别的错误直接影响后边界的识别效果，因此，后边界的识别效果较差。总体而言，本文模型对三个标签的识别效果都有不同程度的提升，并且有效提高了兼语结构边界识别任务的效果。

此外，我们还对模型的识别结果进行了错误分析。兼语结构前边界的识别错误主要发生在低频兼语动词中，大部分为语料中只出现一次的兼语动词。尽管使用字向量表示句子，缓解了低频兼语动词难以识别的问题，但是对于本身出现频率较低、“使令”义不强的兼语动词识别效果较差。比如“留她吃饭”中的兼语动词“留”在语料中只出现过一次，且其“使令”义较弱，因此模型未识别出该兼语结构。此外，模型会将部分高频兼语动词构成的非兼语结构错判为兼语结构，比如“使了个瞒天过海之计”中的“使”是高频兼语动词，模型将其误判为兼语结构。兼语动词的识别错误会导致错误传播，直接影响兼语后边界的识别效果。此外，兼语结构的后边界识别错误主要出现在包含定中结构或做定语的兼语结构。比如“让儿子买本养花的书参照执行”是包含定中结构的兼语结构，模型将兼语结构的后边界识别为“养花”。而在“战争是迫使敌人服从我们意志的一种暴力行为。”这一句子中，兼语结构作为定语修饰“暴力行为”，模型将该兼语结构的后边界识别为“行为”。由此可见，模型对于这两类兼语结构的后边界判别能力较差。

6 总结与展望

本文根据中文AMR标注体系的特点，制定了一套面向中文AMR标注体系的兼语结构标注规范，并利用此规范对收集的语料进行了兼语结构标注，得到4760句兼语句，5248个兼语结构，缓解了面向中文AMR标注体系的兼语语料库缺乏的问题。基于该兼语语料库，本文使用添加词典信息的字符神经网络模型识别兼语结构，避免了分词以及词性标注系统造成的错误传播，有效提高了兼语结构的识别效果，以期对今后的中文AMR语料构建及解析任务提供帮助，从而为语义解析及其下游任务奠定基础。

基于字符的神经网络模型缓解了低频兼语动词难以识别的问题，但低频兼语动词的存在仍然影响兼语结构前边界的识别效果。且模型对于包含定中结构或做定语的兼语结构识别效果较差。因此解决低频兼语动词的识别以及定中结构的边界判定是今后提高兼语结构识别的重点。此外，我们仍需要不断标注新的语料，使得模型学习到更多复杂的句子形式，提高模型处理复杂句子的能力。

参考文献

陈静, 王东波, 谢靖, 郑建明. 2012. 基于条件随机场的兼语结构自动识别. 情报科学,30(03): 439-443.

傅成宏. 2007. 现代汉语兼语结构的自动识别. 南京师范大学.

郭丽娟. 2019. 汉语依存句法分析树库构建与应用研究. 苏州大学.

胡裕树. 1962/1979. 现代汉语. 上海：上海教育出版社.

李斌, 闻媛, 宋丽, 卜丽君, 曲维光, 薛念文. 2017. 融合概念对齐信息的中文AMR语料库的构建. 中文信息学报,31(6): 93–102.

李航. 2012. 统计学习方法. 北京：清华大学出版社,pages 191-210.

李婷玉, 王亚, 曹聪. 2017. 兼语语义类的分类研究. 计算机应用研究,34(01):15-20.

马德全, 王利民. 2010. 兼语句的语义分析. 内蒙古民族大学学报（社会科学版）,36(04): 30-32.

曲维光, 周俊生, 吴晓东, 戴茹冰, 顾敏, 顾彦慧. 2017. 自然语言句子抽象语义表示AMR研究综述. 数据采集与处理,32(1): 26–36.

司玉英. 2010. 双宾兼语句的语法、语义和语用特征. 内蒙古大学学报（哲学社会科学版）,42(01): 148-152.

邢福义, 汪国胜. 2010. 现代汉语. 北京：高等教育出版社.

张志公. 1957. 修辞概要. 上海：上海新知识出版社.

周鸣. 2018(24). 浅谈兼语式定义问题. 汉字文化, pages 90–92.

周强. 2004(04). 汉语句法树库标注体系. 中文信息学报, pages 1-8.

Alex Graves and Jurgen Schmidhuber.2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*,18(5):602-610.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Design Challenges and Misconceptions in Neural Sequence Labeling.2016. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. San Diego, California : Association for Computational Linguistics*,pages 260-270.

Jason P. C. Chiu and Eric Nichols.2016(4). Named Entity Recognition with Bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*,pages 357-370.

Jie Yang, Shuailong Liang and Yue Zhan.2018. Design Challenges and Misconceptions in Neural Sequence Labeling. *Proceedings of the 27th International Conference on Computational Linguistics. Santa Fe, New Mexico, USA: Association for Computational Linguistics*,pages 3879-3889.

Kingma, Diederik P and Ba, Jimmy.2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980.*

Minlong Peng, Ruotian Ma, Qi Zhang, and Xuanjing Huang.2020. Simplify the Usage of Lexicon in Chinese NER. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistic*, pages 5951–5960.

Naiwen Xue, Fei Xia, Fu-Dong Chiou, and Marta Palmer.2005. The Penn Chinese treebank: Phrase structure annotation of a large corpus. *Natural Language Engineering*,11(02): 207-238.

Pinherio, Ronan Collobert Pedro HO, and H. Pedro.2014. Recurrent Convolutional Neural Networks for Scene Parsing. *International Conference of Machine Learning.*

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean.2013. Distributed Representations of Words and Phrases and their Compositionality. *Proceedings of Advances in Neural Information Processing Systems 26*,pages 3111–3119.

Yue Zhang and Jie Yang.2018. Chinese NER Using Lattice LSTM. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Melbourne, Australia: Association for Computational Linguistics*,pages 1554-1564.

面向人工智能伦理计算的中文道德词典构建方法研究

王弘睿
北京语言大学
whongrui18@163.com

刘畅
北京语言大学
liuchang2014@gmail.com

于东*
北京语言大学
yudong_blcu@126.com

摘要

道德词典资源的建设是人工智能伦理计算的一个研究重点。由于道德行为复杂多样，现有的英文道德词典分类体系并不完善，而中文方面目前尚未有相关的词典资源，理论体系和构建方法仍待探究。针对以上问题，该文提出了面向人工智能伦理计算的中文道德词典构建任务，设计了四类标签和四种类型，得到包含25,012个词的中文道德词典资源。实验结果表明，该词典资源不仅能够使机器学会道德知识，判断词的道德标签和类型，而且能够为句子级别的道德文本分析提供数据支持。

关键词： 伦理计算 ； 道德判断 ； 机器学习

Construction of a Chinese Moral Dictionary for Artificial Intelligence Ethical Computing

Hongrui Wang
Beijing Language
& Culture University
whongrui18@163.com

Chang Liu
Beijing Language
& Culture University
liuchang2014@gmail.com

Dong Yu
Beijing Language
& Culture University
yudong_blcu@126.com

Abstract

The construction of the moral dictionary is based on artificial intelligence ethical computing. Moral behavior is complex and varied. The existing moral dictionary classification system for English language is still under development. Meanwhile, there are currently no relevant dictionary resources in Chinese. The theoretical system and construction method are still to be explored. In this paper, the task of constructing a Chinese moral dictionary for artificial intelligence ethics calculation is proposed. Four polar labels and four types of labels are designed to obtain a Chinese moral dictionary resource containing 25,012 words. Experimental results show that the dictionary resource can not only enable the machine to learn moral knowledge and judge the moral polarity and type of words, but also provide data support for the analysis of moral text at the sentence level.

Keywords: ethical computing , moral judgment , machine learning

*为通讯作者
基金项目： 国家社会科学基金(17ZDA305);教育部人文社会科学研究青年基金项目(19YJCZH230);北京语言大学中青年学术骨干支持计划

1 引言

道德判断是人工智能伦理计算的一个重要问题。随着人工智能技术的快速发展，机器接管了越来越多人类的工作任务，社会对人工智能决策的道德性的担忧也与日俱增：人工智能能否理解我们的道德观念，又能否学会人类的道德判断？正如Picard(Picard, 1997)所说，“一台机器的自由度越大，它就越需要道德标准”。将重大决策的控制权交给机器之前，机器首先需要具有符合人类道德标准的判断能力。

使用词汇来识别复杂的道德概念，从而让机器学会道德标准甚至具备道德判断的能力，被认为是一种可靠的方法。Graham等人(Graham et al., 2009; Hofmann et al., 2014; Feinberg and Willer, 2013; Clifford and Jerit, 2013)研究表明，词汇是可以帮助机器进行道德判断的一个相当可靠的标识符。在过去的十年里，已经有许多研究运用词汇学的方法来分析文本数据的道德基础(Kaur and Sasahara, 2016; Hoover et al., 2019; Sagi and Dehghani, 2014)。机器可以了解到“恶意谋杀”是不道德的，“诚实守信”则是一种好的行为。道德词典成为实现人工智能伦理计算的重要数据资源。

目前发布的英文道德词典主要存在两个问题，一是目前的道德词典只考虑了与道德行为有关的动词，词性构成和词的类型比较单一；二是目前的道德词典普遍规模较小，难以较为全面地覆盖道德行为。因此，现有的英文道德词典体系尚不完善。目前尚未有公开发布的中文道德词典，其理论体系及构建方法值得分析研究。

针对第一个问题，本文认为，除了动词，名词、形容词以及一些成语也与行为的道德倾向息息相关，如活雷锋、无私、公正廉洁等。除此之外，汉语中有不少体现道德倾向的被动表达，比如被害、弃婴等。因此，本文将动词、名词、形容词、成语和汉语中的被动表达都纳入到道德词典理论体系中，并根据所标注词在事件行为中所处的位置，分为事件行为、事件状态、事件属性、事件要素四种类型，其中事件要素再分为对象、媒介和地点三类。本文研究丰富了词典的词性构成和词的类型，覆盖了更多的道德行为。

针对第二个问题，本文根据《现代汉语常用词表》(张清源, 1992)整理出基础道德词表，并通过词向量对基础词表进行扩展。道德词典由2,777个词扩展到29,907个词，经过标注核查后最终包含25,012个有效词，减少了人工标注的成本，高效地扩展了道德词典的规模。

最后，为了检验词典的有效性，本文从词的标签及类型识别和判断句子道德倾向两个维度进行了实验。实验结果表明，道德词典资源能够较为准确地判断词的标签和类型，也能够较好地判断句子的道德倾向。

综上，针对中文道德词典资源缺乏的问题，本文提出了面向人工智能伦理计算的中文道德词典构建任务。本研究的主要贡献包括以下三个方面：

- 提出面向人工智能伦理计算的中文道德词典构建任务，将动词、名词、形容词、成语以及汉语的被动表达纳入道德词典体系中，设计了中文道德词典的理论体系，包含四类标签和四种类型。
- 通过词向量扩展和人工核查，构建了包含25,012个词的中文道德词典资源，尽可能全面地覆盖了各类道德行为。
- 为了验证词典的有效性，本文从词的标签及类型识别和判断句子道德倾向两个维度进行了实验。实验结果表明，道德词典资源能够较为准确地判断词的标签和类型，也能够较好地判断句子的道德倾向。

2 相关工作

道德判断是一个传统的哲学问题。近些年来，社会心理学和认知语言学等领域也出现了跨学科的研究，但对道德倾向的大规模形式化处理，特别是道德的分类，仍处于自然语言处理的初级阶段。随着机器获得更多的自主性，需要以更精细的方式来进行伦理计算，使其基于道德进行决策。(Dennis et al., 2016)

道德观念的研究可以追溯到情感分析任务。道德价值观被认为是人格特征更高层次的组织结构。而情感分析中对人格(Schwartz et al., 2013; Yarkoni, 2010)和人类价值(Boyd et al., 2015; Chen et al., 2014)等的评估，为分析人类的道德观念提供了基础。但是，情感词典侧重的是人

的主观情绪，道德词典则是研究客观的事件行为。例如，“杀人”这一事件，人们对它的情感态度可能是“愤怒”、“震惊”等，而道德词典则判断其是一个“不道德”的行为。

目前，道德词典的研究可以分为理论体系设计和词典资源构建两部分。

2.1 理论体系设计

道德基本理论(Moral Foundations Theory, MFT)(Haidt and Graham, 2007; Haidt and Joseph, 2004; Graham et al., 2011)解释了道德的起源、心理基础、发展和文化差异，被广泛应用于计算社会科学领域。该理论定义了五个明确的分类，每个分类包括美德和恶习两个维度，分别如下：

关心/伤害 这一分类与我们作为哺乳动物的长期进化有关，我们拥有依恋系统和感知他人痛苦的能力。它是善良、同情等美德的基础。

公平/欺骗 这一分类与人类社会的互惠性、利他主义有关。它产生正义、不平等和权利等观念。

忠诚/背叛 这一分类与我们部落联盟的悠久历史有关。包含有爱国主义、忠诚、自我牺牲精神等。

权威/颠覆 这一分类由我们长期的等级社会历史形成。包括对社会秩序的维护、对合法权威的尊重和对传统的继承等。

纯洁/堕落 这一分类由宗教观念发展而来，即努力以一种高尚的、不那么肉欲的方式生活。包括贞洁、健康和控制欲望等。

道德基本理论提供了五种典型的道德类型，分类定义非常具体，因此，有大量的道德行为无法简单归类到这五种类型之中，这一点限制了其体系覆盖道德行为的全面性。另外，道德作为一种文化现象，在不同国家及文化背景下有其独特之处，MFT理论并不适宜直接套用于汉语体系。

Jentzsch等(Jentzsch et al., 2019)通过设计道德选择的问题模板，覆盖了更多类型的道德行为。模板从第一人称出发，以问句形式呈现（如表1所示）。标注者将待标注词填入模板，如“我应该杀人吗？”，答案模板为“应该/不该”。这些问题使标注者可以从决策层面判断动作行为的对或错。但目前的模板问题只适用于动词，难以对名词、形容词等其他词类进行判断，限制了其体系覆盖道德行为的规模。

问题	答案
XX是可以的吗?	是/否
我应该XX吗?	应该/不该
我必须XX吗	是/否
我可以XX吗?	是/否
我被允许XX吗?	是/否
XX是被提倡的吗?	是/否
XX是被要求的吗	是/否
XX是礼貌的吗?	是/否
XX是好的行为吗?	是/否
XX是一种典范行为吗	是/否

Table 1: 道德选择问题模板

道德维度	美德	邪恶
关心/伤害	95(16)	85(35)
公平/欺骗	69(26)	57(18)
忠诚/背叛	99(29)	72(23)
权威/颠覆	160(45)	101(37)
纯洁/堕落	97(35)	161(55)
合计	520(151)	476(168)

Table 2: MoralStrength分类及扩展情况（括号内为扩展前数量）

2.2 资源建设

第一个用词汇进行道德判断的语言资源是道德基础词典(Moral Foundations Dictionary, MFD)(Graham et al., 2009)。该资源使用道德基础理论的分类体系和极性标签。词典包含151个美德词和168个邪恶词，词典构建完全依赖人工标注，构建成本较高，不利于扩大规模。

MoralStrength(Araque et al., 2020)在道德基础词典分类的基础上，通过WordNet词汇数据库对MFD进行了扩展。扩展前后各分类的分布情况如表2所示。经过人工标注核查后，得到了包含520个美德词和476个邪恶词的数据集。

从以上研究可以看出，现有的英文道德词典词性构成单一，覆盖的道德行为较少，不利于扩大规模；且分类标准立足于英语文化，不能直接应用于汉语。因此，本文使用汉语文化思

维，将词性类型扩大到动词、名词、形容词、成语以及汉语中的被动表达，丰富中文道德词典词性构成，覆盖更多的道德行为。

3 道德词典理论体系

本文将道德定义为具有普适性的行为规范，分为四类标签和四种类型。具体类别及示例如表3所示。

本章将首先介绍道德词典中道德的界定和理论基础，然后介绍道德词典的分类体系，包括标签体系和类型体系两部分。

标签 / 类型	正向道德	负向道德	中性	被动
行为	捐献	拐卖人口	询问	被害
状态	淡泊名利	惨无人道	安于现状	受尽屈辱
属性	传统美德	传销活动	事业	涉嫌诈骗
要素-地点	福利院	黑作坊	加工点	被殖民地
要素-对象	活雷锋	无良商家	儿童	弃婴
要素-媒介	爱心专座	违禁物品	小作坊	被盗物品

Table 3: 道德词典分类体系及示例

3.1 道德界定及理论基础

本文将道德定义为具有普适性的社会行为规范，根据词汇本身体现的行为信息进行道德判断，探讨道德上要求、禁止或允许的事件行为。

一方面，本文将道德的范围限定为具有普适性的社会行为规范，如“恶意谋杀”这一行为，普遍被认为是不道德的。这一限定可以缩小道德判断问题的范围，回避道德困境类问题。自动驾驶汽车是否应该撞向障碍物，危及乘客，以避免与乱穿马路的人相撞？对于这类问题，没有普适性的行为规范进行判断，不在我们讨论的范围内。另一方面，本文以非结果主义中的规则义务论为指导，根据行为本身的特征或行为体现的规则来判断行为本身是否具有道德价值，将道德行为规范定义为“应该做什么”和“不应该做什么”的普遍规则。对非结果主义者来说，杀死某个人是错误的，因为杀人这一行为本身就是错误的；而对结果主义者来说，如果这个人正在杀死另外十个人的路上，这一杀人行为可能是正当的。由于结果主义的道德判断常常需要大量的背景信息，而在词汇级别，大多数词本身无法完整体现出其行为的结果，所以本文认为，以非结果主义作为指导去判断词的道德倾向更为合理。

综上，本文将道德的范畴限制为具有普适性的社会行为规范，并且只根据词汇本身所体现的行为信息进行道德判断。

3.2 道德词典分类设计

本文通过分析汉语中动词、名词、形容词、成语以及汉语中被动表达体现的道德特征，设计了中文道德词典的分类体系。分类体系包括四类标签和四种类型两部分。

3.2.1 道德词典标签分类

本文将道德词典中的标签分为正向道德、负向道德、中性、被动四类，如表4所示。

正向道德 符合社会道德规范的事件行为，即被认为是应该做、需要做、提倡做的好的行为。例如捐献、正能量、福利院、活雷锋等。

负向道德 不符合社会道德规范的事件行为，即被认为是不该做、不能做、禁止做的坏的行为。例如拐卖、惨无人道、传销、违禁品等。

中性 在大多数情况下与社会道德规范无关的事件行为。例如问话、人山人海、闹剧、食品加工点等。

被动 被动发生的、与社会道德规范有关的事件行为。例如被害、弃婴、饱受争议、涉嫌诈骗等。

标签	例词
中性	问话、人山人海、闹剧、食品加工点
正向道德	捐献、正能量、福利院、活雷锋
负向道德	拐卖、惨无人道、传销、违禁品
被动	被害、弃婴、饱受争议、涉嫌诈骗

Table 4: 道德词典标签分类及例词

3.2.2 道德词典类型分类

本文根据词在事件行为中的位置和作用，将道德词典中的类型分为事件行为、事件状态、事件属性以及事件要素四类，并为每个分类设计了对应的问题模板，如表5所示。

事件行为 这一分类针对的是动作行为本身，判断该行为是否符合道德规范，一般表现为动词。例如诈骗、出尔反尔、恶意透支等。

事件状态 这一分类是对事件行为的状态描述或评价，一般表现为形容词。例如合法合规、投机取巧等。

事件属性 这一分类是较为抽象的事件行为，或一系列事件行为的定义总称，一般表现为名词。例如封建思想、敬业精神等。

事件要素 这一分类是常常和事件行为的一起出现辅助因素，一般表现为名词。包括对象、地点、媒介。其中，对象指的是事件行为的参与者，例如逃犯、弃婴等。地点指的是事件行为发生的地点场所，例如黑作坊、非法赌场等。媒介指的是事件行为的媒介手段和工具。爱心专座、假币等。

类型	问题模板	例词
事件行为	是不是可以做/不能做的行为?	诈骗
	是不是应该做/不该做的行为?	出尔反尔
	是不是提倡做/禁止做的行为?	恶意透支
事件状态	是不是对某种道德/不道德行为的状态描述?	合法合规
	是不是对某种道德/不道德行为的评价?	投机取巧
事件属性	是不是某类道德/不道德行为的总称?	封建思想
	是不是抽象的道德/不道德行为?	敬业精神
事件要素	是不是道德/不道德行为发生的地点?	黑作坊
	是不是道德/不道德行为涉及的对象?	逃犯
	是不是道德/不道德行为使用的媒介工具?	爱心专座

Table 5: 道德词典类型问题模板及例词

4 道德词典构建

本文通过词向量扩展的方法生成了备选词表，并对备选词表进行了人工标注，得到包含25,012个词的中文道德词典。

本章首先介绍备选词表的生成方法，然后介绍标注方法及流程，最后对标注结果进行统计和分析。

4.1 备选词表生成

《现代汉语常用词表》(张清源, 1992)是具有权威典范性的中文词表，表中词汇覆盖范围较广，且具有常用性和代表性。本文首先按道德分类体系对《现代汉语常用词表》中的56,008个词进行标注，得到1,164个正向道德词，1,619个负向道德词，构成了包含2,777个词的基础道德词表。

从基础道德词表的标注情况可以看出，道德词的比例仅占《现代汉语常用词表》的5%左右，大量的道德词分散在其他汉语词汇当中。如果由人工一一进行标注，工程量大且构建成本较高。

因此，本文通过词向量对基础道德词表进行了扩展。对词表的扩展基于这样一个假设：如果一个词具有道德倾向性，那么在词向量空间中与其距离相近的词也可能具有道德倾向性。(Mikolov et al., 2013)因此，本文将基础道德词表作为种子，使用腾讯发布的AILab词向量(Song et al., 2018)和gensim(Rehurek and Sojka, 2010)计算找出与基础道德词表中每个词的余弦距离最近的十个词，构成扩展词表。扩展示例如表6所示。然后，将扩展词表去重并按照词性进行筛选，形成最终的备选词表。

种子词	扩展词				
绑架	绑架勒索 绑匪	绑票 遭绑架	劫持 要挟	胁迫 挟持	敲诈 绑架儿童
黑社会	黑帮 混黑道	黑道 黑老大	黑势力 黑道大哥	黑社会组织 黑帮老大	黑社会分子 黑社会老大
惨无人道	毫无人性 非人道	泯灭人心 残忍	惨绝人寰 屠杀	灭绝人性 残害	丧心病狂 兽行

Table 6: 词向量扩展示例

4.2 标注方法

本文参考中文情感词典的构建思路(柳位平et al., 2009; 饶洋辉et al., 2014; 赵妍妍et al., 2010)设计了人工标注的流程，通过一系列步骤保证标注结果的有效性。

标注内容 根据道德词的定义及分类体系，判断待标注词的道德标签（中性、正向道德、负向道德、被动）和类型（行为、状态、属性、要素）。

标注人员 招募10名培养层次为研究生的在校学生作为标注人员。

标注流程 两位标注人员为同一个词进行标注。导入待标注词后，标注员首先需要判断所给词是否具有道德倾向，标出其标签。然后，标注员需要判断所给词的类型。最后，标注员需要对标注词进行检查，检查无误后提交标注结果。最终标注示例如表7所示。

一致性 为保证标注工作的质量，此次标注设计了培训、试标环节来检查一致性。培训内容包括道德理论、词典体系以及标注流程等。标注人员通过培训熟悉标注规范后，对语料进行试标注，试标结果错误率超过10%的标注员将被劝退。正式标注期间标注的一致性为83.6%。每个词由两位标注员进行标注，两人标注结果不一致的词，交由第三个人进行核查，有争议的词将被剔除。

待标注词	标签	类型	备注
黑作坊	负向道德	要素	地点
私吞公款	负向道德	行为	/
弃婴	被动	要素	对象
无私奉献	正向道德	状态	/

Table 7: 数据标注示例

4.3 标注结果及分析

经过标注和一致性检查后，本文共获得有效标注结果25,012个（见表8）。

各标签的分布情况如图1所示。分析结果可得，正向道德词、负向道德词和中性词的规模相似，而被动词所占比例较少，仅占2%。被动类型由于涉及两个以上对象，道德行为比较复杂。经过进一步分析，发现其中正向道德和负向道德的比例大约是1：6，这是由于被动表达中涉及受害者的词汇较多造成的。

各类型的分布情况如图2、图3所示。行为词数量最多，占到词典总数的一半，其他三分类的比例相似。而要素的小类中，对象类明显较多，地点类所占比例最小。

5 道德词典的有效性验证

本文认为，道德词典资源的有效性可以体现在以下两个方面：一是机器能否通过道德词典

类型 \ 标签	正向道德	负向道德	中性	被动	合计
行为	3946	4155	3947	417	12465
状态	1787	1225	1914	26	4952
属性	1199	1069	1730	12	4010
要素-地点	76	27	78	/	181
要素-对象	546	798	712	32	2088
要素-媒介	358	373	582	3	1316
合计	7912	7647	8963	490	25012

Table 8: 道德词典分类结果

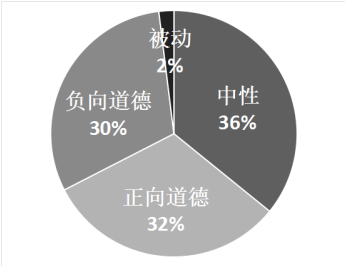


Figure 1: 标签分布

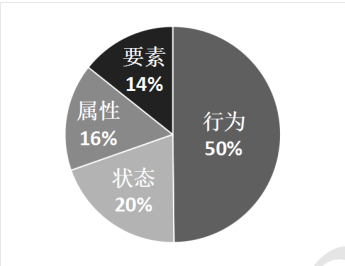


Figure 2: 类型分布

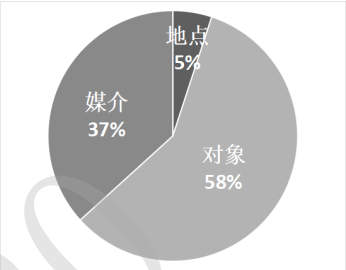


Figure 3: 事件要素分布

识别道德词的标签和类型；二是机器能否通过道德词典识别句子的整体道德倾向，辅助道德文本分析。

针对这两个方面，本文设计了两个对应实验，分别为词的标签及类型识别和判断句子的道德倾向。对于前者，本文使用逻辑回归（logistic regression，LR）和支持向量机（support vector machine，SVM）两种模型。对于后者，本文结合道德词典，使用两种方法对句子的道德倾向进行判断。

5.1 基于道德词典的分类实验

通过道德词标签和类型的识别情况，可以检验机器是否能通过道德词典学习到有效的道德知识。本实验使用带标签、类型信息的道德词典资源对模型进行训练，然后使用预留的测试集对模型的分类识别能力进行评估。

5.1.1 实验数据

本实验将道德词典按照8：1：1的比例划分为训练集、验证集和测试集。两个实验使用共同的测试集，确保不同实验的结果具有可对比性。输入模型的训练数据是词项对应的预训练词向量(Song et al., 2018)。标签信息则依据预测目标的不同，分别对应词典的标签列和类型列。

5.1.2 实验设计

预测词的标签 本实验使用两种思路对词的标签进行预测。

思路一：四分类标签预测。这一思路是将词典中的四种标签分类信息，即中性、正向道德、负向道德和被动，直接作为标签对模型进行训练。

思路二：两步分类标签预测。这一思路是先判断词是否包含道德倾向，即分为道德词（包括正向道德和负向道德）或其他词（包括中性和被动），再对两者分别进行正向道德或负向道德，中性或被动的分类。

思路一和思路二均先使用LR和SVM在验证集上优化参数，再对测试集进行预测，取测试集的结果作为最终的结果。

预测词的类型 本实验使用六分类的方法对词的类型进行预测。

六分类类型预测：本文将行为、状态、属性和要素-地点、要素-对象、要素-媒介六个分类设置为标签对模型进行训练，并对测试集进行预测。

考虑到数据的各个分类数量并不平衡，本文采用weighted average来计算F1。

5.1.3 实验结果及分析

表9展示了LR和SVM在三个实验中预测测试集的F1值。从实验结果可以看出，经过训练后，机器可以较好地掌握道德知识，对未知分类的词进行预测，给出可靠的标签。预测词的标签实验中，四分类标签预测结果好于两步分类标签预测。观察测试集输出结果发现，两步分类标签预测过程中存在误差叠加的问题，即第一步分类中有误差的结果会对第二步分类造成影响。预测词的类型实验中，虽然有的类型数据不平衡，如要素-地点类的数据量较少，但是模型仍能给出比较理想的结果。

实验名称	LR	SVM
四分类标签预测	0.75	0.81
两步分类标签预测	0.73	0.80
六分类类型预测	0.76	0.80

Table 9: 道德词典分类实验结果

三个实验中，分类结果较好的模型对测试集预测结果的混淆矩阵热力图如图4、图5所示。可以看出，直接四分类和两步分类实验中，中性类与负向道德、正向道德标签识别的错误率均比较高。类型分类的混淆矩阵热力图如图6所示。可以看出，行为类和状态类的区分较为困难。

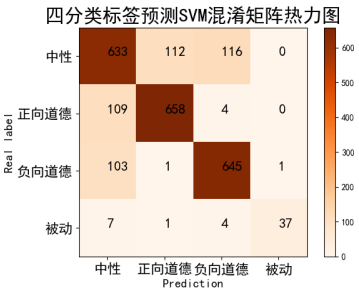


Figure 4: 四分类标签预测

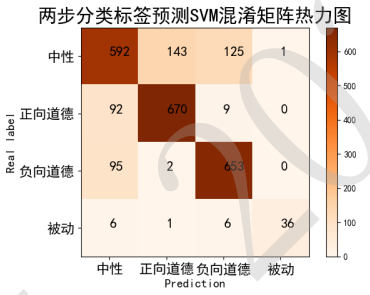


Figure 5: 两步分类标签预测

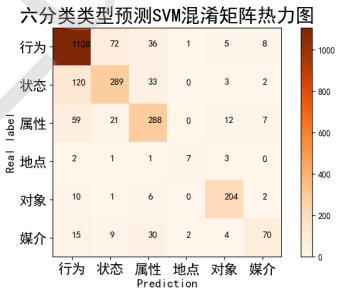


Figure 6: 六分类类型预测

对此，本文统计了人工标注环节标签和类型标注不一致的情况。从图7可以看出，最容易混淆的标签分别是中性-负向道德和中性-正向道德，说明正确区分中性-正向道德和中性-负向道德是一个难点，即使对人类标注者而言也很容易出错。从图8可以看出，容易混淆的类型是状态和行为，占不一致样本总数的40.2%。和图5中反映的规律相类似。

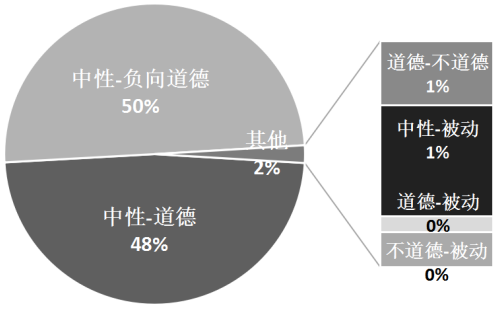


Figure 7: 人工标注标签不一致情况

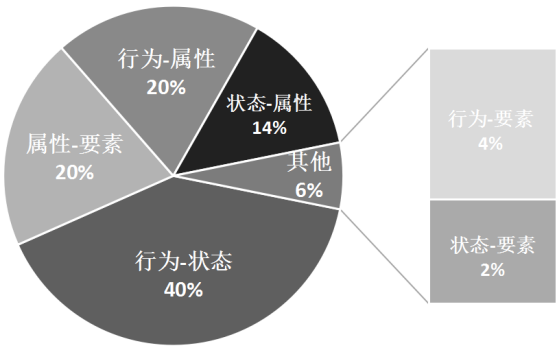


Figure 8: 人工标注类型不一致情况

此外，如表10所示，对一些有一定难度、存在迷惑性的词项，模型也可以很好地做出判断。如“不随地吐痰”和“随地吐痰”非常类似，但是标签相反，很容易误判为负向道德标签等。

5.2 基于道德词典判断句子道德倾向

本实验通过结合道德词典，使用两种方法判断句子总体的道德倾向性，以验证道德词典对判断句子道德倾向的有效性。

5.2.1 实验数据

本实验使用社会新闻类报道用作判断的句子来源，对2018-2019年新浪新闻社会新闻版块的新闻数据进行了爬取，收集了10.4万篇新闻数据。然后按句拆解每篇新闻，保留结构完整且句长为10-45词的句子，作为判断的来源句集。

5.2.2 实验设计

本实验使用两种方法对句子道德倾向进行判断。

方法一：对照(Yuan et al., 2013)使用情感词典对句子进行情感分析的方法，建立基于道德词典的道德分析。首先使用HanLP(He, 2020)对句子进行分词，分词前将道德词典添加到分词器的用户词典中，确保不会将道德词典中较长的词组切分开导致召回率下降；然后根据句子分词结果，统计其与道德词典中正向道德、负向道德、中性和被动标签的对应情况，将对应数量最多的标签作为句子整体的道德倾向。

方法二：道德词典与依存句法分析结合的道德分析。在方法一的基础上，对含有道德词典中词的句子进行依存句法分析。依存句法理论认为，每个句子中存在一个唯一的中心词，支配着句子中其他所有的词，其他词直接或间接依赖于中心词。同时，句子中除了中心词外，每个词都只被一个词支配(计峰and 邱锡鹏, 2009)。通过对句子做依存句法分析，可以理解句子中各部分的关系，以及各部分在各自关系中扮演的角色。根据不同类型的词在句子中常扮演的角色，本实验将依存句法分析中的几种关系与道德词典中的类型建立联系，对应关系如表11所示。其中，依存句法分析的关系体系中没有与地点和媒介两种类型较为合适的对应关系。最后，将句中词项对应的依存关系与该词项在词典中类型的对应关系进行对比，取词项类型对应的标签为句子的整体道德倾向。

词项	标签	预测的标签
妒贤	负向道德	负向道德
不随地吐痰	正向道德	正向道德
发生争吵	中性	中性
受到破坏	被动	被动

Table 10: 词项标签预测结果示例

类型	依存句法关系
行为	核心关系
状态	状中关系
属性	定中关系
要素-对象	主谓关系
要素-地点	/
要素-媒介	/

Table 11: 类型与依存句法对应关系

5.2.3 实验结果及分析

本实验从10.4万篇新闻报道中进行抽取，得到1,627,123条句子供处理。由于这些句子都是没有标签的数据，为了对方法一和方法二输出的结果进行评估，我们从两种方法的结果中各随机抽取了400条结果进行了人工标注，得到的结果如表12所示。

从表12可以看出，结合道德词典做词匹配的方法可以得到还不错的正确率，而结合依存句法分析的方法可以更为可靠地对句子的道德倾向进行判断。实验结果证明了道德词典在判断句子道德倾向上的有效性。

判断方法	正确率
方法一	65.67%
方法二	71.30%

Table 12: 两种判断方法结果

判断结果与句子实际道德倾向性不一致的情况，除去一些不满足普适性道德判断的句子，如有争议性的国际新闻等，比较典型的错误如：句子讨论的话题涉及道德，但句子整体的道德

倾向与句中道德词的倾向不同。如表13所示，方法一的例句中含有两个负向道德词，被判断为负向道德，但句子整体的道德倾向偏向于中性；方法二的例句中含有负向道德词“惯匪”，但句子整体的道德倾向是中性的。这个问题涉及句子中的语义信息，从词汇层面很难解决，未来我们会结合句子的语义信息进行完善。

判断方法	例句-错判标签	对应词项-标签
方法一	请广大群众做到不造谣、不传谣、不信谣。-负向道德	造谣-负向道德, 传谣-负向道德
方法二	社区人士呼吁广大民众对此类模式作案提高警惕，切勿因为嫌麻烦给钱了事，以免惯匪越发猖獗。-负向道德	惯匪-负向道德

Table 13: 道德倾向判断错误样例

6 结语

本文通过对词的道德倾向性进行研究分析，提出面向人工智能伦理计算的中文道德词典构建任务。我们将词典词分为四类标签和四种类型，通过词向量扩展和人工标注构建中文道德词典资源。该词典包含25,012个词，其中正向道德词7,912个，负向道德词7,647个，中性词8,963个，被动词490个。

同时，我们也探讨了道德词典资源的有效性表现。从词的标签及类型识别和判断句子道德倾向两个维度进行了实验设计。实验结果显示，该词典资源不仅能够判断词的标签和类型，而且能够较好地判断句子的道德倾向，为今后句子级别的道德文本分析提供了数据支持。

将社会伦理道德规范与科学技术创新结合是一条漫长的道路。目前词典的分类方法比较粗糙，未来我们会根据词的语义特征进一步细分，深入研究事件行为之间的语义关系，以便更好地解决人工智能伦理计算的道德判断问题。

参考文献

Oscar Araque, Lorenzo Gatti, and Kyriaki Kalimeri. 2020. Moralstrength: Exploiting a moral lexicon and embedding similarity for moral foundations prediction. *Knowledge-based systems*, 191:105184.

Ryan L Boyd, Steven R Wilson, James W Pennebaker, Michal Kosinski, David J Stillwell, and Rada Mihalcea. 2015. Values in words: Using language to evaluate and understand personal values. In *Ninth International AAAI Conference on Web and Social Media*.

Jilin Chen, Gary Hsieh, Jalal U Mahmud, and Jeffrey Nichols. 2014. Understanding individuals' personal values from social media word use. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 405–414.

Scott Clifford and Jennifer Jerit. 2013. How words do the work of politics: Moral foundations theory and the debate over stem cell research. *The Journal of Politics*, 75(3):659–671.

Louise Dennis, Michael Fisher, Marija Slavkovik, and Matt Webster. 2016. Formal verification of ethical choices in autonomous systems. *Robotics and Autonomous Systems*, 77:1–14.

Matthew Feinberg and Robb Willer. 2013. The moral roots of environmental attitudes. *Psychological science*, 24(1):56–62.

Jesse Graham, Jonathan Haidt, and Brian A Nosek. 2009. Liberals and conservatives rely on different sets of moral foundations. *Journal of personality and social psychology*, 96(5):1029.

Jesse Graham, Brian A Nosek, Jonathan Haidt, Ravi Iyer, Spassena Koleva, and Peter H Ditto. 2011. Mapping the moral domain. *Journal of personality and social psychology*, 101(2):366.

Jonathan Haidt and Jesse Graham. 2007. When morality opposes justice: Conservatives have moral intuitions that liberals may not recognize. *Social Justice Research*, 20(1):98–116.

Jonathan Haidt and Craig Joseph. 2004. Intuitive ethics: How innately prepared intuitions generate culturally variable virtues. *Daedalus*, 133(4):55–66.

Han He. 2020. HanLP: Han Language Processing.

Wilhelm Hofmann, Daniel C Wisneski, Mark J Brandt, and Linda J Skitka. 2014. Morality in everyday life. *Science*, 345(6202):1340–1343.

Joe Hoover, Gwenyth Portillo-Wightman, Leigh Yeh, Shreya Havaldar, Aida Mostafazadeh Davani, Ying Lin, Brendan Kennedy, Mohammad Atari, Zahra Kamel, Madelyn Mendlen, et al. 2019. Moral foundations twitter corpus: A collection of 35k tweets annotated for moral sentiment. *Social Psychological and Personality Science*, page 1948550619876629.

Sophie Jentzsch, Patrick Schramowski, Constantin Rothkopf, and Kristian Kersting. 2019. Semantics derived automatically from language corpora contain human-like moral choices. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 37–44.

Rishemjit Kaur and Kazutoshi Sasahara. 2016. Quantifying moral foundations from various topics on twitter conversations. In *2016 IEEE International Conference on Big Data (Big Data)*, pages 2505–2512. IEEE.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Rosalind W Picard. 1997. A ective computing.

Radim Rehurek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Citeseer.

Eyal Sagi and Morteza Dehghani. 2014. Moral rhetoric in twitter: A case study of the us federal shutdown of 2013. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 36.

H Andrew Schwartz, Johannes C Eichstaedt, Margaret L Kern, Lukasz Dziurzynski, Stephanie M Ramones, Megha Agrawal, Achal Shah, Michal Kosinski, David Stillwell, Martin EP Seligman, et al. 2013. Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one*, 8(9):e73791.

Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. 2018. Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 175–180.

Tal Yarkoni. 2010. Personality in 100,000 words: A large-scale analysis of personality and word use among bloggers. *Journal of research in personality*, 44(3):363–373.

Bo Yuan, Ying Liu, and Hui Li. 2013. Sentiment classification in chinese microblogs: lexicon-based and learning-based approaches. *International Proceedings of Economics Development and Research*, 68:1.

张清源. 1992. 现代汉语常用词词典.

柳位平, 朱艳辉, 栗春亮, 向华政, and 文志强. 2009. 中文基础情感词词典构建方法研究. 计算机应用, 29(10):2875–2877.

计峰and 邱锡鹏. 2009. 基于序列标注的中文依存句法分析方法. 计算机应用与软件, (10):44.

赵妍妍, 秦兵, 刘挺, et al. 2010. 文本情感分析. 软件学报, 21(8):1834–1848.

饶洋辉, 李青, 刘文印, and 李晶晶. 2014. 公众文本之情感词典研究进展. 中国科学:信息科学, 44(07):825–835.

汉语否定焦点识别研究：数据集与基线系统

盛佳璇[1]，**邹博伟**[1,2*]，**沈龙骧**[1]，**叶静**[1]，**洪宇**[1]
苏州大学计算机科学与技术学院，苏州，215000[1]
新加坡资讯通信研究院，新加坡，138632[2]
shengjiaxuan1996@gmail.com, zou_bowei@i2r.a-star.edu.sg,
{lxshen.scu,jye.scu,tianxianer}@gmail.com

摘要

自然语言文本中存在大量否定语义表达，否定焦点识别任务作为更细粒度的否定语义分析，近年来开始受到自然语言处理学者的关注。该任务旨在识别句子中被否定词修饰和强调的文本片段，其对自然语言处理的下游任务，如情感分析、观点挖掘等具有重要意义。与英语相比，目前面向汉语的否定焦点识别研究开展缓慢，其主要原因是尚未有中文数据集为模型提供训练和测试数据。为解决上述问题，本文在汉语否定与不确定语料库上进行了否定焦点的标注工作，初步探索了否定焦点在汉语上的语言现象，并构建了一个包含5,762个样本的数据集。同时，本文还提出了一个基于神经网络模型的基线系统，为后续相关研究提供参照。

关键词： 否定焦点；数据集；人工标注

Research on Chinese Negative Focus Identification: Dataset and Baseline

Jiaxuan Sheng[1],**Bowei Zou**[1,2*], **Longxiang Shen**[1], **Jing Ye**[1], **Yu Hong**[1]
School of Computer Science and Technology, Soochow University, Suzhou, 215000[1]
Institute for Infocomm Research, Singapore, 138632[2]
shengjiaxuan1996@gmail.com, zou_bowei@i2r.a-star.edu.sg,
{lxshen.scu,jye.scu,tianxianer}@gmail.com

Abstract

There are a large number of negative expressions in natural language texts. As a more fine-grained negative semantic analysis task, negative focus identification has begun to attract the attention of natural language processing (NLP) researchers in recent years. The task aims to identify the text fragments modified and emphasized by negative cues in the sentence, and it is of great significance to the downstream tasks of NLP, such as sentiment analysis and opinion mining. Compared with English, the study on negative focus identification for Chinese is currently slow, the main reason is that there is no Chinese dataset to provide training and test data for the models. To solve the above issue, this paper carried out the manual annotation of negative focuses on the Chinese Negative and Speculation corpus (CNeSp), initially explored the language phenomena of negative focus on Chinese, and constructed a dataset containing 5,762 samples. Besides, we also come up with a baseline system based on neural network model to provide a reference for subsequent studies.

Keywords: Negative focus , Dataset , Manual annotation

* 通讯作者

1 引言

自然语言文本中存在大量包含否定语义的表达，通常由表示否定语义的词(如“不”、“没有”等)对断言或其某一方面的含义进行反转，这类词被定义为否定线索词(Negative cue)，而被否定线索词强调的文本片段为否定焦点(Negative focus)。更确切地说，否定焦点指表述中最显著被否定的文本片段，从更细粒度上对文本中的肯定含义与否定含义进行了区分 (Blanco and Moldovan, 2011)。在例1中，否定断言为“酒店不提供24小时热水”，并包含否定线索词“不”，而从下文中“下午5点以后才有”来看，该断言中被否定的含义是“提供热水的时限”，而并非“提供热水”本身，因此，根据定义，其对应的否定焦点为“24小时”。需要注意的是，该断言隐含“酒店提供热水”的肯定含义。

例1 酒店**不**提供24小时热水，问了前台说要下午5点以后才有。

否定焦点的另一个特点是：同一个否定断言，在不同的上下文语境中，其否定焦点可能不相同。例如，在例2的三个句子中，根据不同的上下文，否定线索词所强调的语义，即否定焦点，发生了变化。

例2.1 酒店**不**提供24小时热水，但出门左边有浴池提供。[1]
例2.2 酒店**不**提供24小时热水，问了前台说要下午5点以后才有。
例2.3 酒店**不**提供24小时热水，仅能保证冷水供应。

否定焦点这一语言现象最早由Rodney和Pullum提出 (2002)，而Blanco和Moldovan (2011)在自然语言处理领域首次将否定焦点的自动识别作为任务提出，并基于PropBank (Palmer et al., 2005)语料库标注了否定焦点数据集。借助该数据集，*SEM'2012 (Morante and Blanco, 2012)将否定焦点识别作为其评测任务之一，之后该任务开始受到自然语言处理领域学者的关注 (Zou et al., 2014; Zou et al., 2015b; Shen et al., 2019)。作为细粒度的否定语义解析任务，否定焦点的自动识别对用户意图识别或基于属性的情感分析等下游应用具有重要作用。如例2.1中，用户负面评价的对象是酒店；而在例2.2中，用户对酒店的负面评价则更具体，针对的是服务时限。

目前，否定焦点识别的相关研究均面向英语，尚未有文献针对汉语中相关语言现象进行研究。主要归结为以下两方面原因：首先，语料库建设是开展自然语言处理相关研究的基础，由Blanco和Moldovan (2011)标注的数据集为面向英语的否定焦点识别提供了实验基础，而该研究领域尚未有面向汉语的语料资源。另一方面，由于句子结构和表述方式等诸多差异，使得英语中基于规则或特征的否定焦点识别模型和方法难以直接迁移到汉语中。

为了解决上述问题，本文首先在汉语否定与不确定数据集(Chinese Negation and Speculation dataset, CNeSp) (Zou et al., 2015a)的基础上人工标注并构建了汉语否定焦点识别数据集，该数据集共标注4,039个样本，其规模与英语否定焦点识别数据集相当，包括科技文献、酒店评论、金融新闻三个不同领域的文本，能够客观反映汉语表述中否定性语言现象，为相关研究的开展提供基准。其次，本文提出一个基于BiLSTM-CRF网络的否定焦点识别模型，在本文构建的数据集上的准确率为57.8%，与其在英文数据集上的性能相比，性能下降约为13%，表明了汉语否定焦点识别任务的难度，该模型能够作为基线系统为相关研究提供参照。

本文结构组织如下：第二章介绍否定焦点识别的相关工作；第三章介绍本文构建的否定焦点识别数据集，包括标注方法、数据统计等；第四章介绍本文提出的否定焦点识别基线系统；第五章介绍实验设置，给出实验结果并进行分析；第六章给出本文的结论。

2 相关工作

否定焦点识别的相关任务涉及句子的语义研究，属于自然语言处理技术中较难的问题；此外，与句法分析、机器翻译等领域相比，该问题很难标注大规模数据集。因此，针对否定焦点的工作主要集中在利用传统机器学习模型，尝试各种词法、句法和语义等特征。识别句子中否定词及其对应的否定含义，可以划分为否定线索词识别任务和否定焦点识别任务。

否定线索词识别研究主要包含基于词表、基于统计和基于序列标注的方法。基于词表的方法主要依赖于构建和扩充线索词词表算法，在识别线索词时，需严格匹配命中词表或词典中的词项，因此最终的识别性能主要受词表或词典质量的影响。例如，Kilicoglu和Bergler (2008)将

[1]本文中，否定线索词采用粗体表示，否定焦点采用下划线表示。

从生物医学领域的专业词表及WordNet中提取的事实性和不确定性概念之间的词法以及语义关系加入词表。基于统计的方法的关键在于如何提取各类有效的句法、词法、语义等特征，同时将它们组合或者筛选。Light等 (2004)最先使用此类方法，他们利用词特征，并采用了支持向量机辨别医学论文摘要相关句子内是否具有不确定性信息；之后，Georgescul (2012)利用基于高斯径向基核函数对Light的方法作出改善，并对类别的权重进行调节，以缓解训练数据不平衡的问题。Øvrelid等 (2010)将线索词识别任务视为二元分类任务，并利用了句法、词性等特征进行线索词识别。鉴于线索词可能由多个连续的词组成，因此也有相关研究采用序列标注的方法识别线索词。Tang等 (2010)利用条件随机场模型的序列标注方法和大规模基于边界模型分类器进行数据训练，利用了命名实体、词性等特征。Zhang等 (2010)提出基于标准化特征的最大熵马尔可夫模型的线索词识别方法。Vinczel (2014)搭建了面向匈牙利语的不确定性语料库并把线索词识别作为序列标注任务，引入了语义、语用、词法、句法等特征并提出一个有监督的机器学习方法来识别线索词。总的来说，否定线索词识别任务相对简单，目前该任务的性能达到95%以上的准确率。

否定焦点识别研究最初由Blanco和Moldovan (2011)提出，他们从语义关系角度描述和定义否定焦点，并构建了否定焦点识别数据集，然后提出一种基于决策树的模型对否定焦点进行识别。*SEM'2012 (Morante and Blanco, 2012)将否定焦点识别作为其评测任务之一，然而由于任务难度较大，仅有一家机构提交了实验结果 (Rosenberg and Bergler, 2012)，他们根据不同触发词之间词性的差异，提出了基于启发式规则的方法对否定焦点进行识别，其实验性能达到58.40%(F值)。之后，Zou等 (2014; 2015b)融合上下文特征，提出"词-主题"双层图模型识别否定焦点。随着深度学习的发展，Shen等 (2019)将否定焦点识别作为序列标注问题，提出了一种基于BiLSTM-CRF的否定焦点识别模型，并利用词级别和主题级别注意力机制的方法来更好地捕获句子间的上下文信息，一方面利用基于词级别上下文注意力机制来捕获当前句子中候选否定焦点和上下文句子的关联度，另一方面运用基于主题级别的上下文注意力机制计算当前句子中的每个候选否定焦点和上下文句子在主题上的分布相似度，他们提出的方法在英文否定焦点识别数据集上的性能达到了70.51%(准确率)。

目前，尚未有面向汉语的否定焦点识别研究，其主要原因是缺乏人工标注的数据集，而由于该任务难度较大，无监督的自动标注方法难于实现；另一方面，本文在汉语否定焦点识别数据集的标注过程中发现，该语言现象在英汉两种语言上有一定差别，主要集中在当文本中包含省略现象时，否定焦点可能发生转移。以上原因也说明了汉语否定焦点识别数据集标注的必要性。此外，相对否定线索词识别任务，汉语否定焦点识别研究更具一定挑战性，因此，本文专注于汉语否定焦点识别研究，实验中采用标准否定线索词。

3 汉语否定焦点识别数据集

自然语言处理任务的研究通常依赖于相关数据集，面向英语的否定焦点识别研究已经发布了较为成熟的数据集 (Blanco and Moldovan, 2011)和评测任务 (Morante and Blanco, 2012)。然而汉语否定焦点识别数据集研究较为匮乏，导致了面向汉语的相关研究受到极大限制。针对该问题，本文标注并构建了汉语否定焦点识别语料库。

考虑到不同领域或体裁的语料资源在语言特点上具有一定差别，本文根据以下三个方面选择标注文本：

- 尽可能基于现有的汉语否定数据集进行标注，因为其提供了很多与否定语义相关的标注信息，供标注人员参考，在降低标注任务难度[2]的同时，提高标注效率。
- 不同领域或语境下的文本在语言特点上具有差异性，本文希望通过构建该数据集，能够较全面地涵盖汉语中否定焦点的各种表述方式。
- 标注文本的所属领域中，相同类型的文本数量足够大，以增强该数据集的可扩展性；同时，也为将来能够开展半自动甚至自动标注研究提供可能。

基于此，本文选择现有汉语否定与不确定性语料库(CNeSp)[3]中包含否定线索词的文本作为基础，在其上人工标注否定焦点。目前，该语料库包含：1) 19篇《计算机学报》科技论文，该类型文本在语言表达上相对严谨，歧义性较小，但句子长度一般较长，且句法结构复杂；2)

[2]Blanco和Moldovan标注英文否定焦点识别数据集时，标注一致性仅为0.72。
[3]http://nlp.suda.edu.cn/corpus/CNeSp/

821篇携程网酒店点评文章，由于大多文本由用户撰写，该类型文本语言表达方式多样化，句式简单，但修辞及省略现象较常见；3) 1,311篇新浪金融板块文章，该文本由专业编辑撰写，语言特性介于前两种类型之间。

3.1 标注规范

汉语否定焦点识别数据集标注方法部分参照了Blanco和Moldovan (2011)的标注规范，由标注者根据自己对文本的理解进行标注。由于否定焦点标注涉及较多语言现象，标注难度较大，为此本文为标注者制订了一套标注规范。否定焦点的基本标注概念和准则如下：

- 否定焦点：否定线索词明显强调的句子片段。
- 否定焦点是一段连续文本。

标注最大化原则 当片段中存在多个否定解释时，否定焦点应包含尽可能多的句法成分。如例3所示，否定线索词“不如”对应的否定焦点可能为：a) “好的”，解释为“这个星级酒店不如好的招待所，但差一些的招待所可以”，否定词强调“好的”；b) “招待所”，解释为“这个星级酒店只是不如招待所”，否定词强调“招待所”。根据上下文难以判断哪种解释更合理，此时，本文规定将其合并标注为否定焦点。

例3 这个星级酒店还**不如**好的招待所，只有表面功夫，位置有点喧闹...

上下文最优原则 否定焦点标注首先依赖上下文中的信息和证据。从例2.1-2.3可以看出，当给定不同的上下文信息时，所强调的部分(否定论元)可能不同，因此在标注句子中否定论元时，上下文信息是标注者首先需要参考的。在例4中，若不考虑下文中的信息，“全天”或“暖气”均可以作为否定焦点，而根据下文中“只有晚上提供”和“小旅社还提供24小时暖气呢”就能够判断作者强调的否定含义是“酒店供应暖气的时间不是全天”而非“不供应暖气”，据此，标注者将该句子的否定焦点标注为“全天”。

例4 酒店居然**没有**全天的暖气，问前台说只有晚上提供，就算小旅社还提供24小时暖气呢。

积极意义优先原则 标注者标注否定焦点时，优先考虑否定表述中是否存在潜在积极意义。如果将“这家酒店”或“酒店”作为否定焦点，则失去了该表述的潜在积极意义，即“下次去郑州会住酒店”，因此，应仅标注“这家”为否定焦点。注意，该原则与标注最大化原则在不同理解下存在冲突，当无法判断时优先采用标注最大化原则。

例5 下次去郑州一定**不会**入住这家酒店。

动词性否定(Verbal Negation)中，否定线索词作用在动词上，但通常作者并非否定该动词表示的动作或事件，因此，需根据其具体含义标注否定焦点。例6.1中，“修”这一动作已经发生，否定线索词“没”否定了动作产生的结果，因此，其否定焦点标注为“好”。如果忽略上下文，通常情况标注者会将例6.2中的否定焦点标注为“正常工作”，而从上文来看，作者强调的是只有卫浴设备不能工作，其否定焦点应为“卫浴设备”。因此，标注否定焦点时，首先考虑上下文最优原则。针对例6.3中的转述动词(Reporting Verbs)或引述动词(Introducer)，如“知道”、“认为”、“说”等，否定焦点通常不是动词本身，而应从其转述内容中进行标注。

例6.1 门会发出报警，修了几次也**没**修好。
例6.2 其它设施可以使用，只有卫浴设备**不能**正常工作。
例6.3 这是一家相当差的酒店，**不**知道其他人是如何评价它的。

副词标注 汉语中副词的用法多种多样，在早期标注过程中，关于副词是否被标注有较大争议。除了相对明显的由副词本身作为否定焦点(例7.1)，本文对其它情况进行了规定：1) 程度副词具有实际含义时，是否将其标注为否定焦点主要取决于被其修饰的词。如例7.2所示，“太”作为程度副词，说明“近”的程度，因此将“太近”作为一个整体，标注为否定线索词“不”对应的否定焦点。2) 汉语中副词有时在习惯搭配或口语中并不具有实际意义，该情况下，不应将其标注在否定焦点内。如例7.3所示，“太”仅表示委婉语气，而并非修饰“好”，否则其含义为“周围环境不是非常好”，与作者原意“周围环境不好”不一致，因此该情况下，本文认为否定焦点应标注为“好”。

例7.1 酒店装修大多集中在下午，对客人的休息**没有**产生严重影响。
例7.2 到公园的路程**不**算太近。
例7.3 周围环境**不**太好，被噪声干扰到很晚。

3.2 标注中的特殊情况

不存在明显否定焦点 一般来说，本文规定否定焦点不应包含其对应的否定线索词。而在一些样本中，不存在明显的否定焦点，但确实包含否定语义。例8中，“没有”具有否定含义，但可能是由于评论者情绪较差，导致表达中省略了关键成分，若补全可能为“没有好感”或者“没有好的看法”，而该文本中并没有合适作为否定焦点的片段，因此该情况下，我们将否定线索词直接作为否定焦点。

例8 别问我对这家酒店怎么看，**没有**，感觉很差！

否定词不表示否定含义 特殊语境下，否定词并不表示否定含义。如例9.1中，否定词“不是”在反问句中仅具有强调语气的功能；同样，例9.2中，评论者采用双重否定的修辞方式“不是没有”来强调其中的肯定含义。本文认为这些情况中并不存在否定语义，因此也不存在否定焦点。在标注过程中，我们发现科技论文和金融文章文本中，由于表述相对严谨，极少出现反问或多重否定等修辞，而在酒店评论这类口语化程度较高的文本中存在较多。

例9.1 本人问服务员怎么没有荤菜肉菜，答复：鸡蛋**不是**荤菜?
例9.2 酒店管理混乱，外来人员都可以进电梯，看来丢东西这种事**不是没有**可能。

以上涵盖了否定焦点标注主要规则，以及针对标注中出现的特殊情况的处理，由于篇幅原因，更详细的标注规则将随同数据集一起发布。

3.3 标注过程

汉语否定焦点识别数据集标注工作由三位在本领域具有两年以上经验的硕士研究生(标注者)完成标注，并邀请一位从事该领域研究多年的计算语言学专家(指导者)参与数据集标注规则的制定、定期讨论标注结果、裁判标注结果等工作。标注过程大致分为以下两个阶段。

- 第一阶段：经过调研现有否定焦点识别工作及英文数据集，由指导者与标注者共同讨论和制定初步标注规范；之后，三位标注者两两分组，针对三个不同领域的文本，各抽取30%的样本进行标注，期间只允许标注者与指导者进行交流，而标注者之间禁止讨论；标注完成后，指导者与标注者针对标注结果不一致的样本进行讨论，对标注规范进行调整和修改，并将修改后的该30%的数据集作为最终标注结果。
- 第二阶段：仍由三位标注者两两分组，对三个领域剩余70%的样本进行标注；该期间同样只允许标注者与指导者进行交流；标注完成后，由另外一位标注者对其他两位标注者不同的标注结果进行重新审核，如果其与其中一位标注者的标注结果相同，则该结果作为最终标注结果，如果三位标注者结果均不同，则提交给指导者进行判断，给出最终标注结果；同时也会对标注规范进行修订。

最终以第二阶段修订后的标准作为数据集标注规范。标注过程中，尽管标注人员遵循统一的标注规范，但由于每个人对上下文语境理解存在差异且受限于标注者语言知识水平，导致数据集的标注结果存在不一致现象。标注一致性是用来评价数据集质量的重要指标，能够验证标注者在同一标注规范下对标注结果的主观性差异程度以及问题本身的难易程度。本文采用Kappa值 (Mchugh, 2012)评价标注的一致性，计算公式如下：

$$Kappa = \frac{P_0 - P_c}{1 - P_c} \tag{1}$$

其中，P_0和P_c分别表示观察一致率和期望率，$Kappa \in [-1, 1]$。通常认为$Kappa$值大于0.6表示标注具有较好的一致性，而$Kappa$值小于0.4表示标注一致性较差。

本文以否定焦点样本为单位，当两位标注者标注的否定焦点完全相同时，认为该样本的标注结果一致。表1给出了标注第二阶段三个子数据集上各自的Kappa值。可以看出，三个子数据集的标注一致性均在0.65-0.7之间，一方面，表明该语料的标注结果是可靠的，另一方面，也

子数据集	Kappa值
科技论文	0.69
酒店评论	0.65
金融文章	0.68

表 1: 汉语否定焦点识别数据集标注一致性

表明否定焦点识别任务具有较大的挑战性。此外，科技论文子数据集上的标注一致性最高，而酒店评论子数据集的标注一致性最低，这说明在表述相对严谨的文本上，否定焦点比较容易判断，而在口语化表达较多的酒店评论子数据集上，则存在较大的争议，其否定焦点的识别难度也较高。

3.4 数据集统计与分析

汉语否定焦点识别数据集包含科技论文、酒店评论、金融文章三个子语料库，共4039句，5762个样本。表2展示了该数据集的相关数据统计。比较三种不同领域的数据集，可以看出，酒店评论数据与另外两种类型的数据相比，其句子长度更短，这是因为该语料来源于携程网1500家酒店的评论页面，造成该语料表达方式较为口语化，写作风格相对自由。此外，从统计数据可以看出，酒店评论语料中否定句子数、否定句子占比、以及否定样本个数也远高于其它两类数据，造成这一现象的原因是酒店评论中包含了大量评论者的负面观点或意见，这也表明正确识别这些评论或意见中的否定焦点对情感分析(Jansen et al., 2009)、观点挖掘(Dundar et al., 2018)等领域具有重要意义。

统计项	科技论文	酒店评论	金融文章
文档数	19	821	1311
句子数	4626	4997	7213
词数	140900	120436	221265
句子平均长度	30.4	24.1	30.6
否定句子数	132	2644	1263
否定句子占比%	2.9	52.9	17.5
否定样本个数	161	3985	1616
否定线索词集合	25	121	200
否定线索词平均长度	1.58	1.47	1.86
否定焦点平均长度	6.25	2.86	4.34

表 2: 汉语否定焦点识别数据集统计

科技论文数据集中，否定句子数/占比和否定线索词集合大小均远低于其它两种类型的数据集，可以看出，在科技论文中，作者在使用否定或负面表述时较为谨慎。此外，科技论文类型的否定焦点平均长度高于其它类型的数据集，也表明了科技论文具有叙述更完整、论证更严谨的语言特点。

金融文章数据集来源于由经济分析师或财经记者撰写的股市评论类文章，其否定线索词的平均长度大于其它类型的数据集，这是由于作者在表达否定观点时，倾向于使用更长的否定词使表述更加精确或完整，例如“不容乐观”、“并不是”、“并没有”等。

4 否定焦点识别任务基准模型

否定焦点是一段连续的文本。因此，本文将否定焦点识别作为序列标注任务，采用基于BiLSTM-CRF的神经网络模型作为基准模型。如图1所示，其中，Bi-LSTM网络能够同时有效利用句子的上下文信息并抓取全局特征，而CRF层能够充分学习输出标签序列间的前后依赖关系。本节将按照Embedding层、Bi-LSTM层、CRF层和输出层的方式详细介绍否定焦点识别方法。

Embedding层 给定句子$S = (w_1, w_2, ..., w_n)$, w_i表示句子S中的词，其中$i \in [1, n]$。本文利用向量矩阵W_E将S中的每个词转换成维度为d_w的实值向量，其中$W_E \epsilon R^{d_w \times |v|}$，$|v|$为词表大

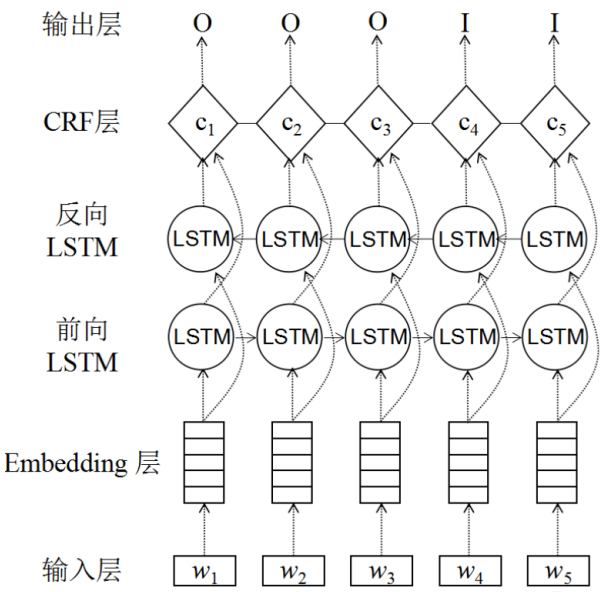


图 1: BiLSTM-CRF模型框架.

小。此外，本文将相对位置特征矩阵与矩阵W_E拼接，作为Embedding层中的输入。即：向量矩阵$F_E \epsilon R^{d_{loc} \times |v_{loc}|}$，其中$d_{loc}$为$S$中的每个词与否定线索词间的相对距离映射的一个实值向量，V_{loc}是相对距离的集合，采用随机初始化。

Bi-LSTM层 传统循环神经网络（Recurrent Neurual Networks, RNN）适合为序列化数据建模，然而，在实际应用中，其受限于梯度消失和梯度爆炸问题 (Bengio et al., 1994; Razvan et al., 2013)，之后为缓解该问题，Hochreiter和Schmidhuber (1997)提出了长短期记忆网络（Long Short-Term Memory, LSTM），该网络能够有效利用长距离依赖关系，缓解冗余上下文信息带来的影响。图2给出了LSTM记忆单元的结构，其由输入门（Input Gate）、输出门（Output Gate）、遗忘门（Forget Gate）和一个细胞状态（Cell）组成，它们控制着当前时刻信息传递到下一时刻的比例。本文将经过Embedding层编码后的矩阵输入Bi-LSTM层，分别从两个相反的方向并行计算，然后将通过前向LSTM网络的矩阵$\overrightarrow{H}$和通过后向LSTM网络的矩阵$\overleftarrow{H}$进行拼接得到矩阵H，如公式(2)所示。该矩阵能够有效的捕获了正向和反向的双向上下文信息。

$$H = \overrightarrow{H} \bigoplus \overleftarrow{H} \tag{2}$$

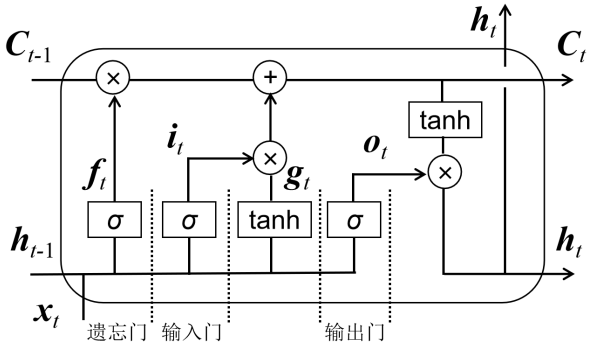


图 2: LSTM记忆单元结构.

CRF层 本文将否定焦点识别作为序列标注任务，通常情况下，一个词的标签生成与其周围词存在一定关联。而条件随机场（Conditional Random Fields, CRF）能够将句子中当前词与周围词的标签关系考虑在内，从而解码出全局最优标签序列。因此，本文将经过Bi-LSTM层

得到的矩阵H输入CRF层解码出最优标签序列。对于给定句子$\boldsymbol{S}=(\boldsymbol{x}_1,\boldsymbol{x}_2,...,\boldsymbol{x}_n)$，其预测标签序列为$\boldsymbol{y}=(\boldsymbol{y}_1,\boldsymbol{y}_2,...,\boldsymbol{y}_n)$，定义其得分为:

$$G(\boldsymbol{S},\boldsymbol{y})=\sum_{i=0}^{n}T_{y_i,y_{i+1}}+\sum_{i=1}^{n}E_{i,y_i}, \tag{3}$$

其中，$\boldsymbol{T}$表示转移得分矩阵，$\boldsymbol{T}_{i,j}$表示从标签i到标签j的转移得分，$y=0$与$y=n$是句子起始标签和终止标签，$\boldsymbol{T}$的维度为$(k+2)\times(k+2)$；$\boldsymbol{E}$是Bi-LSTM的输出得分矩阵，其维度为$n\times k$，其中k为不同标签的数量，$\boldsymbol{E}_{i,j}$表示句子中第i个词的第j个标签的得分。本文采用softmax对所有可能出现的预测序列得分进行归一化表示:

$$p(\boldsymbol{y}|\boldsymbol{S})=\frac{e^{G(\boldsymbol{S},\boldsymbol{y})}}{\sum_{\tilde{y}\in Y_S}e^{G(\boldsymbol{S},\tilde{y})}} \tag{4}$$

然后，对正确的否定焦点标签预测序列对数概率进行最大化:

$$\log(p(y|S))=G(\boldsymbol{S},\boldsymbol{y})-\log(\sum_{\tilde{y}\in Y_s}e^{G(S,\tilde{y})})=G(\boldsymbol{S},\boldsymbol{y})-\log add_{\tilde{y}\in Y_s}G(\boldsymbol{S},\tilde{y}) \tag{5}$$

Y_S表示句子S所有可能预测出的否定焦点标签序列。解码时，将得分最高的标签序列作为句子S最终对应的否定焦点标签序列，如公式(6)所示:

$$\hat{\boldsymbol{y}}=\arg\max_{\tilde{y}\in Y_S}G(\boldsymbol{S},\tilde{y}). \tag{6}$$

输出层 本文采用I/O标记方案来定义CRF层解码出的标签序列，即：标签I表示句子中的词位于否定焦点内；标签O表示句子中的词不属于否定焦点。如例10所示，该句子的否定焦点为“到位”，因此这个词的标记为I，其余词标记为O。

例10 这/O 是/O 服务/O 态度/O **没有**/O 做/O 到位/I 。/O

5 实验

5.1 实验设置

本文实验数据采用第三章介绍的汉语否定焦点识别数据集。实验中，本文采用随机初始化的50维词向量和相对位置特征向量作为Embedding层的输入。模型超参数设置如下：LSTM隐藏层维度为150，dropout设置为0.3，并采用随机梯度下降算法(Stochastic Gradient Descent, SGD)对参数进行更新，其中，学习率为0.015，动量为0.9。本文采用准确率(Accuracy, Acc)作为实验性能评价指标，当一个样本中预测的标签序列全部正确时，则否定焦点正确。

考虑到每个领域的数据集中样本数量较少，本文采用两种不同的训练方式来进行模型的训练，分别为单领域训练方式以及多领域训练方式:

- 单领域训练：对于科技论文、酒店评论和金融文章这三个领域的语料，分别按照70%、10%、20%的比例划分为训练集、开发集和测试集，仅使用单领域的训练集进行训练。
- 多领域训练：将所有单领域的训练集合并，利用合并后的训练集进行训练。

5.2 不同系统对否定焦点识别性能影响

表3给出了基于不同神经网络模型的汉语否定焦点识别性能，同时考虑到虽然三个语料来源于不同领域，但是在汉语语言共性上仍然存在一定关联。因此，本文在汉语否定焦点语料上分别尝试了单领域和多领域两种不同的训练方式。同时，本节对不同语料间的差异及不同系统在三个语料上的实验进行了详细分析。

不同语料之间的对比。表3中2-4行实验结果可以看出：1）不同领域之间的否定焦点识别性能存在着一定的差异性，酒店评论语料上的实验性能最好，而科技文献语料上的实验性能最低，造成这一现象的主要原因是酒店评论表达方式较为口语化，其写作风格相对自由，因此句

语料	单领域训练			多领域训练
	LSTM	BiLSTM	BiLSTM-CRF	BiLSTM-CRF
科技论文	24.24	27.27	30.30	**39.39**
酒店评论	49.25	54.38	59.27	**62.53**
金融文章	40.43	48.15	57.10	**60.18**
全部	46.07	51.87	57.84	**61.21**

表 3: 汉语否定焦点识别模型性能比较

子平均长度以及否定焦点长度更短，模型更容易识别出否定焦点。而科技论文写作方式更为严谨，该语料中句子长度和否定焦点长度在三个语料上最长，导致模型在学习时更容易遗忘长距离信息，所以系统识别性能较低；2）面向汉语的否定焦点识别难度较大，挑战性比较高，虽然酒店评论和金融文章识别性能相对较好，最好性能达到60%以上，但是整体识别性能均没有超过65%，这也验证了本文对否定焦点识别任务难易程度所作的判断。

不同模型之间的对比。为验证不同模型对识别性能的影响（表3中第2列的3个子列），本章对LSTM、BiLSTM和BiLSTM-CRF三个模型进行性能上的比较，从实验结果可以看出，1）BiLSTM模型的性能均比LSTM模型性能高，主要原因是BiLSTM模型考虑了前后两个方向的信息，相较于单向的LSTM模型，它能更充分的利用上下文信息；2）BiLSTM-CRF模型相较于BiLSTM模型，在三个语料上实验性能分别提高了3.03%，4.89%，8.95%，这表明CRF层对于否定焦点识别的有效性，因为否定焦点通常是一段连续的文本，相邻词之间具有较强的依赖关系，而CRF层能够有效捕捉当前词标签与周围词标签之间存在的关联，然后解码出全局最优的标签序列。

同时，本文发现，相较科技论文和酒店评论语料，模型在金融文章上的性能提升尤为显著，其原因可能是：1）科技论文语料中没有足够的否定样本，所以性能上存在着偶然性和不确定性，且科技论文的否定焦点长度普遍较长（平均长度为6.25），因此识别难度较大，性能提升不明显；2）酒店评论语料中的否定焦点长度较短（平均长度为2.86），因此识别较为容易，所以仅用LSTM模型即可获得较高的识别正确率，而金融文章的否定焦点长度介于其它两个语料之间（平均长度为4.34），该语料对上下文信息和相邻标签之间联系的依赖程度较酒店评论语料更大，因此引入了BiLSTM和BiLSTM-CRF模型后性能提升更为显著。

不同训练方式的对比。多领域训练，即：将来源于三个不同领域的语料训练集统一为一个整体训练集，并利用不同神经网络模型在该训练集上进行了实验。如表3中2-3列，为验证多领域训练方式的有效性，本节将两种不同训练方式的识别性能进行对比，实验结果显示，采用多领域的训练方式后系统识别性能提升显著。这是因为一方面神经网络模型在学习过程中，语料的数量对系统识别性能具有重要影响，而多领域训练方式扩充了数据集，使得模型的鲁棒性得以提升；另一方面，三个不同领域语料之间虽然存在着语义、句法上的差异，但是仍然存在汉语上的语言共性，而神经网络可以有效捕获这部分特征。

6 结论

本文在汉语否定与不确定语料库(CNeSp)的基础上，人工标注了首个汉语否定焦点识别数据集，该数据共包含5,762个样本。此外，本文还提出了一个基于BiLSTM-CRF的基准系统。该数据集和基准系统为汉语否定焦点识别的后续研究提供了基础。未来工作中，如何针对汉语语言特点以及如何将英语中相关模型迁移到汉语否定焦点识别模型中，是需要探索的方向。另外，汉语否定焦点识别数据集和基准系统将同论文一起发布。

致谢

本文工作得到国家自然科学基金（基金号61703293，61672368，61672367），江苏省高校优势学科建设工程资助项目资助。

参考文献

Yoshua Bengio, Simard Y. Patrice and Frasconi Paolo. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks.* 5(2): 157-166.

Eduardo Blanco and Dan Moldovan. 2011. Semantic Representation of Negation Using Focus Detection. *In Proceedings of the 49th Annual Meeting of the Association for Computional Linguistics (ACL),* pages 518-589. Association for Computational Linguistics.

Betül Dündar, Diyar Akay, Fatih Emre Boran and Suat Özdemir. 2018. Fuzzy Quantification and Opinion Mining on Qualitative Data using Feature Reduction. *International Journal of Intelligent Systems.* 33(9): 1840-1857.

Maria Georgescul. 2012. A Hedgehop over a Max-Margin Framework Using Hedge Cues. *In Proceedings of the Fourteenth Conference on Computational Natural Language Learning: Shared Task.*

Sepp Hochreiter and Schmidhuber Jurgen. 1997. Long short-term memory. *Neural Computation.* 9(8): 1735-1780.

Bernard J. Jansen, Mimi Zhang, Kate Sobel and Abdur Chowdur. 2009. Twitter power: Tweets as electronic word of mouth. *Journal of the Association for Information Science and Technology.* 60(11): 2169-2188.

Halil Kilicoglu and Sabine Bergler. 2008. Recognizing speculative language in biomedical research articles: a linguistically motivated perspective. *Journal of BMC Bioinformatics.* 9(11):S10.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami and Chris Dyer. 2016. Neural architectures for named entity recognition. *North american chapter of the association for computational linguistics.* pages 260-270.

Marc Light, Xin Ying Qiu and Padmini Srinivasan. 2004. The Language of Bioscience: Facts, Speculations, and Statements in between. *In Proceedings of the HLT-NAACL 2004 Workshop: Linking Biological Literature, Ontologies and Databases.* pages 17-24. Association for Computational Linguistics.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. *In Proceedings of the 54th Annual Meeting of the Association for Computional Linguistics (ACL),* pages 1064-1074. Association for Computational Linguisitcs.

Marry L. McHugh. 2012. Interrater reliability: the kappa statistic. *Biochemia Media.* 22(3):276-282.

Roser Morante and Eduardo Blanco. 2012. *SEM 2012 Shared Task: Resolving the Scope and Focus of Negation. *In proceedings of the First Joint Conference on Lexical and Computational Semantics (*SEM),* pages 265-274. Association for Computational Linguistics.

Xiaofeng Mu, Wei Wang adn Aiping Xu. 2020. Incorporating token-level dictionary feature into neural model for named entity recognition. *Neurocomputing,* pages 43-50.

Lilja Ovrelid, Erik Velldal and Stephan Oepen 2010. Syntactic scope resolution in uncertainty analysis. *In Proceedings of the 23rd International Conference on Computational Linguistics.* pages 1379-1387.

Martha Palmer, Daniel Gildea and Paul Kingsbury. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics.* 31(1):71-106.

Pascanu Razvan, Tomas Mikolov and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. *International conference on machine learning.* pages 1310-1318.

Huddlestone Rodney and Geoffrey K. Pullum. 2003. The Cambridge Grammar of the English Language. *Modern Language Review.* 98.3.

Sabine Rosenberg and Sabine Bergler. 2012. UConcordia: CLaC negation focus detection at *SEM2012. *In proceedings of the First Joint Conference on Lexical and Computational Semantics (*SEM),* pages 294-300. Association for Computational Linguistics.

Longxiang Shen, Bowei Zou, Yu Hong, Qiaoming Zhu, Guodong Zhou and Ai Ti Aw. 2019. Negative Focus Detection via Contextual Attention Mechanism. *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP),* pages 2251-2261. Association for Computational Linguisitcs.

Buzhou Tang, Xiaolong Wang, Xuan Wang, Bo Yuan and Shixi Fan 2010. A Cascade Method for Detecting Hedges and their Scope in Natural Language Text. *In Proceedings of the Fourteenth Conference on Computational Natural Language Learning-Shared Task.* pages 13-17.

Veronika Vincze1 2014. Uncertainty Detection in Hungarian Texts. *In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics.* Association for Computational Linguistics.

Shaodian Zhang, Hai Zhao, Guodong Zhou and Baoliang Lu 2010. Hedge detection and scope finding by sequence labeling with normalized feature selection. *In Proceedings of the Fourteenth Conference on Computational Natural Language Learning-Shared Task.* pages 92-99.

Bowei Zou, Guodong Zhou and Qiaoming Zhu. 2014. Negation Focus Identification with Contextual Discourse Information. *In Proceedings of the 52nd Annual Meeting of the Association for Computional Linguistics (ACL),* pages 522-530. Association for Computational Linguistics.

Bowei Zou, Guodong Zhou and Qiaoming Zhu. 2016. Research on Chinese negation and speculation: corpus annotation and identification. *Frontiers of Computer Science in China* 10(6): 1039-1051.

Bowei Zou, Qiaoming Zhu and Guodong Zhou. 2015. Negation and Speculation Identification in Chinese Language. *In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL),* pages 656-665. Association for Computational Linguistics.

Bowei Zou, Qiaoming Zhu and Guodong Zhou. 2015. Unsupervised Negation Focus Identification with Word-Topic Graph Model. *In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP).* pages 1632-1636. Association for Computational Linguistics.

面向医学文本处理的医学实体标注规范

张欢[1,2]，**宗源**[1,2]，**常宝宝**[1,2]，**穗志方**[1,2]，
昝红英[2,3]，**张坤丽**[2,3]
（1. 北京大学计算语言学教育部重点实验室，北京100871；
2. 鹏城实验室，广东深圳518055
3. 郑州大学信息工程学院，河南郑州450001）

摘要

随着智慧医疗的普及，利用自然语言处理技术识别医学信息的需求日益增长。目前，针对医学实体而言，医学共享语料库仍处于空白状态，这对医学文本信息处理各项任务的进展造成了巨大阻力。如何判断不同的医学实体类别？如何界定不同实体间的涵盖范围？这些问题导致缺乏类似通用场景的大规模规范标注的医学文本数据。针对上述问题，该文参考了UMLS中定义的语义类型，提出面向医学文本信息处理的医学实体标注规范，涵盖了疾病、临床表现、医疗程序等9种医学实体，以及基于规范构建医学实体标注语料库。该文综述了标注规范的实体体系、标注细则、混淆处理、语料标注以及医学实体自动标注基线实验等相关问题，希望能为医学实体语料库的构建提供可参考的标注规范，以及为医学实体识别提供语料支持。

关键词： 智慧医疗；医学实体；标注规范；标注语料

Medical Entity Annotation Standard for Medical Text Processing

ZHANG Huan[1,2]，**ZONG Yuan**[1,2]，**CHANG Baobao**[1,2]，
SUI Zhifang[1,2]，**ZAN Hongying**[2,3]，**ZHANG Kunli**[2,3]
（1. Key Laboratory of Computational Linguistics，Ministry of Education，
Peking University，Beijing 100871，China；
2. Peng Cheng Laboratory，Shenzhen，Guangdong 518055，China；
3. School of Information Engineering，Zhengzhou University，
Zhengzhou，Henan 450001，China）

Abstract

With the popularization of smart healthcare, the demand of applying natural language processing technology to identify medical information is increasing day by day. At present, there is no unified annotation standard for medical named entities in China, and the medical shared corpus is still in a blank state, which causes great resistance to the progress of medical text information processing tasks. How to judge different categories of medical entities? How to define the coverage of different entities? These problems lead to the lack of a similar mass of general scenario standard of medical text data. In view of the above problems, We referred to the semantic types defined in UMLS and proposed a unified medical entity annotation standard for medical text processing, covering 9 kinds of medical entities such as disease, symptom, medical procedure and so on，and constructed medical entity annotated corpus based on standards. This paper summarizes related issues such as the entity system, annotation principles, obfuscation processing, corpus annotation process, and medical entity automatic labeling baseline experiments, hoping to provide reference for medical entity corpus build annotating standard, as well as the medical support the corpus entity recognition.

Keywords: Smart healthcare , Medical entity , Annotation standard , Annotated corpus

1 引言

近年来，互联网和数字化已为众多行业带来颠覆性变革，医疗健康领域也不例外。伴随着智慧医疗的到来，在很大程度上改进了医院的管理及运营模式、改进了对大众的医疗服务。

医学领域存在大量自然语言文献，例如医学教材、医学百科、临床路径、病历、医学期刊、检验报告等，这些医学文本中蕴含了大量的专业知识和丰富的医学信息。医学领域中的命名实体识别指的是将重要的医学实体，如疾病、症状等从医学文本中抽取出来，这个步骤也是医学关系提取等各项任务的基础。

命名实体识别的主要技术方法分为：基于规则和词典的方法、基于统计的方法、二者混合的方法。基于规则和词典的方法是命名实体识别中最早使用的方法，但规则往往过于依赖知识库，故而充满局限性。基于统计的方法利用人工标注的语料进行训练，现已成为目前研究的主流方法。对于医学实体而言，医学共享语料库仍处于空白状态，这对医学文本信息处理各项任务的进展造成了巨大阻力。目前针对不同的标注任务，其医学实体标注规范各有不同，医学实体的分类也是大不相同。如何判断不同的医学实体类别？如何界定不同实体间的涵盖范围？这些问题导致缺乏类似通用场景的大规模规范标注的医学文本数据。因此亟需建立医学实体标注的规范，并以此建立医学实体标注语料库。

基于这样的前提，本文做出的主要贡献有：

（1）制定了面向医学文本信息处理的医学实体标注规范。以更加细致的划分方式将医学实体划分为九大类，包含“疾病”、“临床表现”、“医疗程序”、“医疗设备”、“药物”、“医学检验项目”、“身体”、“科室”和“微生物类”。

（2）基于规范进行了约263万字的儿科类医学教材的语料标注，在标注质量评测方面，采取迭代标注、抽样检查等多种措施，提高标注效率和标注质量。

（3）构建了一系列医学实体自动标注基线实验，取得了F值为71.0%的标注性能，为后续的工作打下了基础。

2 相关工作

一体化医学语言系统（Unified Medical Language System；UMLS）是美国国立医学图书馆(NLM)自1986年起研究和开发的一体化医学语言系统 (Bodenreider O, 2004)，是对生物医学科学领域内术语词表的统一汇编，并提供了UMLS数据库，如超级叙词表、语义网络和专家词典，以及相关软件工具，如MetamorphoSys、MMTx等。2010 i2b2/VA challenge (2010)会议发布了电子病历命名实体的分类，该会议参考UMLS定义的语义类型，将医学实体分为3种：医疗问题（Medical Problem）、检查（Test）和治疗（Treatment）。Roberts et al. (2007)等人随即选择了50份临床记录、X射线和病理报告进行标注，将这些医学文本中的医学实体分为6种：状况（Condition）、药物（Drug）、干预（Intervention）、部位（Locus）、检查（Investigation）、结果（Result）。South et al. (2009)使用了316例炎症性肠病的临床记录进行标注，其中医学实体种类分为4种：体征或症状（Signs or symptoms）、诊断（Diagnoses）、程序（Procedures）和药物（Meditations）。

相比国外对医学语料库的构建以及相关任务的展开，国内没有公开可获得的面向医学实体识别的数据集。2014年Lei et al. (2014)等人使用北京协和医院2013年的电子病历进行标注，其中医学实体分为4种：医疗问题、治疗程序、药物和检查。2014年，Xu et al. (2013)使用一家中国医院提供的336个个出院总结进行标注，其中医学实体同样分为4种：医疗问题、治疗程序、药物和检查。2016年，哈尔滨工业大学团队 (2016)使用来自哈尔滨医科大学附属第二医院的122个科室的电子病历进行标注，并且制定了新的中文电子病历命名实体和实体关系标注规范。该规范将医学实体分为5种：疾病、疾病诊断分类、症状、检查和治疗。2019年，Gao et al. (2019)等人使用了255份来自中国湖南省某著名医院的真实入院记录进行标注，并在其论文

中提出的新的标注方案，将医学实体分为9种：医疗发现、症状、时间词、疾病、身体部位、药物、治疗、实验室检查和（非实验室）检查。

本文旨在提出面向医学文本处理的医学实体标注规范，并将医学实体划分为九大类，包含“疾病”、“临床表现”、“医疗程序”、“医疗设备”、“药物”、“医学检验项目”、“身体”、“科室”和“微生物类”。回顾以上中英文医学文本语料库标注准则，大部分都将“检查”和“治疗”视为医学实体。本文认为，“检查”和“治疗”应该是作为医学实体之间的某种关系，比如“某种药物治疗某种疾病”、“使用某种设备检查身体部位”或“检查哪种检验项目”等等，从而建立“底层”独立实体之间的“高阶”关系，“实体”和“关系”是相互独立的，这样的优势在于让标注者能够更好地理解实体的概念。本文引入的“医疗程序”和“医疗设备”实体就是将“检查”和“治疗”更加“实体化”和“概念化”。一种医疗程序既可以作为治疗某种疾病的过程，也可以作为检查某种疾病是否存在的过程；一种医疗设备同样可以用来治疗疾病，也可以用来检查疾病或身体；等等。此时，疾病、医疗程序、医疗设备和身体作为“底层”实体，而检查和治疗作为“高阶”关系，这样使标注者能够真正建立起医学的概念，在医学实体标注阶段仅仅集中研究实体本身的含义，而不关注实体之间的关系。这种实体的细致划分一方面是为了概念细粒度化，另一方面为后续的实体关系提取提供了良好的数据基础。

3 总体原则

3.1 简单性原则

本文将医学实体划分为九大类，包括“疾病”、“临床表现”、“医疗程序”、“医疗设备”、“药物”、“医学检验项目”、“身体”、“科室”和“微生物类”，并详细介绍了各个医学实体的涵盖范畴，阐述实体间的混淆处理，用大量示例举例说明。标注者无需太多专业知识，实体类别定义简易明了，方便标注者理解和区分。

3.2 易操作性原则

对于医学文本的医学实体标注，标注者需严格遵从本文提出的医学实体标注规范，使用“[named-entity]tag”的方式进行紧密标记（左右括号与标记实体首末字符之间无空格，括号需成对出现），若出现标注实体的英文缩写、中文简称或者俗称，均需要标注，各类医学实体标签如表1中所示。可嵌套标注的实体内部包含的实体应作为方括号嵌套成分，如“[[named-entity]tag XXXXX]tag”。

医学实体类别	标签/tag	备注
疾病	dis	disease
临床表现	sym	symptom
身体	bod	body
医疗程序	pro	procedure
医疗设备	equ	equipment
药物	dru	drug
医学检验项目	ite	item
科室	dep	department
微生物类	mic	microbes

Table 1: 医学实体标记

3.3 一致性原则

医学实体的标注包括实体类型和实体边界两个部分。医学实体的分词存在很多歧义，如何切分较长的疾病或药物名称等，这给标注工作带来了很多困难。对于除“临床表现”这个复杂的医学实体外，人们往往关注的是医学实体的含义，比如什么疾病、什么药物、哪个科室等等，这类医学实体内部无需分词，仅仅作为一个整体来看待。因此本文遵从以下统一原则：

1. “临床表现”实体内部允许分词，并且该实体内部允许嵌套标注，即若“临床表现”实体内部存在其他8种实体，标注者也应该将其标注出来，“临床表现”实体内部其他文本内容的分词

原则同3处理。

2. 除“临床表现”外的医学实体内部不允许分词。标注应当遵循“最大单位标注法”，即若一个实体内包含其他实体，则标注“最大”的实体，不做嵌套标注。.
3. 除以上实体之外的其他文本内容的分词原则遵从《北京大学现代汉语语料库基本加工规范（2002版）》 (2002)

另外，为保证实体意义的完整性和可理解性，所有的实体可以是一个词、短语和句子，实体中可包含标点符号。也就是说，当标点符号存在某种意义时，标注者同样应该将其标注。

4 医学实体体系

本文将医学实体划分为九大类，包含“疾病”、“临床表现”、“医疗程序”、“医疗设备”、“药物”、“医学检验项目”、“身体”、“科室”和“微生物类”，如图1所示。故本文提出的规范对于医学实体的划分上更加细致，这也有便于未来医学实体关系提取等各项工作的开展。本文借鉴UMLS语义类型界定实体涵盖的范围，但不局限于UMLS的定义。

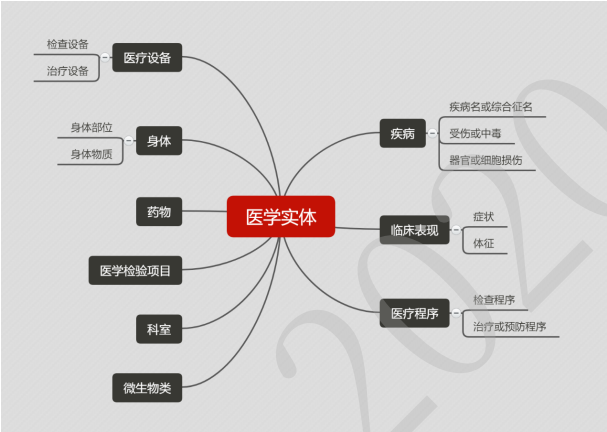


Figure 1: 医学实体架构图

第一类实体是疾病，疾病是指导致病人处于非健康状态的原因或者医生对病人做出的诊断，并且是能够被治疗的 (2016)。包括疾病或综合征、中毒或受伤、器官或细胞受损，其对应的UMLS语义类型有疾病或者综合征（disease or syndrome）、中毒或受伤（injury or poisoning）等；

第二类实体是临床表现，临床表现是疾病的表现，泛指患者不适感觉以及通过检查得知的异常表现。主要包括症状、体征，其对应的UMLS语义类型有症状或体征（sign or symptom）、异常检查结果（abnormal test results）等；

第三类实体是医疗程序，在本文中，医疗程序泛指为诊断或治疗所采取的措施、方法及过程。主要包括检查程序、治疗或预防程序，其对应的UMLS语义类型有化验过程（laboratory procedure）、治疗或预防过程（therapeutic or preventive procedure）、等；

第四类实体是医疗设备，在本文中，医疗设备泛指为诊断或治疗所使用的工具、器具、仪器等。主要包括检查设备、治疗设备，其对应的UMLS语义类型有医疗设备（medical device）、药物传输设备（drug delivery device）等；

第五类实体是药物，药物是指用来预防、治疗及诊断疾病的物质，其对应的UMLS语义类型有临床药物（clinical drug）、抗生素（antibiotic）等；

第六类实体是医学检验项目，医学检验项目是指检查涉及到的体液检查项目、重要生理指标以及其他检查项目，本文规定“医疗检验项目”主要针对人体而言，是能够通过设备或实验检测出的项目，并且是能够被量化，有其对应的测量值或指标值。其对应的UMLS语义类型有实验室检查（laboratory test）等；

第七类实体是身体，身体泛指细胞、组织、及位于人体特定区域的由细小物质成分组合而成的结构、器官、系统、肢体，另外包括身体产生或解剖身体产生的物质等。主要包括身体部位、身体物质，其对应的UMLS语义类型有身体部位（body part）、组织（organ）、组织成分（organ component）等；

第八类实体是科室，科室主要是指医院或医疗机构所设有的科室其对应的UMLS语义类型有医疗保健相关组织（healthcare related organization）等;

第九类实体是微生物类，微生物类包括细菌、病毒、真菌以及一些小型的原生生物、显微藻类等在内的一大类生物群体，另外包括微生物类产生的毒素、激素、酶等，其对应的UMLS语义类型有细菌（virus）、真菌（fungus）等;

5 医学实体标注细则

5.1 疾病

5.1.1 疾病或综合征

疾病或综合征是指疾病或综合征名称。比如高血压、肺炎、心脏病、败血症、畸形等。

① 测定 结果 的 分析： [肺炎]dis 、 [败血症]dis 、 严重 [先天性心脏病]dis 或 [畸形]dis 影响 新生儿 代谢 和 循环 功能， 特别 是 [严重 感染]sym 时， 可 [微循环障碍]dis 和 [DIC]dis

5.1.2 受伤或中毒

患者在受伤或中毒后，对人体造成某种伤害，导致患者处于非健康状态。

① [局部灼伤]dis 处理 与 一般 [烫伤]dis 处理 相同

5.1.3 器官或细胞损伤

器官、细胞等发生异常或损伤后，如果能够危及人的机体，此时虽然它们属于身体的一部分，但是已成为一种致病因素，危害人体健康。

① 还有 [颅内出血]dis （ [产伤]dis 、 [外伤]dis ）， [颅脑损伤]dis ， [脑血管畸形]dis 。

在标注“疾病”实体时，需要注意:

（1）一般有些疾病名称很长，前面会有“XX性”、“XX状”、“XX型”等，以及身体部位（一个或多个）的修饰，在保证疾病完整性和具体性的情况下，在标注时应该与这些前缀一起标注。

① 有些 [急性病毒感染]dis 引起 明确 的 相关性 疾病， 如 [HAV感染]dis 与 [急性黄疸型肝炎]dis ； [轮状病毒感染]dis 与 [季节性婴幼儿腹泻]dis

③ 小儿 患有 [肝脏、肾脏、甲状腺疾病]dis

（2）大部分统称均标注，比如营养性疾病、代谢性疾病、化脓性和非化脓性综合征等，此类虽然是统称，但是有对应的疾病范畴，所以应该标注。特殊情况：像常见病、多发病、疾病等单独出现时，此类统称范围太广，不应标注。

① [高血压]dis 是 严重 危害 人类 健康 的 常见病 、 多发病 （3）当疾病有若干种分型时，“疾病+分型”或“分型+疾病”整体标注，分型单独出现不标注。

① 可将 [MPS]dis 分为 I ～ VII 型， 除 [MPSII型]dis 为 X 连锁 隐性 遗传 外

5.2 临床表现

5.2.1 症状

症状是指病人自己向医生陈述（或是别人代述）的不适或痛苦表现。通常是病人主观感觉的不适，如腹痛、头晕等，或是自己发现的病理改变，如血尿便血、活动障碍等。

5.2.2 体征

体征是指医生观察到的或者通过检查程序或设备检查到的发生于病人的异常变化以及异常检查结果。通常是指医师利用自己的感官（视触叩听）或者医疗器具（血压计叩诊锤等医疗设备）发现的病人的病理生理变化。

在标注“临床表现”实体时，需要注意:

（1）在“临床表现”实体内部，若包含除“临床表现”之外的其他实体（“疾病”、“身体”、“医学检验项目”等），内部实体应作为方括号嵌套成分（如“[[肺动脉干]bod 突出]sym”）。

① 如 出现 [[肺动脉]bod 高压]sym ， [[肺动脉干]bod 突出]sym

（2）如果前后文中有“表现为”、“表现有”、“有”、“不良反应有”、“症状有”、“等症状”、“反应”等描述症状出现的词，则标注对应实体为“临床表现”。

① 不良 反应 有 [[气管]bod 痉挛]sym 、 [[心功能]ite 不全]sym 、 [恶心]sym 、 [呕吐]sym

（3）对于“临床表现”的修成成分，通常表示其严重程度、频率等，为保证标注完整性，在标注时应该将修饰和症状一起标注。

① 遇 紧急 情况 ， [气管]bod 异物 导致 [严重 呼吸 困难]sym

（4）对于“体征”，一般都是通过医疗设备观察到的病理或生理改变等客观表现，因此多出现在表示检查的词后面，比如“见”，“可见”、“及”、“闻及”、“显示”等，这里有两种情况：一种是“医疗程序/医疗设备+（检查词）+体征”，则在标注时仅将对应的体征标注为“临床表现”，不用标注此类检查词；另一种是“身体部位/主体+（检查词）+体征”，为保证标注意义完整，则在标注时应该将“身体部位/主体+（检查词）+体征”作为整体标注成“临床表现”。

① 行 [头颅CT]pro 显示 [[双侧额部]bod 或 [额顶部]bod 有 [蛛网膜下腔]bod 增宽]sym

（5）描述“临床表现”时，通常是对病人（一个或多个）身体部位进行描述，在标注时应该与这些身体部位一起标注。若出现“部位/主体+有/无+临床表现”，应该整体标注为“临床表现”。若出现“无+症状/体征”，“无”作为描述临床表现的一种修饰成分，应该整体标注为“临床表现”。

① 常 伴有 [[膀胱逼尿肌]bod 无 抑制性 收缩]sym ， 其中 25 % 患儿 有 [尿失禁]sym

（6）如果临床表现实体后面紧跟“症状”、“体征”、“反应”，此时应该将实体和此类词语整体标注；若是非临床表现实体后面紧跟此类词语，则不标注；若单独出现“症状”、“体征”、“反应”表示的是一种统称含义，则不标注。

① [[中毒]dis 症状]sym 与 [[颅高压]dis 征象]sym 明显 、 [[神经系统]bod 局灶 定位 体征]sym 出现 ， [神经影像学检查]pro 帮助 诊断 。

（7）临床上的“指征”，一般是指手术指征。在标注工作中，指征标注为“临床表现”。

① 临床 指征 为 ： [血便]sym ； 有 [里急后重]sym ；

5.3 医疗程序

“医疗程序”泛指为诊断或治疗所采取的措施、方法等，包括检查程序和预防或治疗程序。

5.3.1 检查程序

检查程序包括通用检查方法、专项检查、医学影像检查等，检查方法是医生为达到化验目的而采取的某种手段；专项检查是病人通常情况下做的某种检查；医学影像检查是放射科或核医学部门的医疗程序。

① [肝活检]pro 应 争取 在 起病 后 4 ～ 5 日 内 进行

5.3.2 治疗程序

治疗或预防程序是医生为达到治疗目的而采取的某种手段，如化疗、放疗、手术、透析、紧急救治等。

① [静脉注射]pro 用 [丙种球蛋白]dru （ [IVIG]dru ） 对 部分 [狼疮]dis 有 一定 疗效

在标注“医疗程序”实体时，需要注意：

（1）当医疗程序前面有身体部位（对某部位进行检查或治疗），应该整体标注。

① 通常 需 经 [胸部X线平片]pro 进行 诊断

（2）辅助治疗和非药物治疗也标注为“医疗程序”。

5.4 医疗设备

“医疗设备”泛指为诊断或治疗所使用的器具、或仪器等，包括检查设备、治疗设备。

5.4.1 检查设备

检查设备通常指的是医院中用来检查或检验的仪器，比如血细胞分析仪、生化分析仪等。

① 通过 [血细胞分离仪]equ 可 分离 得 [白细胞]bod

5.4.2 治疗设备

治疗设备是医生为达到治疗目的而单独或者组合使用于人体的仪器、设备、器具，如注射器、供养面罩、呼吸器等。

① [血管内支架]equ 在 [先天性心脏病]dis 中 的 应用 ： 常用 [4通道测压导管]equ 。

在标注“医疗设备”实体时，需要注意：

（1）医疗设备的属性不可标注为“医疗设备”。

① 适用 的 [注射器]equ 范围 大 ， 包括 10 ～ 50 ml [注射器]equ 均 可 适用 。

5.5 药物

药物广义上是指用来预防、治疗及诊断疾病的物质，另外也包括临床诊断试剂。
在标注“药物”实体时，需要注意：：
（1）药物的属性不可标注为“药物”。

① 可用 [9α-氟氢可的松]dru 0.05 ～ 0.1 mg / d

（2）大部分药物的统称均标注，比如营养素、抗菌药物、急救药物等，此类虽然是统称，但是有对应的治疗范畴。像常用药、药物等单独出现时，此类统称范围太广，不应标注。

① 配备 [急救药物]dru 可 增加 安全性 。

5.6 医学检验项目

检查涉及到的体液检查项目、重要生理指标以及其他检查项目，本文规定“医疗检验项目”主要针对人体而言，是能够通过设备或实验检测出的项目，并且是能够被量化，有其对应的测量值或指标值。

在标注“医学检验项目”实体时，需要注意:
（1）测量值不应该标注为“医学检验项目”。

① [输血]pro 指征 ： [[心率]ite ＞ 110 次 / 分]sym （“＞110次/分”不标注）

（2）上下文有关键的提示文字的检查：“检查”、“检”、“查”、“测定”、“检验”等，其对应的检查项目标注为“医学检验项目”。

① 对 [腹泻]sym 较 重 的 患儿 ， 应 及时 检查 [血pH]ite 、 [二氧化碳结合力]ite 、 [碳酸氢根]ite 、 [血钠]ite 、 [血钾]ite 、 [血氯]ite 及 [血渗透压]ite 。（3）在“临床表现”中，有时会涉及到对医学检验项目的描述，比如一些生理指标等，标注时应该将这些生理指标嵌套标注为“医学检验项目”。

① 患者 的 临床 症状 有所 好转 ， [[血清CK]ite 下降]sym

（4）常规的检查项目：尿常规、血常规等标本采集类项目，应该标注为“医疗检验项目”。

5.7 身体

“身体”身体泛指细胞、组织、及位于人体特定区域的由细小物质成分组合而成的结构、器官、系统、肢体，另外包括身体产生或解剖身体产生的物质等，包括身体部位和身体物质。

5.7.1 身体部位

身体部位包括器官或器官组成、身体系统、身体位置或区域。

① 常 见于 尿布 包裹 处 及 光滑 [皮肤]bod 相互 直接 摩擦 部位 ， 如 [肛周]bod 、 [臀部]bod 、 [外阴]bod 、 [腹股沟]bod 以及 [腋窝]bod 、 [下颏]bod 等 处 [皮肤]bod

5.7.2 身体物质

身体物质可由组织、细胞（包括细胞成分、细胞结构）、生物大分子、以及身体或解剖身体产生的物质。

在标注“身体”实体时，需要注意:
（1） 表 示 身 体 部 位 具 体 位 置 的 方 位 词 或 者 数 量 词 ，如“上”、“下”、“左”、“右”、“部”、“侧”、“双”、“多”等应当一同标注。

① 尤 以 [双下肺]bod 明显 ， 严重 病例 可 合并 [[胸腔]bod 积液]sym 或 [脓胸]sym 。

（2）当出现多个身体部位的组合，分开标注会失去原本含义，应作为整体标注为“身体”。

① [指、趾甲]bod

（3）病变细胞亦标注为“身体”

① [肿瘤细胞]bod 、 [白血病细胞]bod 、 [狼疮细胞]bod

5.8 科室

“科室”主要是指医院或医疗机构所设有的部门以及科室。

① [外科]dep [血液/肿瘤科]dep 和 [放疗科]dep 为 基本 组成 单位

5.9 微生物类

微生物类包括细菌、病毒、真菌以及一些小型的原生生物、显微藻类等在内的一大类生物群体。微生物类亦可分为致病微生物和非致病性微生物。

① 其他 [细菌]mic 也 可 产生 [肠毒素]mic ， 如 [耶尔森菌]mic 、 [鼠伤寒沙门菌]mic

在标注“微生物类”实体时，需要注意：

（1）出现“病毒+病毒的一部分（基因、DNA）”应该整体标注为“微生物类”。

① 此法 能 发现 不 完整 [病毒]mic 如 潜伏 [病毒DNA]mic

6 分类混淆处理

6.1 疾病(dis)和临床表现(sym)

疾病和临床表现的最大区别是：疾病是通过鉴别诊断的，疾病实质上就是身体受损；而临床表现实质上是身体受损后所表现出来的现象，比如说病人的不适感觉、身体出现的异常变化，但是这些往往是病人或者医生看到的表面现象。而作为医生则需要通过进一步的鉴别诊断来确认病人所患疾病，这也就说明疾病和临床表现存在着本质性的差异。

在遇到“感染”相关的实体时，有以下几点需要注意一下：

（1）若出现明确致病原因（病毒、细菌或身体部位等名称）与“感染”组合成词，则整体标注为“疾病”。如[HAV感染]dis、[球菌感染]dis、[上呼吸道感染]dis。

（2）当单独出现“感染”一词时，若上下文明显表示是对某种疾病的指代，则标注为疾病，否则标注为“临床表现”

（3）若“感染”一词前面的修饰词表明程度或频率时，则整体标注为“临床表现”

6.2 医疗程序(pro)和医疗设备(equ)

医疗程序是指检查过程以及预防或治疗过程，描述的是医生为诊断或者治疗而采取的一系列操作或过程。而医疗设备是指诊断或者治疗而使用的设备，描述的是具体的设备实体（工具、器具、仪器或机器），是医生进行诊断或者治疗的工具。标注者在标注时应该谨慎区分。

① 临床 发现 有些 [头颅]bod 较 大 的 婴儿 ， 行 [头颅CT]pro 和 [MRI检查]pro

6.3 药物(dru)和身体(bod)

标注实体是某种身体物质时，但是有“口服”、“注射”等字眼显式地表明该实体是一种药物（是外来的），此时应该将该类实体标注为“药物”。否则，如果只是表明该实体在人身体中的一种状态（是内在的），应该标注为“身体”。

① [糖尿病]dis 患儿 由于 [[胰岛素]bod 分泌 不足 或 缺如]sym

② 纠正 [高血钾]dis ： [葡萄糖]dru 0.5 g / kg 加 [胰岛素]dru 0.3 U / kg [静滴]pro

6.4 医学检验项目(ite)和医疗程序(pro)

医学检验项目是指体液检查项目、生理测量、重要生理指标以及其他检查项目，通常是名词。但如果医学检验项目名称后面紧跟着“检查”、“测定”、“诊断”、“分析”等，表明是医生为诊断或者治疗而采取的一系列操作或过程，故应将医学检验项目名称和这些词语作为整体一起作为标注为“医疗程序”。

① 监护 包括 [脉搏]ite 、 [血压]ite 、 [尿量]ite 、 [血乳酸含量测定]pro 和 [血气分析]pro

6.5 医学检验项目(ite)和身体(bod)

标注实体是某种身体物质时，但是在检查中涉及到对该实体的指标限定，显式地表明该类实体应该是某种检查项目，并且后面通常紧跟着测量值或者指标值，此时应该将该实体标注为“医学检查项目”。否则，如果只是表明该实体在人身体中的一种状态，则标注为“身体”。

① [输血]pro 指征 ： [[心率]ite ＞ 110 次 / 分]sym ； [[红细胞]ite ＜ 3 × 1012 / L]sym

7 医学实体语料标注

基于前文提出的医学实体体系和标注细则，本文制定了完整的医学实体标注规范。为检验规范的可行性、为医学实体识别提供语料支持，我们选择了约263万字的儿科类教材进行医学实体标注，选择儿科类的原因是儿科实质上是全科医学，医学知识涵盖范围广，具有代表性。

7.1 语料标注过程

医学实体标注规范的制定难度较大，不仅涉及专业的医疗知识,而且涉及到对医学实体的定义和分类。我们制定出初步规范,然后采用多轮迭代的模式进行规范的修订和标注工作。主要分为三个阶段来进行：

在第一阶段，组织标注人员学习本规范，组织标注人员预标注，目的在于熟悉医学实体标注规范，以及收集在实际标注医学文本中发生的问题。两轮预标注后，经过与医学专家讨论，进一步对标注规范进行完善，使标注规范更贴近本次研究任务，为正式标注打下坚实基础。

第二个阶段，在标注平台上利用第一阶段形成的医学实体资源库进行医学实体标注。标注过程采取多轮迭代模式，即每个医学文本由两名标注人员负责。一标者完成标注任务后，记录存在疑问的地方，接着由二标人负责检查并记录下不一致和不确定的地方。与医学专家商量讨论后获得统一的解决方案。讨论之后再由一标者负责修改标注，形成最后的三标文件。在这个阶段，会根据标注人员标注时的反馈意见修改标注规范，使标注规范更加适用于医学文本。

第三阶段进行分词的校对。开发可以修改分词的标注工具，标注人员在新的标注平台上修改分词错误以及检查实体标注情况是否符合目前已经更新的规范，同时查看是否存在实体缺失、错位等问题，提升标注质量。

7.2 医学实体分布统计

鉴于标注语料中同一类别的重复实体较多，我们从例数和型数两个方面对各个类别的医学实体数量进行统计，其中例数包含同一类别的重复实体，型数不包含同一类别的重复实体。

首先，我们对标注完成的所有医学实体数量进行了统计,见表2。根据统计显示，在这9种医学实体中，“临床表现”实体总数最多，“疾病”实体次之；“科室”实体总数最少。其次，除了“临床表现”实体外，其他实体均不含嵌套实体，关于嵌套实体统计如表3，其中“嵌套类临床表现”实体型数占“临床表现”实体型数约三分之一。

医学实体类型	例数	型数
疾病	28913	10494
临床表现	22989	14482
身体	27078	7223
医疗程序	11545	5095
医疗设备	1836	851
医学检验项目	4570	1935
药物	7549	2714
科室	574	112
微生物类	3863	1036
总计	109097	43942

Table 2: 语料实体类型与数量统计表

医学实体类型	例数	型数
临床表现	22989	14482
嵌套类临床表现	5375	4749

Table 3: 临床表现及嵌套实体统计表

7.3 医学实体自动标注实验

我们将整个标注数据按照8:1:1的比例随即划分为训练集、验证集和测试集，并统计了对应集合的实体数量分布，如表4所示，例数为各类中包含重复实体的总数，型数为各类中不包含重复实体的总数。我们在数据集上展开一系列的基线实验，在实验中，为体现整体效果，我们将嵌套类临床表现均视为整个临床表现实体来对待。基线实验如下：

CRF：CRF通过引入自定义的特征函数，不仅可以表达观测之间的依赖，还可表示当前观测与前后多个状态之间的复杂依赖。本模型使用：“前一个词，当前词，后一个词；前一个词+当前词，当前词+后一个词”作为特征。

BiLSTM-CRF：在训练过程中，LSTM能够根据识别实体自动提取观测序列的特征，但是缺点是无法学习到状态序列（输出的标注）之间的关系。CRF的优点就是能对隐含状态建模，学习状态序列的特点。所以在LSTM后面再加一层CRF，以获得两者的优点。

医学BERT-BiLSTM-CRF：使用经大量医学文本预训练好的BERT模型作为预训练模型，将BERT预训练的输出输入到一个双向的LSTM网络，在双向的LSTM网络上层再叠加一个CRF层，能够对标签信息加以利用，最终得到输出标签序列。

医学实体类型	Train		Dev		Test	
	例数	型数	例数	型数	例数	型数
疾病	23297	8970	2794	1705	2822	1740
临床表现	18544	11892	2241	1787	2204	1792
身体	21887	6182	2720	1353	2471	1213
医疗程序	9315	4313	1071	763	1159	778
医疗设备	1500	717	139	98	197	130
医学检验项目	3748	1674	435	262	567	342
药物	6216	2375	639	441	694	433
微生物类	3214	925	342	150	307	163
科室	459	94	80	34	35	13
总计	88180	37142	10461	6593	10456	6604

Table 4: 实验数据统计

7.3.1 自动标注实验结果分析

不同模型的结果如表5所示。标注的数据集包含9种医学实体，涵盖广、类别多；对于所有的医学实体而言，同时存在较长或较短文本。以上存在的两个问题，给识别任务增加了一定难度，故准确率会明显低于召回率。

就医学Bert-BiLSTM-CRF模型而言，表6为分类统计的结果，可以看出“疾病”、“药物”实体的识别能力较强。结合数据特点和结果，存在以下几个问题：部分医学实体存在“碰撞”问题，也就是说相同实体在不同情况下会具有不同实体类别，比如“疾病”和“临床表现”，这在一定程度上影响了模型的识别能力；“临床表现”实体普遍较长，模型的识别能力有待提高；“科室”实体较少，模型识别能力不强；在基线实验中，嵌套类临床表现均视为整个临床表现实体，并未识别内部嵌套的实体。这些问题对未来的工作提出了新的难题和挑战。

Model	P(%)	R(%)	F (%)
CRF	62.05	55.94	58.84
BiLSTM-CRF	63.56	56.58	59.87
医学Bert-BiLSTM-CRF	69.6	72.5	71.0

Table 5: 实验结果展示

医学实体类型	P(%)	R(%)	F (%)
疾病	79.4	82.3	80.8
临床表现	56.9	61.2	59.0
身体	68.1	71.5	69.8
医疗程序	66.8	70.1	68.5
医疗设备	70.7	71.9	71.3
药物	83.3	85.7	84.5
医学检验项目	61.5	59.5	60.5
科室	61.1	66.7	63.8
微生物类	76.4	71.3	73.8
总体	69.6	72.5	71.0

Table 6: 各类医学实体的具体结果

8 结束语

本文提出的面向医学文本信息处理的医学实体标注规范，详细介绍了各个医学实体的涵盖范畴，阐述实体间的混淆处理，用大量示例举例说明，适用于多种类型的医学文本。并且详细描述了医学语料标注任务的过程以及基线实验。基于本规范我们将发布一定规模的标注样例，希望能为医学文本信息处理提供最基础的数据资源。

致谢

本文工作得到国家重点研发项目(2018AAA0102003)，特此致谢。

参考文献

2010 i2b2/VA challenge. 2010. *https://www. i2b2.org/NLP/Relations/assets/Concept Annotation Guideline.pdf.*

Bodenreider O. 2004. Nucleic Acids Res. *The Unified Medical Language System (UMLS):integrating biomedical terminology*, 2004(32): 267-270.

Gao, Yao Gu, Lei Wang, Yefeng Wang, Yandong Yang, Feng. 2019. BMC Medical Informatics and Decision Making. *Constructing a Chinese electronic medical record corpus for named entity recognition on resident admit notes*, 19(56):67-78.

Lei J, Tang B, Lu X, Gao K, Jiang M, Xu H. 2014. J Am Med Inf Assoc. *A comprehensive study of named entity recognition in Chinese clinical text*, 21(5):808–14.

Roberts A, Gaizauskas R, Hepple M, Davis N. 2007. J Am Med Inf Assoc. *A comprehensive study of named entity recognition in Chinese clinical text*, 21(5):808–14.

South BR, Shen S, Jones M, Garvin J, Samore MH, Chapman WW, et al. 2009. BMC Bioinformatics. *Developing a manually annotated clinical document corpus to identify phenotypic information for inflammatory bowel disease*, 10(12):1-32.

Xu, Yan Wang, Yining Liu, Tianren Liu, Jiahua Fan, Yubo Qian, Yi Tsujii, Jun'ichi Chang, Eric. 2013. Journal of the American Medical Informatics Association. *Joint segmentation and named entity recognition using dual decomposition in Chinese discharge summaries*, 2013(21):84-92.

杨锦锋, 关毅, 何彬, 曲春燕, 于秋滨, 刘雅欣, 赵永杰. 2016. 软件学报. 中文电子病历命名实体和实体关系语料库构建, 27(11):27252746.

YANG Jin-Feng, GUAN Yi, HE Bin, QU Chun-Yan, YU Qiu-Bin, LIU Ya-Xin, ZHAO Yong-Jie. 2016. Journal of Software. *Corpus Construction for Named Entities and Entity Relations on Chinese Electronic Medical Records*, 27(11):2725-2746.

俞士汶, 段慧明, 朱学锋, 孙斌. 2002. 中文信息学报. 北京大学现代汉语语料库基本加工规范, 16(5):51-66.

汉语块依存语法与树库构建

钱青青
北京语言大学/ 北京海淀学院路15号
qianqingqing1996l@foxmail.com

王诚文
北京语言大学/ 北京海淀学院路15号
chengwen_wang15@126.com

摘要

本研究依据以谓词为核心的块依存语法构建块依存树库，在句内和句间寻找谓词所支配的组块，利用汉语中组块和组块间的依存关系补全缺省部分，明确谓词支配关系。目前共标注2199篇文本，涵盖百科、新闻两个领域，共约187万字语料。本文简述了块依存语法的原则，并对组块及其依存关系进行了定义。将详细介绍标注流程、标注一致率、数据分布等情况。基于现有的树库，本研究发现汉语中有约25%的小句是非自足的，约有88%的核心谓词可支配1~3个从属成分。

关键词： 组块 ； 块依存语法 ； 树库

Chinese Chunk-Based Dependency Grammar and Treebank construction

Qian Qingqing
BLCU / 15th Xueyuan Road, Beijing
qianqingqing1996l@foxmail.com

Wang Chengwen
BLCU / 15th Xueyuan Road, Beijing
chengwen_wang15@126.com

Abstract

This study create a treebank according to Chunk-Based Dependency Grammar ,which take predicate as the core and chunk as the object. With this grammar, predicate-dominated chunks can be found within and between sentences, default parts of sentences can be completed by the relations between chunks, and the dominant relations can also be clarifies .We have currently labeled 2199 texts, covering encyclopedia and news based on the Chunk-Based Dependency Grammar system, totaling about 1870,000 words.This paper briefly describes the principles of Chunk-Based Dependency Grammar and defines the chunks and relations.The labeling process, labeling consistency, data distribution, and so on are described in detail.Based on the existing tree bank, this study found that about 25% of clauses in Chinese are not self-sufficient, and about 88% of core predicates can control 1-3 subordinate components.

Keywords: chunk , Chunk-Based Dependency Grammar , treebank

1 引言

依存句法是自然语言处理领域的热门研究也是基础研究，其目的是将输入句子从序列形式变为依存树状结构，通过判断句内的词语之间是否存在依存关系以及存在何种依存关系，能够

适应灵活的语序特征，将句子分析为更加扁平的结构，从而降低了分析、标注、储存的难度，在问答系统、知识图谱、信息抽取等任务上发挥着重要作用。

值得注意的是，传统依存句法分析大多以词作为最小单元，在汉语中的应用也存在不适应的地方。汉语实际语篇中，词的词性、词义较为灵活，存在大量的活用、增加语境义等现象，传统依存句法分析以词作为分析节点的处理方式较难适应该特性；汉语具有典型意合特征，同样的语义内容可由语序不同的语言单元表达，关注其中的“词-词”关系，使句子依存结构更为繁琐；词与词之间的关系复杂、多变，依存关系类划分的太细，降低了标注的可操作性，带来数据稀疏问题，也会因此影响到分析器的适应面和鲁棒性。此外，一些传统句法分析难以解决的问题在依存句法分析中也存在。句法分析一般以标点作为边界，而汉语中多流水句，主语、宾语、状语等的省略现象层出不穷，为分析结果的实际应用带来了困难。

为了解决以上问题，本研究提出了块依存语法，以组块为研究对象，以谓词为核心，在句内和句间寻找谓词所支配的组块，利用汉语中的组块和组块间的依存关系，既能够适应汉语灵活的语序特征[1]，又能够将小句间成分缺省的问题转化为句间组块缺省成分补全的问题。同时以谓词为核心进行块依存关系构建能清晰呈现出句子的骨干结构，为后续任务提供准确的分析单元，关于块依存理论的详细说明请见另文讨论。

基于块依存理论，本研究对汉语组块理论、依存树库构建进行深入研究，以数据标注规范作为指导，以两两对比标注的模式，在基于浏览器的在线标注系统中，标注百科文本、新闻文本，构建了汉语块依存树库。

2 相关研究

在传统的句法分析中，首先对句子进行分词和词性标注，再进行后续的句法语义分析工作。分词和词性标注的错误会带来较大的错误级联问题。与此同时，汉语有许多形式和语义上比较凝固的块成分，尤其是一些构式性成分，整体表示一定的语义，反而不适宜进行分词和词性标注基础上的句法语义分析。

组块分析理论由Abney在20世纪90年代初提出，CoNLL2000会议将组块分析作为Share Task提出(Erik & Buchholz 2000)，使该理论得到推广应用。国内学者也开展了大量的块研究工作。其中，刘芳、赵铁军等（2000）将块(Chunk)定义为一种包含一层或二层的符合一定句法功能和反映组成意义的短语结构，并将其分为八种类型；周强（2001,2007）从功能的角度对汉语中的语块进行了研究，定义了主语语块、述语语块、宾语语块、兼语语块、状语语块、补语语块、独立语块、语气块8类语块，形成了一套基于拓扑结构的汉语语块描述体系；其后，陈亿、周强等人（2008）设计了多层次功能块分析体系，进一步分析长功能块的内部结构；李素建（2002）将组块定义为符合一定句法功能的非递归短语，在划分组块时遵循非递归、无重叠、全覆盖的原则。

在依存句法树库方面，哈工大的汉语依存结构句法树库发表于2012年，以句法关系为主，语义信息知识作为补充，标注了人民日报约111万词的汉语语料。北大汉语依存结构句法树库发表于2015年，以依存句法为核心，并形成多种视图的标注体系，标注了新闻、专利及医药等约140万词的汉语语料。苏州大学面向多领域多来源文本构建了 3 万句左右的汉语依存句法树库。

在将组块理论与依存分析结合方面，Zhou（2000）较早地提出了一种基于块的依存分析器，分析块之间的依存关系，在非限制性的中文文本翻译中取得了较好的效果。但闻媛等人（2018）也指出由于中文中的模态词提升、话题化、成分分离等，在中文中存在较多的非投影结构，遵循依存语法的四条准则，为分析中文也带来了一定的难度。

此外，为解决汉语中多缺省的现象，宋柔（2013）归纳了广义话题结构遵从的堆栈模型和拓展后的流水模型，并将汉语的句子大致界定为自足的广义话题结构，把小句界定为基于广义话题结构的话题自足句（宋柔, 2017），利用流水模型生成这两类汉语篇章结构单位，为自然语言处理篇章分析单位提出了新的角度，从汉语篇章微观话题结构的角度为流水句提供了佐证和启示。但汉语中标点句并非只缺省句首的话题成分或主语，大量句中或句尾的宾语、补语等的缺省也值得关注；按照广义话题结构所生成的句子仅仅提示其话题-说明结构，与句子更深层次的结构分析之间缺少衔接，大多还是停留在拆分复杂结构，生成“能说”的自足句层面。苏州

[1]汉语语序灵活，但组块内成分具有相对稳定性

大学的多领域文本依存句法树库中也设置了表示谓语之间共同主语的依存关系，但并未全面地有针对性地解决缺省的问题。

3 汉语块依存树与组块关系

相对于细粒度的词来说，组块内部的句法、语义结构更加稳定，更符合语言的认知规律，是一种整存整取的单位。以组块为研究对象，能够避免纠结于“词-词”之间的依存关系，更关注于句子的整体结构，进一步降低存储和分析的复杂性，也能够达到减少分词碎片、加强鲁棒性的目的，因此本文的依存关系构建以“组块”为最小单元。

本文将组块定义为：由连续词语或语素整合而成的序列，表现为同一句子层级中充当句法成分的各个连续单元。在句法结构层面的组块按照功能可分为谓词块、主语块、宾语块、状语块、补语块，其中主语块和宾语块按照其性质可继续下分为谓词性主语块、谓词性宾语块、体词性主语块和体词性宾语块；除此之外，组块还包括篇章层面的衔接组块和辅助组块。块依存语法主要分析非篇章成分的组块，即基于句法结构层面的6类组块。

我们认为核心谓词组块是句子的核心，各类短组块均受核心谓词组块的支配并依存于核心谓词组块之上，在块依存关系分析中以谓词块作为句子的核心，寻找谓词所支配的各类组块。若某短语块和核心谓词组块之间存在依存关系，则称该短语块为核心谓词组块的从属成分，核心谓词组块为该短语块的依存对象。除了一些特殊的独词句，一般认为句子中都存在一个或多个核心，短语块至少依存于一个核心谓词组块之上。

核心谓词组块作为句内各组块的依存对象，其左右上下各有四个点位，分别表示其主语位（1号位）、修饰语位（2号位）、宾语位（3号位）、述语位（4号位）。各非谓词块按照其类别分别依存于谓词组块的四个节点上，依存线条从谓词组块的四个节点指向其从属成分。

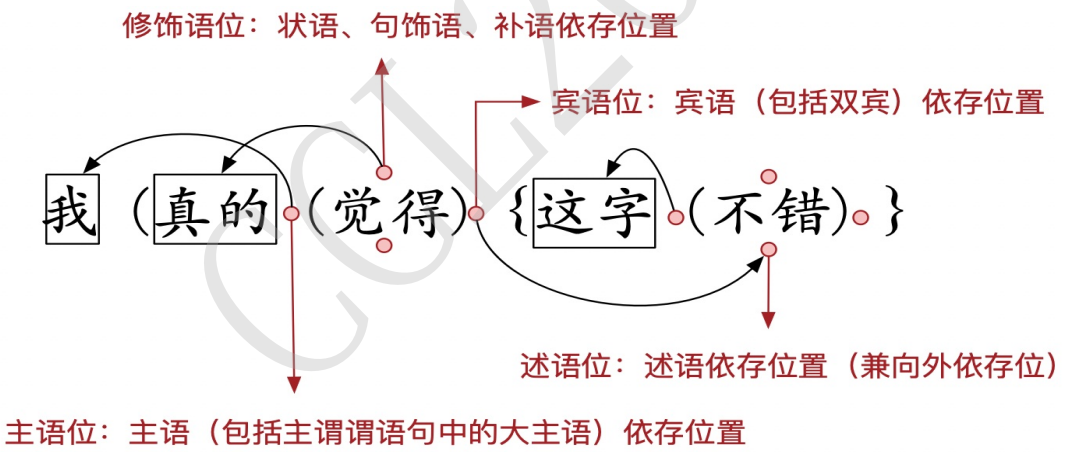


图 1: 块依存标注图示

主语，包括主谓谓语句中的大小主语依存于谓词组块的1号位；在后续分析中，我们将谓词块与1号位上的体词性成分之间的关系定义为NP-SBJ，与谓词性成分之间关系定义为VP-SBJ。

状语、补语依存于谓词组块的2号位；在后续分析中，我们将谓词块与2号位上的成分之间的关系定义为NULL-MOD。

宾语，包括双宾语中的远近宾语依存于谓词组块的3号位；在后续分析中，我们将谓词块与3号位上的体词性成分之间的关系定义为NP-OBJ，与谓词性成分之间关系定义为VP-OBJ。

谓词性关联块与核心述语的4号位置连接，当某谓词组块依存于其他谓词组块时从4号位向外依存。在后续分析中，我们将谓词块与4号位上的成分之间的关系定义为VP-EMP。

因此，我们可以将谓词与其依存块之间的关系初步区分为以下6种：

标识	关系说明	举例
NP-SBJ	表示该谓词的体词性主语块	**我**说了一件事儿
VP-SBJ	表示该谓词的谓词性主语块	**学习**对我来说总是快乐的
NP-OBJ	表示该谓词的体词性宾语块	吃**苹果**
VP-OBJ	表示该谓词的谓词性宾语块	进行**调查**、**研究**
NULL-MOD	表示该谓词的修饰语块	**上周**她出门了。 跑**得很快**。
VP-EMP	表示该谓词的谓词性关联块，即与谓词块相同的空述语。例句中的空述语“()”是“有”的谓词性关联块	有书三本，()笔三支。

表 1: 块依存标签说明

以谓词为抓手使得分析更具有灵活性，经过块依存语法分析的句子，能够展现为块依存图的形式。整个句子以空节点为根，指向句中的核心谓词，核心谓词又有各个线条指向其支配成分。在句间关系分析中，无论是寻找句间关系还是直接分析谓词间关系，都能够以更准确的分析单元为着力点。

在进行块依存分析时，不限于小句或句子内部，而是补全因上下文而缺省的单元，能够将句子还原为更完整的形式，也为后续分析提供更完整的单元。例1中包含两个句子，其中第二个句子的两个核心谓词“苦”“蕴含”的主语是缺失的。通过观察上下文，我们可以找到对应主语应为前一句子主语修饰语“哥伦比亚咖啡”。相同地，例2中第二句缺失的状语“这些年”也可通过相似的方法在上下文中找回。

1) 哥伦比亚咖啡的风味是丰富多样的。不仅不苦，还蕴含着水果、坚果、谷物等不同气息。

2) 这些年，他通过努力进步了不少，是我们学习的榜样。而很多人却没有珍惜时间，仍在原地踏步。

3) 吕先生和许多严肃的学者一样，不会随便去别人家串门，把宝贵的时间都浪费在无聊的事情上。

块依存方法能够在补全缺省成分的同时明确句中成分的指向、句子的结构。在例3中，话头为“吕先生和许多严肃的学者一样，不会”，其中既有体词性成分，也有修饰性成分，修饰性成分中存在框式结构。利用块依存方法进行补全更具有理据性——话头部分能够成为另一个小句的一部分是因为它内部的两个组块都受到其中核心谓词的支配。

4 块依存树库建设

本文所标注的语料，均来自于“基于篇章的汉语句法结构树库”（下文称句法结构树库）。该树库目前已人工构建约1000 万字符集规模，包括1 万语篇文本、26.5 万单复句、64.4 万小句；以新浪及新华社新闻、百度百科、专利申请书、小学生作文、法律案件判决书等应用性文本为标注语料；树库中人工标注数据一致性校验Kappa 值均大于0.8。

在句子成分标注中已经明确了句、小句、组块的边界，并运用标注符号标识组块功能，本文所进行的标注是在句法结构标注基础上所进行的块依存标注。句法结构树库将句子成分分为主语、宾语、述语、句饰语、衔接语、辅助语，并分别以不同符号进行标注，如例4、5所示，“<>”表示衔接语，“<<>>”表示辅助语，“[]表示句饰语（即与述语分离的状语或补语），“()”表示述语，述语内部又可利用“()”区分出状语、核心述语、补语，“{}”表示谓词性的主语或者宾语，体词性主语或宾语则无需用标注符号标注。

4) <但是>，[到今天为止]我(还是(放心)不下)你<<啊>>。

5) {(大力(发展))经济}(是)我们目前的工作重心。

在句号、分号、叹号等8个标句点号的基础上对篇章进行划分，通过语篇句子成分标注对句

子边界进行校准，能够明确句、小句的边界。当原本应属同一小句的主宾语、定状补语等向核成分被标点切分开的时候，使用标注符号将标点括在内部，达到取消标句点号的分句功能的目的：

6) [抗日战争胜利前夕，]党中央和毛主席(发出)号召和命令。

7) 他(是)河北辛集马兰村的一名普通农民。

8) “独自飞行1840.018公里的北京山茗网络科技有限公司创始人”\\，(是)彭少仪名片上的唯一头衔。

9) 我(吃)饭| 他(睡觉)。

10) [五个多月来，]中国海监船编队(一直(坚守)在祖国的钓鱼岛海域)(巡航)，他们(克服)重重困难，(涌现了)无数可歌可泣的先进事迹，他们“特别能吃苦、特别能战斗、特别能奉献”的“海监精神”，(也极大地(鼓舞了))全国人民。

例句6~10均为一个“句”，也是一个小句单元，而例9~10是通过分析校正后确定的句子，分别以两个和五个小句构成。

为了提高标注效率和质量，本研究实施在线标注，以下为标注界面：

图 2: 标注页面示例

本研究利用kappa值计算加权的一致率，计算公式为：

$$Similarity = \sum_{i=1}^{n} k_i p_i \tag{1}$$

其中k为该依存关系的kappa一致率，p为该依存关系在全篇任务总依存关系数量中的占比。一般认为若kappa值在0.8以上，则表明二者的一致率较高，本树库为保证高水平的一致率，将阈值定为0.9。数据标注一致率控制流程如图4所示，在早期的标注任务中，采用双人标注的模式，只有当基于kappa计算的一致率不低于0.9时，标注任务才算通过，若一致率低于0.9，则需要通过讨论修改、专家介入等方式提高一致率。质量比对通过后选取当期标注任务中平均kappa值较高人员文本入库。

以上述策略开展标注实践，截止2020年8月，共标注2199篇任务，涵盖百科、新闻两个领域；共计约187万字语料，包含超过4万个句子，10万多小句，最终平均kappa值为0.945。

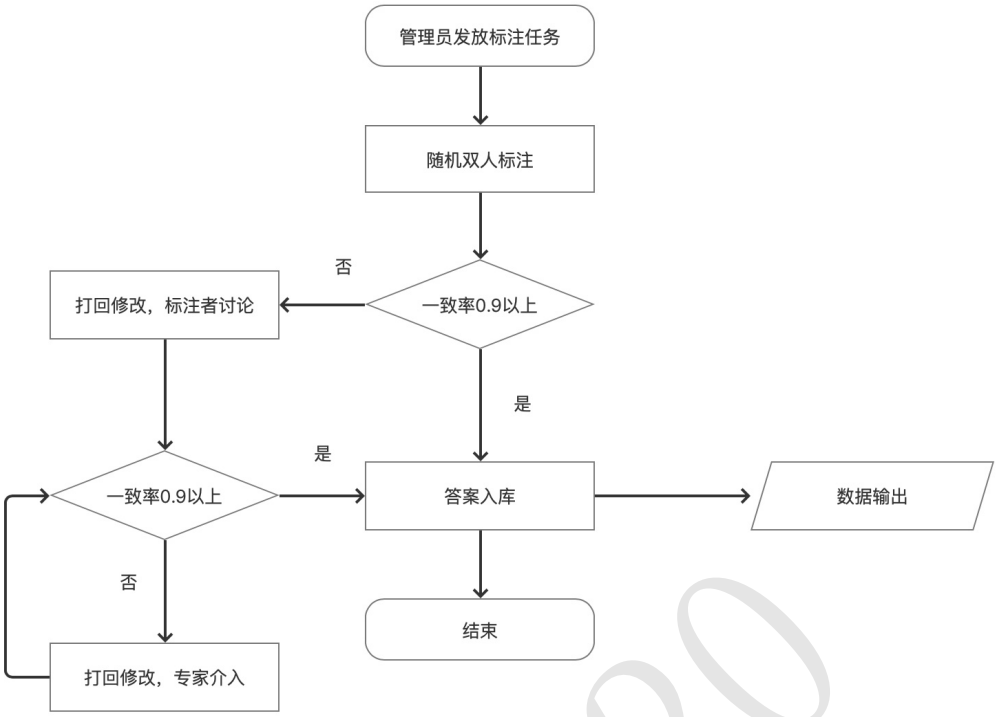


图 3: 数据标注流程

5 汉语组块关系

5.1 自足句与非自足句

下面以树库中的708个任务作数据分析，内含554521字，14181个复句，30724个小句。

此处我们定义两类句子，即“自足句”和“非自足句”。自足句指句中不缺省成分的句子，句中核心谓词的所有依存块均位于该句内部。从此定义出发，可知有一类较为特殊的自足句是独词句或仅包含篇章成分的句子，例如“哗啦哗啦”“因为”等，在后续分析中会将此类独词句独立分析；此外，某些无法补全缺省成分的句子如“(下)雨<<了>>”，也认定为自足句。“非自足句”则指句中缺省成分，句中存在核心谓词有位于其他句的依存块。相应的，可以将小句和句子分为“自足小句”“非自足小句”“自足句”“非自足句”。

11) 国家发改委(联合)相关单位(连续(出台))中长期发展规划和场地建设规划，

12) (督促)地方政府(加快(制定))实施细则；

13) 督查涉及部门(很(多))，内容(非常(具体))<和>(细化)，<应该说>(是)一次无缝的、立体的和全方位的督查。

14) ()研究空气和燃气与发动机各零部件相对运动及其相互作用的学科，(是)流体力学的一个分支。

在例11中，谓词“联合”的主、宾语块分别为“国家发改委”、“相关单位”，谓词“出台”的主语块为“国家发改委”、“相关单位”，宾语块为“中长期发展规划和场地建设规划”，状语块为“连续”。例11内部所有谓词块的从属成分均位于该小句内部，因此我们称这样小句为“自足小句”。

例12中，“制定”的各个从属成分均位于该小句内部，但“督促”的主语块位于其他小句中，在这里应该是“国家发改委”，那么我们称某些小句内部存在谓词块的从属性成分在句外的是“非自足小句”，需要通过块依存标注来补全结构。

例13是一个复句，内部的几个核心谓词“多”、“具体”、“细化”、“是”的各个从属成分均位于该复句内部，虽然“是”的主语块是跨了一个小句的大主语“督查”，但我们依然认为在复句层面，这个句子是自足的。当然，割裂来看，在小句层面，第三个小句是不自足的，经过分析并补全之后，该句可以形成三个内部自足的小句：“①督查{涉及部门(很(多))}，②内容(非常(具体))<和>(细化)，③<应该说>督查(是)一次无缝的、立体的和全方位的督查。”。

与之相对应的是例14，其中包含的一个空述语和谓词“是”在该复句内部均没有相对应的主语，我们需要在上文中找出“某某学科”作为它们的主语。因此我们认为这一类在内部存在无法找到所有从属成分的句子是“非自足”的。

分析数据中的小句和句子中，自足和非自足的分布结果如下图所示：

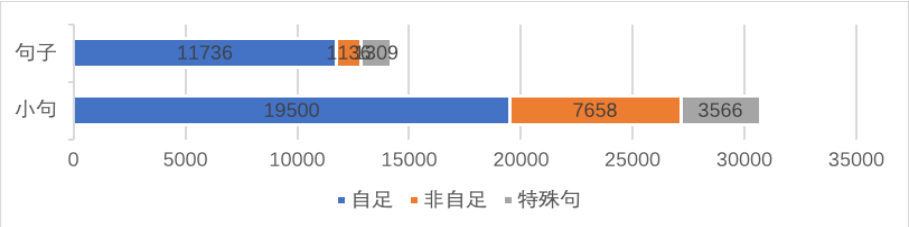


图 4: 句子/小句内自足与非自足分布图

通过统计，发现其中有23066个小句和13045个句子是自足的，分别占总数的75%和92%。若排除上文提到的特殊句，则分别占63%和83%。另外有7658个小句和1136个复句是非自足的，分别占总数的25%和8%。

上述结果在经过结构标注的句边界校准之后得到，由此可知的是：在汉语中，成分缺省是普遍存在的事实。若简单以标点符号分割后的文本来作为分析单元，则将会有25%以上的小句存在内部成分缺省，会造成指代、时间、地点等情态信息不明等问题。而本研究通过块依存的方式，补全了缺省的小句和句子，使这25%的句子变为“自足”的句子，能够极大程度地填补句子缺省信息，便于后续基于自足句子的分析。

5.2 谓词及其依存块

基于已有的树库，分别从核心谓词及和核心谓词支配块数量等角度做了统计。在标注数据中共包含5479项核心谓词，其中有4676项动词存在含有体词性主语块的实例，4089项动词存在含有修饰语块的实例，2943项动词存在含有体词性宾语块的实例，477项动词存在含有谓词性宾语块的实例，229项动词存在含有谓词性主语块的实例，6项动词存在含有谓词性关联块的实例，还有340项动词存在没有任何从属成分的实例。

标识	谓词项	占谓词总比	谓词实例数	依存块数	平均可支配组块数
NP-SBJ	4676	0.853	25744	26673	1.036
VP-SBJ	229	0.041	477	617	1.2935
NP-OBJ	2943	0.537	20786	21093	1.0148
VP-OBJ	477	0.087	2873	3529	1.2283
NULL-MOD	4089	0.746	19351	27278	1.4096
VP-EMP	6	0.001	7	7	1
NoDep	340	0.062	494	0	0

表 2: 各类依存块依存情况统计

如图5所示，在30487个谓词实例中，约有88%支配1~3个从属成分。值得注意的是，从属成分数量在2~3个的谓词数量均多于从属成分为1个的谓词，这表明多从属成分的谓词在汉语中是更加普遍存在的，谓词能够支配的句法成分不单一。从属成分最高为10个，为例15中的“是”，它的从属成分包含1个体词性主语块、7个体词性宾语块和2个谓词性宾语块。在从属成分数量大于6的31个实例中（图7），绝大多数的谓词并非实义动词，“是”“有”“包含”“分为”“进行”等含义抽象的系动词或是形式动词等，且这些动词大多能够支配多个主语块或宾语块；而部分实义动词，如“列入”“拓展”等，更多的是支配了多个修饰语块；对于一些具有认知、言说义的的动词，如“要求”、“意味”等，则相对可能性更多些，它们存在支配多个宾语块的能力，同

时也有较为明确的意义，也能够支配较多的修饰语块。在依存块方面，不同于宾语块，体词性主语块和谓词性主语块的差别较为明显，体词性主语块的数量多维持在一个较为稳定的水平，一般不超过2个，而谓词性主语块的数量则起伏较大，一般无谓词性主语块但也存在谓词性主语块数量大于5的情况。

在340项没有任何从属成分的动词中，有58个动词（约17%）为古诗词或熟语，如“谋长远计”“智者善谋”“功夫不负有心人”等。在结构标注中，为了保持其内部语义的完整性，并未将其切分为多个部分，也因此使其内部一般不缺省成分，其他则多是在上下文语境中难以补全其缺省成分的，如例15中的“学习”。

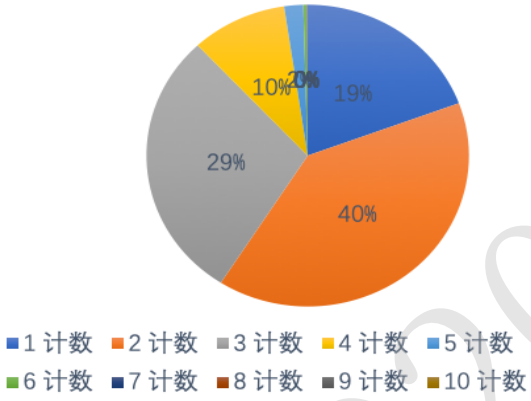


图 5: 谓词块从属成分统计

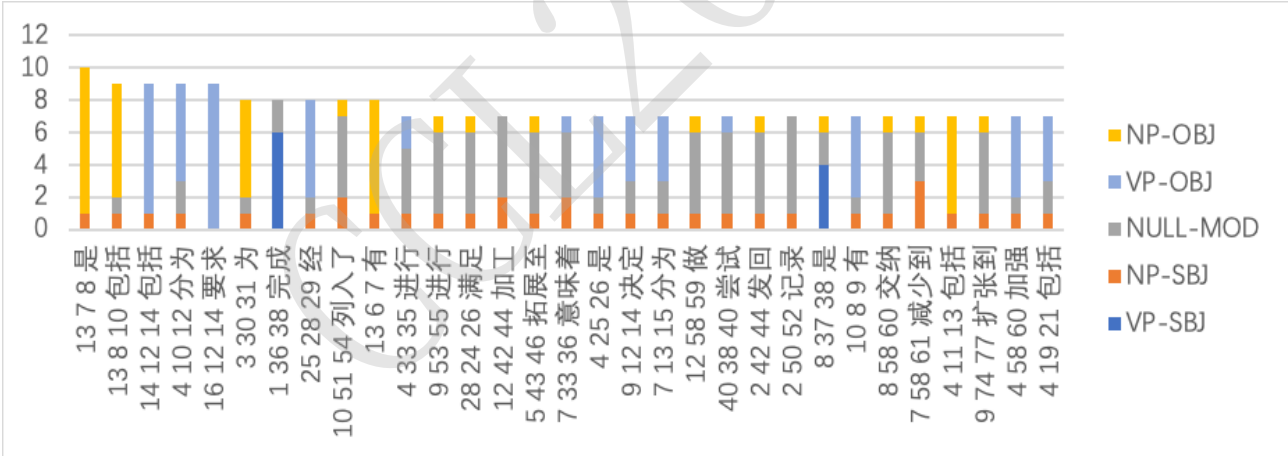


图 6: 从属成分数量最多谓词的从属块类别统计

17) {(学习)、(看)书}(是)让人很愉快的事情。

图 7: 块依存标注结果

从依存块的角度进行分析，该语料中共包含了79197个语块。其中体词性主宾语、修饰语占了绝大部分，均约占30%。与表2结合，我们可以发现，体词性主语和修饰语的分布较为平均，语料中绝大多数的谓词均能够支配主语修饰语组块，仅有约一半的谓词项能够支配宾语

块，这与不及物动词的存在有着密切的关联。而谓词性主语、谓词性宾语则都少于0.1，最少的是VP-EMP组块，仅占0.001，表明该类关系在语言中较为特殊。从可支配组块数量看，每个谓词实例平均可支配体词性主语块为1.04个，体词性宾语块也在1.01左右，谓词性的主、宾语块数量略高，约1.2左右，而修饰语组块最高，为1.41个，表明在实际语料中，往往有较多的修饰语修饰谓词，能够表达较丰富的情态、时态等信息，若简单地线性分析，则会丢失大量的信息。

6 总结

本文介绍了块依存树库建设的相关工作，截至2020年8月为止，本树库共标注了约187万字的高质量语料数据，包含超过4万个句子，10万多小句。本树库主要面向的是汉语中多流水句、缺省普遍的现象，通过依存标注的方法，以谓词为核心，从上下文中寻找支配的组块，从而补全缺省部分、明确句内支配与被支配的关系，也为接下来的语义分析奠定坚实的基础。基于此树库，我们发现汉语中有约25%的小句、8%的复句内部存在缺省现象；汉语中体词性主宾语块、修饰语块占绝大部分，约有90%，谓词块支配体词性主语、修饰语的能力更强，支配多组块的可能性更高。

基于块依存的树库建设充分遵循汉语语序灵活但块内顺序相对稳固的特点，同时将分析单元上升到块，可以有效抓住句子的骨架进行分析，避免了“词-词”依存分析所导致的句子依存结构繁琐问题。另一方面，谓词为中心的依存表征体系也为语义分析提供了结构支撑。通过树库的建设验证了组块依存分析理论的可行性，同时该数据资源能够为自然语言处理发展提供资源支撑。

在接下来的工作中，我们会进一步扩大标注，增加树库的规模和覆盖范围，增加不同领域的文本；探索各类关系内部依据特点的细分，例如修饰块内部可按照其语义关系细分为时间、地点、情态、数量等等。

参考文献

Steven P. Abney. 1991. *Parsing By Chunks. Principle-Based Parsing.*,257-278. Springer Netherlands.

Zhou Ming. 2000. *A Block-Based Robust Dependency Parser for Unrestricted Chinese Text.* The second Chinese Language processing workshop attached to ACL 2000, HongKong.

陈亿,周强,宇航. 分层次的汉语功能块描述库构建分析[J]. 中文信息学报.2008(03):24-31+43.

郭丽娟,彭雪,李正华,张民. 面向多领域多来源文本的汉语依存句法树库构建[J]. 中文信息学报.2019. 38-46.

郭丽娟,李正华,彭雪,张民. 适应多领域多来源文本的汉语依存句法数据标注规范[J]. 中文信息学报.2018. 32-39+56.

李素建. 汉语组块计算的若干研究[D]. 中国科学院研究生院（计算技术研究所, 2002.

宋柔. 汉语篇章广义话题结构的流水模型[J]. 中国语文, 2013(06):483-494+575.

宋柔,葛诗利,尚英,卢达威. 面向文本信息处理的汉语句子和小句[J]. 中文信息学报, 2017,31(02):18-24+35.

闻媛,宋丽,吴泰中,李斌,周俊生,曲维光. 基于中文*AMR*语料库的非投影结构研究[J]. 中文信息学报, 2018,32(12):31-40.

周明,黄昌宁. 面向语料库标注的汉语依存体系的探讨[J]. 中文信息学报, 1994(03):35-52.

周强. 汉语句法树库标注体系[J]. 中文信息学报, 2004, 18(4):2-9.

周强. 汉语基本块描述体系[J]. 中文信息学报,2007(03):21-27.

周强,孙茂松,黄昌宁. 汉语句子的组块分析体系[J]. 计算机学报,1999(11):1158-1165.

汉语学习者依存句法树库构建

师佳璐， 罗昕宇， 杨麟儿， 肖丹， 胡正升，
王一君， 袁佳欣， 余婧思， 杨尔弘
北京语言大学，北京 100083

摘要

汉语学习者依存句法树库为非母语者语料提供依存句法分析，可以支持第二语言教学与研究，也对面向第二语言的句法分析、语法改错等相关研究具有重要意义。然而，现有的汉语学习者依存句法树库数量较少，且在标注方面仍存在一些问题。为此，本文改进依存句法标注规范，搭建在线标注平台，并开展汉语学习者依存句法标注。本文重点介绍了数据选取、标注流程等问题，并对标注结果进行质量分析，探索二语偏误对标注质量与句法分析的影响。

关键词： 汉语学习者 ； 依存句法树库 ； 语料标注 ； 偏误分析 ； 依存句法分析

Construction of a Treebank of Learner Chinese

SHI Jialu, LUO Xinyu, YANG Liner, XIAO Dan, HU Zhengsheng,
WANG Yijun, YUAN Jiaxin, YU Jingsi, YANG Erhong
Beijing Language and Culture University, Beijing 100083, China

Abstract

A dependency treebank of learner Chinese provides parse trees of non-native sentences, which could promote the teaching and researching of Chinese as a second language, as well as support related researches such as syntactic analysis of learner language and grammatical error correction. However, few treebank of learner Chinese has to be seen, and there are still some problems in annotation guideline. So far, we improve the annotation guideline, develop an online annotation platform, and build the Treebank of Learner Chinese. This thesis describes the details in data selection and annotation workflow, evaluates the quality of annotation, and explores the impact of errors on annotation quality and syntactic analysis.

Keywords: Chinese learners , dependency treebank , data annotation , error analysis , dependency analysis

1 引言

基金项目：北京语言大学语言资源高精尖创新中心项目(TYZ19005)；国家语委信息化项目(ZDI135-105)；北京语言大学研究生创新基金（中央高校基本科研业务费专项资金）项目成果(20YCX141)

树库作为一种记录每个句子句法分析结果的标注语料库(黄昌宁and 靳光瑾, 2013)，融合了分词、词性、句法等各种信息。一方面能为语言学、句法学研究提供实例，另一方面也能为句法分析、机器翻译、语法改错等自然语言处理领域的相关任务提供训练数据。与其他类型树库相比，依存句法树库具有以下优点：1）存储空间小，便于大规模储存；2）依存句法的形式简洁，易于理解(刘挺and 马金山, 2009)，更加适合人工标注；3）更加突出中心词的地位，侧重于反映语义关系，有助于语义角色标注、信息抽取等上层应用(刘挺and 马金山, 2009)；4）依存句法在表示交叉关系时更有优势(刘挺and 马金山, 2009)，特别适合于分析语序灵活的汉语(郭丽娟, 2019)。

汉语学习者语料是伴随着汉语国际教育产生的。随着汉语学习在全球的不断开展，语料的规模也不断增长，所构建的汉语学习者语料库也越来越多。汉语学习者语料与一般语料相比有其独特性，包含着大量的偏误，即中介语与目的语规律之间的差距，只有学习外语的人才会产生(鲁健骥, 1984)。正是由于这些语料在语言使用上的独特性，使得汉语学习者语料成为语言信息处理和智能语言辅助学习领域的独特资源。目前的汉语学习者语料库在“字”、“词”的偏误标注上较为深入，但对句法结构的关注度不够(李娟et al., 2016)。

香港城市大学构建了汉语学习者依存句法树库[0]（UD_Chinese-CFL）。它在一定程度上弥补了现有汉语学习者语料库的不足，但是该树库对汉语特殊词性和句式考虑不够周全，标注标签种类过多，且未充分考虑学习者语料中的偏误对标注原则和标注结果的影响。

鉴于此，肖丹et al. (2019)制定了面向汉语中介语的依存句法标注规范，并考虑到了汉语特殊词性、句法结构和汉语中介语的特性等问题，然而该标注规范在标注原则和依存关系标签上仍需要进一步完善。本文对该标注规范进行了改进，使之更符合汉语及汉语学习者语料的特点，并搭建了在线标注平台，对含有偏误句（汉语学习者原始语料）和目标句（纠偏后的句子）的平行句对进行标注，初步构建了汉语学习者依存句法树库，并以此探讨偏误对依存句法的影响。

2 相关研究工作

学习者语料库是第二语言或外语学习者产生的语言数据的电子文本库(Granger, 2012)。它记录的是非母语学习者在使用目的语的过程中产生的语言。这种语言既不同于母语，也不同于目的语，并且带有语法偏误信息。构建带有显性句法信息的学习者语料库对自然语言处理领域具有重要意义，能够为语法改错、句法复杂性研究等任务提供帮助。目前，国外学习者依存句法树库的建设工作已逐渐走向成熟，而相比之下国内汉语学习者依存句法树库建设的相关研究尚处于起步阶段，进展相对缓慢。

英语学习者依存句法树库发展迅速，最具规模。英语学习者句法标注项目（The Project on Syntactically Annotating Learner Language of English，以下简称SALLE)(Ragheb and Dickinson, 2014) 和英语学习者树库（The Treebank of Learner English，以下简称TLE）(Berzak et al., 2016)是英语上影响力最大的两个学习者树库。

SALLE是学习者树库的先驱，由Ragheb和Dickinson等人于2014年构建。它在SUSANNE Corpus(Sampson, 2011)的词性标签集和儿童语言数据交流系统（Child Language Data Exchange System）(MacWhinney, 2014)的依存标签集基础上构建标注规范，对大学生的英语作文语料进行标注。SALLE关注到了句子的表层结构，对推进句法标注在学习者语料库中的发展具有重要意义。但由于SALLE只对学习者语料表现出的语言学特征进行标注，而没有进行偏误标注和修改，因而难以应用到语法错误识别、语法改错等任务中。

TLE是2016年由Berzak等人构建的英语学习者树库。与SALLE相比，它有两大特征：1）在国际通用依存标注体系（Universal Dependencies，以下简称UD）[1] (de Marneffe et al., 2014; Nivre et al., 2016)之下建立标注规范，对多语言对比分析具有极大意义。2）对语法错误进行标注，便于应用到自然语言处理领域之中。UD是目前拥有语言种类最多的通用依存标注体系，它为所有语言提供统一的标注方案，来解决句法分析器在跨语言分析上效果不佳的问题(Nivre et al., 2016)。截止目前，最新版本的UD V2.6已发布了92种语言的标注数据，共163个树库。TLE结合英语学习者的语言特征，对英语母语者语料标注规范进行一定修订，形成了基于UD的英语学习者语料标注规范。TLE的语料来源于剑桥学习者语料库（Cambridge First

[0]UD_Chinese-CFL：https://universaldependencies.org/treebanks/zh_cfl/
[1]UD V2：https://universaldependencies.org/guidelines.html

Certificate in English learner corpus）(Yannakoudakis et al., 2011)。TLE对语法错误进行了标注，在一定程度上弥补了SALLE缺乏偏误标注的问题。TLE的构建影响很大：在二语习得领域，它为偏误分析研究提供了语料支持，促进了第二语言教学与量化研究的发展；在自然语言处理领域，TLE为句法分析器提供了大量的训练语料，并通过实验验证了基于L1和L2的平行依存句法树库对提升句法分析器准确率的影响。

与此同时，汉语学习者树库的构建尚在初探过程中。北京语言大学HSK动态作文语料库[2](张宝林, 2009; 张宝林, 2010) 、全球汉语中介语语料库[3] (张宝林and 崔希亮, 2013)等汉语学习者语料库的关注重心主要在字、词等层面的偏误标注上，而对句法结构信息关注度不够。

已有的汉语学习者依存句法树库构建工作也有待完善。香港城市大学制定了面向汉语学习者的标注规范(Lee et al., 2017)，并构建了汉语学习者依存句法树库UD_Chinese-CFL。CFL遵循TLE“字面标注”的原则，结合汉语学习者语料特点，提出了基于UD的汉语学习者依存句法标注规范。但它也存在一些不足之处：标注原则不够清晰，将许多难以满足标注原则的语料当作例外情况处理；标注过程为了适应规范对语料做了一定程度的修改；对于一些汉语的特殊词性、结构未作详尽考虑。

为解决以上问题，肖丹et al. (2019)基于UD V2(Nivre et al., 2020)提出了一个新的面向汉语中介语的依存句法标注规范，考虑了汉语特殊结构的标注方法，并进一步细化了标注原则。然而，在应用该规范对学习者语料进行标注时，我们发现该规范的适应性不够强，在标注原则和依存关系标签上仍需进一步改进，以增强对汉语学习者语料的适应性。

3 汉语学习者依存句法树库构建

本章介绍了汉语学习者依存句法树库的构建工作。首先，我们对肖丹et al. (2019)制定的依存句法标注规范进行改进，提出了更适用于汉语学习者语料的标注规范。其次，我们从HSK动态作文语料库中筛选出带语法偏误信息的语料构建语料库。最后，我们搭建了在线标注平台，通过人机结合的方式，对分词、词性和依存句法进行标注。

3.1 标注规范

在使用肖丹et al. (2019)制定的标注规范进行标注的过程中，我们发现其存在以下问题：

1）从标注原则上看，面对汉语学习者语料的适应性不够强。它参考TLE制定的“字面标注”原则，未考虑到汉语学习者语料有不同于英语学习者语料的特点。如英语二语学习者在核心动词方面犯的错误往往是动词时态错误，而汉语二语学习者在核心动词上犯的较多的则是缺失错误。核心动词缺失会对句意理解造成很大影响，导致标注者在执行“字面标注”原则的时候，根据个人理解对原意进行判断，随意性较大。

2）从标注框架上来看，其框架未涵盖所有语言现象，导致一些汉语独特的语言结构只能与其他结构共用标签，难以区分，并且可能导致结构关系模糊不明。

针对以上问题，本文对该标注规范进行调整，提出了更细致、更符合汉语特点和汉语学习者语料特点的标注规范。制定标注原则时，充分考察汉语学习者语料，以期增强该规范对汉语学习者语料的适应性。现标注原则包括核心标注原则和非核心标注原则。

核心标注原则：根据纠偏后的目标句进行分词、词性标注和依存句法分析。核心标注原则是最重要的标注原则，即在拿到一个偏误句时，我们首先要尽量根据偏误纠正后获得的目标句的句法结构进行偏误句的依存句法标注。如图1，偏误句中“共供”为别字，同时遗漏了介词“对”，但大的语言单位的句法属性未发生改变，“公共场所的卫生”仍在句中作状语，对于这样的无法判断其句法结构、句法结构不合法或遗漏的单位不会使句法结构发生改变等情况，按照核心标注原则进行标注。

非核心标注原则：根据所观察到的句法结构进行分词、词性标注和依存句法分析。在核心标注原则不适用时，采用非核心标注原则。如图2，如果依照核心标注原则，偏误句中root应在“深入”上，但是，这样就没有办法处理“达到”。因为在我们的标注体系中，表示两个动词之间关系的标签有“conj”和“xcomp”，而这两个标签的方向都是从第一个词指向第二个词。所以，如果按照核心标注原则标注的话，就会和所制定的依存标注框架相违背，因此需要根据观

[2]北京语言大学HSK 动态作文语料库：http://hsk.blcu.edu.cn
[3]全球汉语中介语语料库：http://qqk.blcu.edu.cn/

察到的偏误句句法结构，将root放在“达到”上。对于此类句法结构发生改变或其他原因导致核心标注原则不适用的偏误句，按照非核心标注原则进行标注。

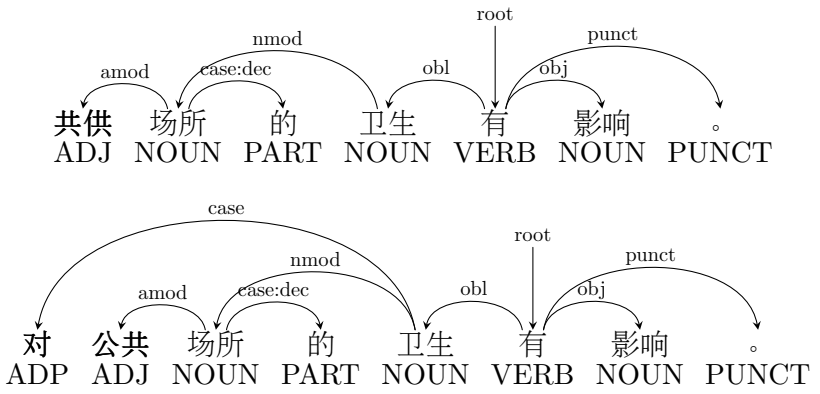


Figure 1: 核心标注原则示例

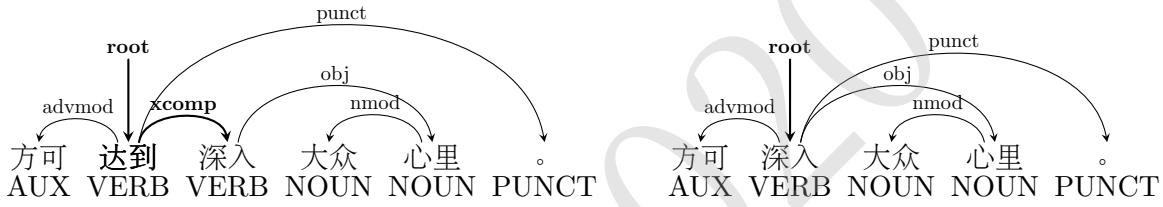


Figure 2: 非核心标注原则示例

制定依存关系标签时，充分借鉴UD体系，在标注语料的过程中弥补现存框架的不足，新增了2个标签类型。具体新增标签如下：

补语标签“xcomp:comp”

在原标注框架中，标签xcomp既用于联结相同主语的两个动词，又用于联结中补结构中的补语与中心语。为了加以区分，现规定xcomp仅用于联结相同主语的两个动词，新增标签xcomp:comp来联结中补结构中的补语与中心语，如图3。

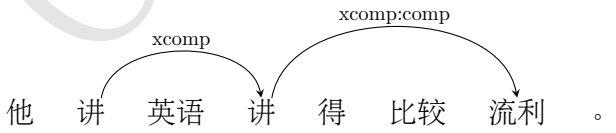


Figure 3: xcomp:comp用法示例

并列关系复句标签“dep:conj”

在原标注框架中，标签dep联结复句中的两个小句。然而当两个小句是并列关系时，就可能导致关系层次模糊不清。例如图4(a)与(b)的复句层次关系不同：(a)中的C小句与A、B小句分别构成并列关系，而(b)中的C小句与A小句非并列关系。但是它们的标签与依存弧方向完全一致，光看图难以区别这两种层次关系。为解决这一问题，为复句间的并列关系另设标签dep:conj，使得复句层次关系更加清晰，如图5。

3.2 数据选取

对汉语学习者语料进行依存句法分析，可以使我们直观地看到汉语学习者语料的，特别是存在偏误等语言现象的语料的句法结构。本文选取带有偏误信息的汉语学习者语料，通过对典型语料的分析窥探汉语学习者的语言特征，并期望对自动句法分析等研究有所帮助。

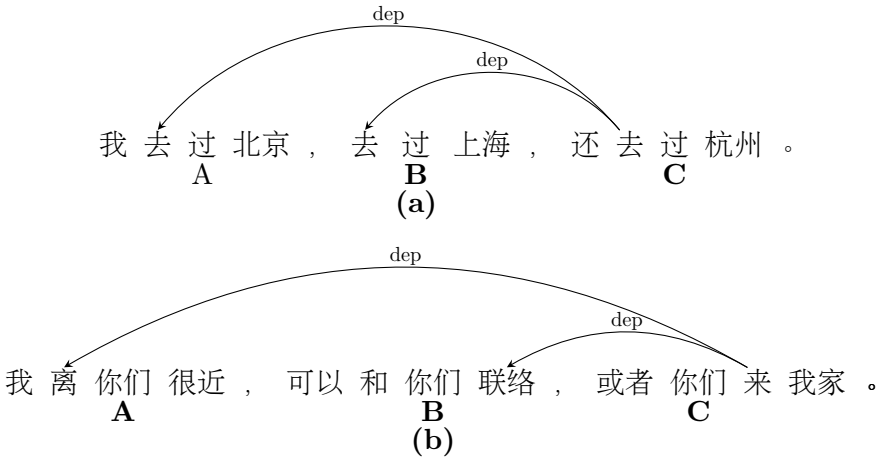


Figure 4: 原标注规范标注示例

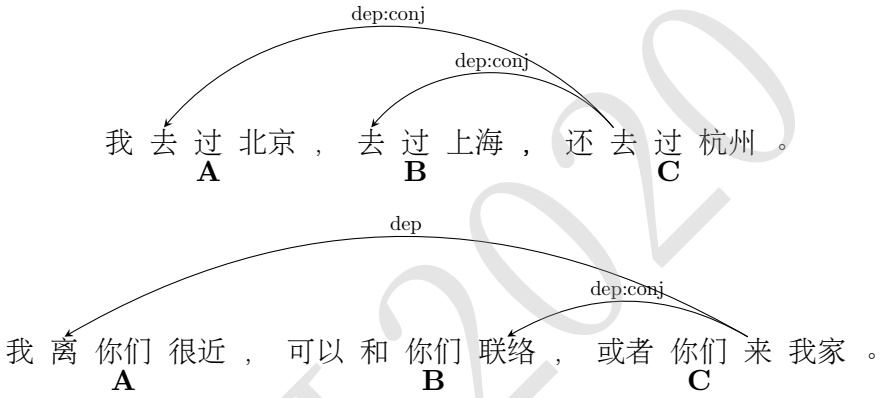


Figure 5: 现标注规范标注示例

首先，从北京语言大学构建的HSK动态作文语料库中选取了9626个带有偏误标注的句子。HSK动态作文语料库是目前国内最大的汉语中介语语料库，它总共收录了11569篇1992-2005年不同母语的留学生作文试卷的语料，并且对字、词、句和篇章进行了偏误标注和修改。为了保证语料的广泛性，以及抽取语料的平衡性和综合性，从包括叙事、议论、应用等不同文体的29个话题中均衡地抽取语料。选取语料的作者母语各异，包括韩语、日语、英语、俄语、法语、蒙古语等14种语言，遍布五大语系，含有孤立语、屈折语、黏着语三种语言形态类型，这可为语言迁移研究提供数据。在选取的篇章上进行分句时，由于学习者语料的标点可能存在偏误，因此按照偏误修改后的语料进行分句，以“。”“！”“？”等标点进行切分。

其次，以语法点为切入点，试图选取具有有效、典型偏误信息的语料。在对外汉语教学当中，语法点教学是常用的一种教学手段，它是对系统语法进行切分之后的结果。本文借鉴了北京师范大学的汉语国际教育动态语料库（CTC）[4](谭晓平et al., 2015)中的语法点信息，并进行了增补和筛选，总结了包括复句、固定结构、介词及介词结构、特殊句型四大类的137个语法点。随后采用人机互助的形式对语料进行语法点标注，共提取出了1056句带有语法点偏误信息的句子。

3.3 标注流程

首先，在标注之前对语料进行一定处理，得出适于标注并易于进行对比分析的偏误句与目标句的平行句对；其次，开展两个层面的标注：词层面的分词及词性标注，句法层面的依存句法标注。在这两个层面上，都采用人机结合的方式标注，并设置了标注员和审核员两种角色，

[4]汉语国际教育动态语料库：http://www.aihanyu.org/basic.aspx

既可以提高标注的效率，也能保证标注的质量。

3.3.1 语料预处理

本树库面向汉语学习者，并期望探索汉语学习者偏误对依存分析的影响，因此树库中的语料均以一组平行句对的形式显示，即汉语学习者生成的偏误句与母语者修改后正确的目标句。HSK动态作文语料库已对偏误句的字、词、标点偏误进行了一定的修改，对于此类已给出修改结果的偏误标注，可以通过程序直接获得偏误句和目标句。

然而，对于句式方面的偏误，HSK语料库只进行了偏误标注，而没有进行修改。比如“人类是有精神方面的追求{CJsd}，必须满足其需求。”一句中，{CJsd}表示此句是“是.....的”句式错误，但没有明确表明应如何修改。此外，HSK语料中还有一些错误存在漏标的情况。针对这两类句子，我们对其进行人工修改，将原句按照最小改动的原则修改为符合汉语语法的句子，得到正确的目标句。在492句偏误句当中，共对227句语料进行了人工修改。

3.3.2 语料标注

标注过程包括分词、词性标注和依存标注。分词及词性标注是做好依存标注的前提，其标注质量直接关系着依存句法的标注质量，一旦分词和词性出现错误，会使得句法层面的标注难以进行，增大标注员的负担。

UD_Chinese-GSD[5] 是谷歌在UD上发表的汉语树库，本文参考了它的分词及词性标准，在实际标注时采用字标注的方式进行标注，如“苍白”一词为形容词（ADJ），则对这两个字分别标注“B-ADJ”和“I-ADJ”两个标签。但在标注过程中发现GSD的分词和词性标注规范仍有一定不足之处，主要有：1）部分出现在相同语言环境中的同一个词的词性不同。2）部分出现在相同语言环境中的同一个词的分词情况不同。3）部分词的分词情况与词典有差别。4）相同成分构成的词，有的被分开列为两个词，有的合起来标为一个词。如同样是“ADV+‘是’”的结构，“都是”、“不是”被标为一个词，“总是”则被分开标为“总”和“是”两个词。针对这样的一些问题，我们参考了宾州中文树库CTB的分词(Xia, 2000b)及词性规范(Xia, 2000a)，对GSD的分词和词性标注规范进行了部分修改，如：1）现阶段不对词缀、类词缀进行单独分词，在后期工作中如需拆分词缀，则针对词缀、类词缀的判别进行规定后制定词缀表，再对词缀进行拆分。2）把成语和俗语当作一个整体来标注。因为成语和俗语是一种相沿习用的固定短语，具有结构定型、意义完整两个基本特点。它们在语句中是作为一个整体来应用的(王兴全and 方忠, 2017)，其意义通常不是字面意义的简单相加。如果拆分进行标注的话，会破坏其所要表达的意义。分词与词性标注完成后，即可进行依存分析。

为了提高标注的效率和一致性，我们采用人机结合的方式进行标注。标注流程如下：

1）用GSD数据训练分词、词性和依存句法分析模型，并得出语料的分析结果。

2）将每条句子的分析结果随机分配给两位标注者标注。标注完成后，若标注结果完全一致，则确立为标准答案，结束流程；否则进入步骤3）。

3）如果两个标注答案不完全一致，则将这条句子分配给专家进行审核，由专家给出标准答案，结束流程。

3.4 标注平台

为了提高标注的效率，降低标注的难度，本文基于Arborator [6]开发了一个在线标注平台[7]。该平台操作简单，可以多人同时进行标注，也能减少标注管理者的工作，便于查看标注的进度和管理标注结果。标注界面如图6所示。

标注平台的主要功能有：

1）修改句子的分词错误和词性标签。当存在分词错误时，标注员或审核员可通过删除、增添、替换等操作对分词情况进行修改；当存在词性错误时，标注员和审核员可在预设好的词性标签中选择正确的标签进行修改，避免人工输入时可能产生的误操作。

2）标注依存弧及标签。标注员和审核员可通过将依存弧从弧起始端的词语拖拽至弧末端对应的词语来标注依存弧，并通过预设的依存标签栏选择正确的依存标签。

[5]UD_Chinese-GSD：https://universaldependencies.org/treebanks/zh_gsdsimp/
[6]Arborator：https://github.com/Arborator/arborator-server
[7]https://yatlc.wenmind.net/

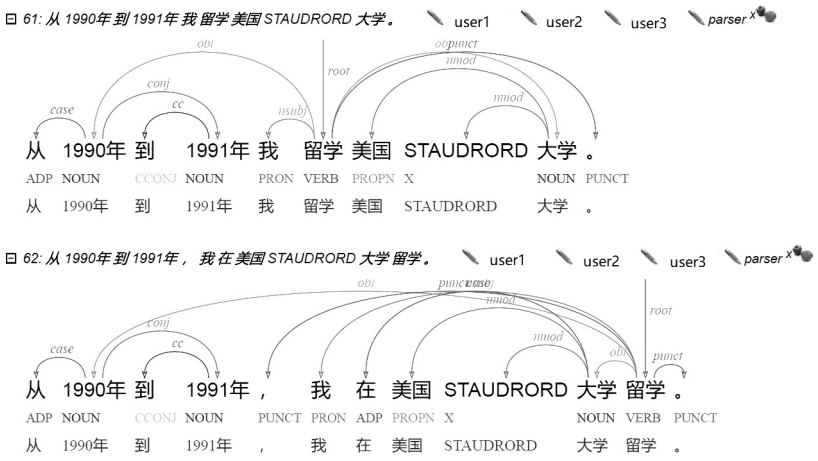


Figure 6: 标注平台界面

3）对比多人标注结果。审核员可以自由选择想要对比的多人标注结果，标注一致的显示为灰色，不一致的弧或标签则以不同的颜色显示不同的标注结果，极大地方便了审核工作。

4 标注数据分析

基于以上标注流程，本文对100条语料进行了依存标注，其中每条语料都至少含有一个非标点符号的语法错误，平均句长为35字，平均每条语料带有4处偏误。首先，统计整体依存标签的分布情况，并尝试从语言学的角度来解释。其次，通过分析两位标注员之间的一致性来观察标注的质量，并通过对比偏误句中带偏误标记的词与无偏误标记的词的一致性来分析偏误对标注质量的影响。最后，分析偏误类型对依存句法分析的影响。

4.1 依存标签分布情况分析

本文对语料库中的依存标签的数量和分布情况进行了统计，总共包含35类、4554个标签。由于个别标签出现的次数非常少，所以只呈现了出现次数在100次以上的依存标签的分布情况，如图7。经过对各标签情况的观察、分析，可以获得以下信息：

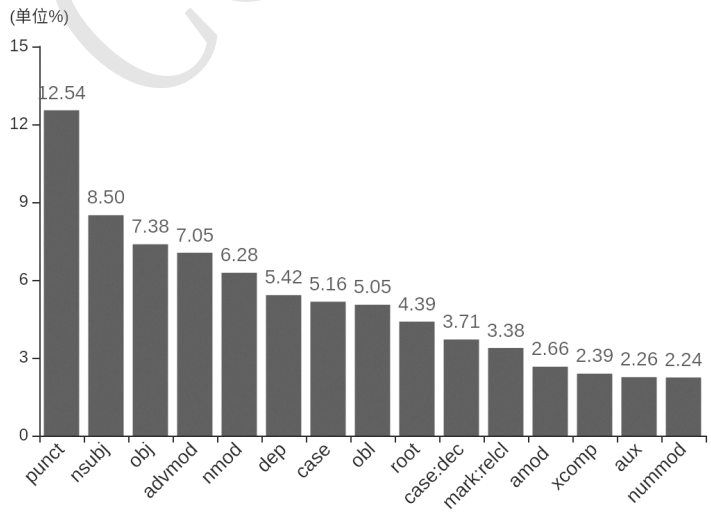


Figure 7: 高频标签分布情况

1）主干成分占比高。句法结构中的主干成分，如表示主语的“nsubj”、表示宾语的“obj”、表示状语的“advmod”“obl”、表示定语的“nmod”“amod”“nummod”等占比明显高于其他成分占

比，基本符合人们的语言认知。这说明，从总体上看我们的标注规范是科学、合理的。在高频标签中没有出现补语标签“xcomp:ccomp”，补语是汉语与印欧语言的一大区别，印欧语言中没有补语这种句法成分，因此二语学习者在语言迁移的过程中受到母语的影响，较少地使用补语成分。

2）复句较多。排名前十的标签中，除了句子的主干成分以外，还包括“punct”、“dep”两类标签。“dep”是表示复句关系的标签。“punct”表示标点，排名第一，占比远高于排名第九表示根节点的标签“root”，说明许多句子包含多个标点符号，符合复句的特点。这是因为语料主要来源于考试中的作文语料，学习者为了得到较高的分数，会更多地使用长难句。这反映了语料的特点：句子长度大，复句较多，各小句间的层次关系比较复杂。

4.2 语法偏误对标注的影响

本文通过计算两位标注员在偏误句、目标句、偏误句中的偏误词语和无偏误词语的一致性来进一步分析语法偏误是否会对标注质量产生较大的影响。

对于一致性和准确率的计算方法如下:

依存弧的一致性（C_A）：假设一个句子有m条依存弧，a和b两个人都对其进行标注。如果两人的标注结果中一致的依存弧为i_A条，则$C_A = i_A/m$;

依存标签的一致性（C_L）：假设一个句子有m条依存弧，a和b两个人都对其进行标注。如果两人的标注结果中有i_L个依存标签相同，则$C_L = i_L/m$;

带标签的依存弧的一致性（C_{LA}）：假设一个句子有m条依存弧，a和b两个人都对其进行标注。如果两人的标注结果中有i_{LA}条依存弧相同且依存标签也相同，则$C_{LA} = i_{LA}/m$;

我们统计了两位标注员间的一致性，如表1所示。首先，以句为单位，对两位标注员的一致性进行统计。“所有句”表示两位标注员标注的所有数据，“偏误句”表示带有偏误的句子，“目标句”是对偏误进行修改后的句子。其次，我们作出一个假设：语法偏误对标注的一致性有影响。为此，通过编辑距离将偏误句与目标句进行比对，提取出偏误句中经过“替换”、“删除”操作的词，将之称为“偏误词”，其他未发生改变的词，称之为“无偏误词”。

类别	C_A	C_L	C_{LA}
所有句	91.59	92.75	87.77
目标句	92.14	93.40	88.59
偏误句	91.02	92.09	86.93
无偏误词	91.07	92.49	87.09
偏误词	90.64	89.14	85.77

Table 1: 一致性分析

由表1中的数据可得到以下结论:

1）标注员总体的标注一致性较高。这在一定程度上肯定了标注规范与流程的合理性。

2）在依存弧、依存标签和带标签的依存弧上，偏误句的一致性都明显低于目标句，偏误词的一致性都低于非偏误词。这可以表明语法偏误在一定程度上增加了标注的难度，对标注的一致性造成影响。

3）比较“C_A”和“C_L”两列数据，在“所有句”、“目标句”、“偏误句”、“无偏误词”中，依存标签的一致性都高于依存弧。其原因是存在某些会产生连锁反应的标签，如表示根节点的“root”，当这些标签指向的词不一致时，会导致这些词与子节点间的弧也发生改变，而其上的一些依存标签却不发生改变。当“root”所指的词不一致时，指向语气词的弧会发生改变，但“discourse”标签不发生改变。在“偏误词”中，依存标签的一致性却低于依存弧，并且与“无偏误词”相比，相较于依存弧，依存标签的一致性差别明显更大，说明语法偏误对依存关系的影响比对依存结构的影响更大。

4.3 语法偏误对依存句法分析的影响

鲁健骥 (1994)认为偏误类型可分为四种：“误加”、“遗漏”、“误代”、“错序”。遗漏偏误是指由于在词语或句子中遗漏了某个/几个成分导致的偏误；误加偏误是指当某些语法形式发生某种变化时，在通常情况下可以/必须使用的某个成分变为一定不能使用这个成分，而汉语学习者

往往不了解这种条件的变化仍然使用这个成分，从而导致的偏误；误代偏误是指从两个或几个形式中选取了不合适于特定语言环境的一个词造成的偏误；错序偏误指的是由于句中的某个或几个成分放错了位置造成的偏误(鲁健骥, 1994)。本文按照这种分类，探索偏误类型对句法分析的影响，以帮助训练二语的句法分析器，提高二语语法纠错任务的准确率。

在对语料中出现的偏误考察后发现:

1）当出现“误加”或“遗漏”偏误，即词语冗余或缺失时，如果该偏误在句中承担核心句法成分或者和其他词语构成某个结构共同承担核心句法成分时，那么该偏误对句法分析的影响较大，即平行句对的标注结果大有不同，具体表现为根节点与依存关系标签发生变化，如图8；反之则适中，即平行句对的标注结果略有不同。略有不同表现在多一个弧、少一个弧、弧长度有所不同或弧交叉，但根节点与依存关系标签未发生变化，如图9。

2）当出现“误代”偏误，即词语使用错误时，如果该词与修改后的词具有相同的词性，且出现的句法环境也相同，那么该偏误对句法分析的影响较小，即平行句对的标注结果基本一样，如图10；反之则较大。

3）当出现“错序”偏误，即词语的位置发生了改变时，如果所修饰的核心词没有发生改变，仅仅是弧的距离发生改变，那么该偏误对句法分析的影响适中；如果所修饰的核心词发生改变，即弧的父亲节点发生改变，那么该偏误对句法分析的影响较大。

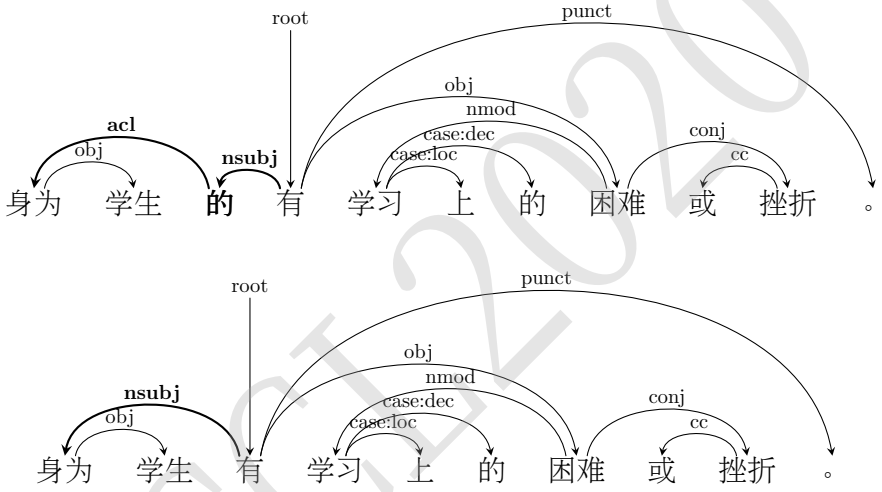


Figure 8: “误加”偏误对句法分析影响较大示例

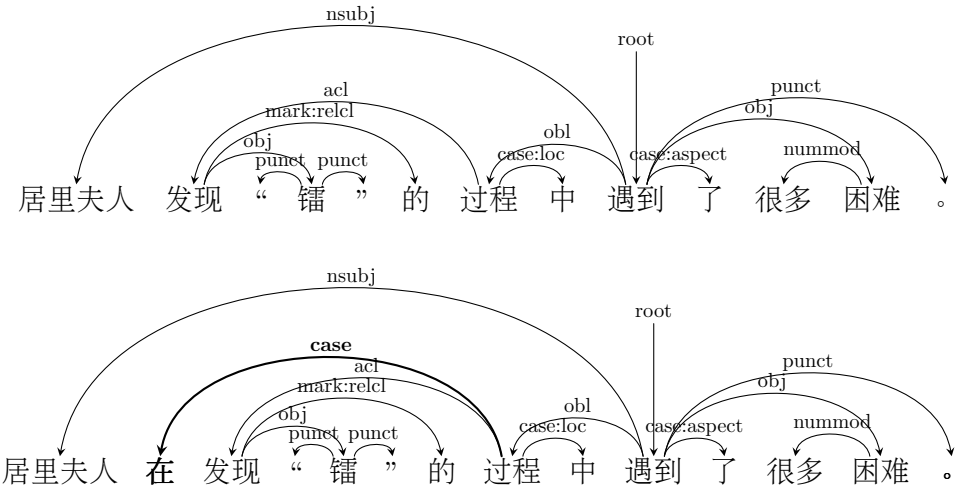


Figure 9: “遗漏”偏误对句法分析影响适中示例

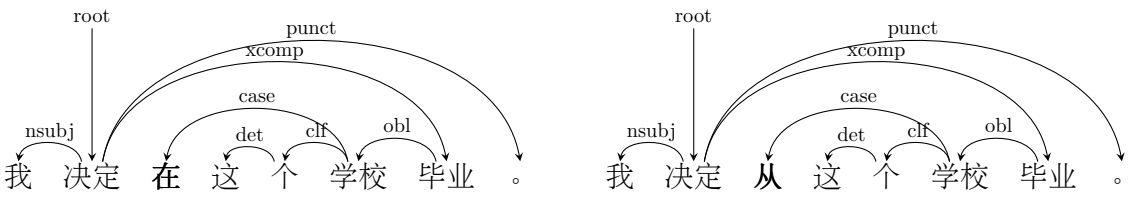


Figure 10: “误代”偏误对句法分析影响较小示例

5 总结与展望

本文介绍了我们在汉语学习者依存句法树库构建上所做的一些工作，包括标注规范的改进、数据选取、标注流程等，并对依存关系标签分布情况及偏误对标注质量及依存句法的影响情况进行了分析。本文的创新之处在于：1）对依存句法标注原则进行了改进，并弥补了现有标注框架的不足，考虑到了汉语及汉语学习者语料的特点，增强了对汉语学习者语料的适应性。2）在严格的标注流程控制后，通过统计与分析发现语法偏误对标注质量和依存句法分析都有一定的影响，对待不同的偏误要采取不同的标注策略，以降低标注难度，节约标注时间。目前我们树库的规模还比较小，未来我们将继续完善标注规范，在标注规范的指导下扩大树库规模，为二语教学与研究提供更多帮助，也为句法分析器、语法改错等相关研究提供更多数据支持。

参考文献

Yevgeni Berzak, Jessica Kenney, Carolyn Spadine, Jing Xian Wang, Lucia Lam, Keiko Sophie Mori, Sebastian Garza, and Boris Katz. 2016. Universal dependencies for learner English. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 737–746, Berlin, Germany, August. Association for Computational Linguistics.

Marie-Catherine de Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D. Manning. 2014. Universal Stanford dependencies: A cross-linguistic typology. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4585–4592, Reykjavik, Iceland, May. European Language Resources Association (ELRA).

Sylviane Granger. 2012. Learner corpora. *The encyclopedia of applied linguistics*, pages 1–8.

John Lee, Herman Leung, and Keying Li. 2017. Towards universal dependencies for learner chinese. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 67–71.

Brian MacWhinney. 2014. *The CHILDES project: Tools for analyzing talk, Volume I: Transcription format and programs*. Psychology Press.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Joakim Nivre, Marie-Catherine de Marneffe, F. Ginter, Jan Hajivc, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and D. Zeman. 2020. Universal dependencies v2: An evergrowing multilingual treebank collection. In *LREC*.

Marwa Ragheb and Markus Dickinson. 2014. Developing a corpus of syntactically-annotated learner language for english. *CLARIN-D*, pages 292–300.

Geoffrey Sampson. 2011. Susanne-a deeply analysed corpus of american english. *New Directions in English Language Corpora: Methodology, Results, Software Developments*, 9:171.

Fei Xia. 2000a. The part-of-speech tagging guidelines for the penn chinese treebank (3.0). *IRCS Technical Reports Series*, page 38.

Fei Xia. 2000b. The segmentation guidelines for the penn chinese treebank (3.0).

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading esol texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 180–189. Association for Computational Linguistics.

刘挺and 马金山. 2009. 汉语自动句法分析的理论与方法. 当代语言学, 011(2):100–112.

张宝林and 崔希亮. 2013. "全球汉语中介语语料库建设和研究"的设计理念. 语言教学与研究, (05):27–34.

张宝林. 2009. "HSK动态作文语料库"的特色与功能. 国际汉语教育, (4):71–79.

张宝林. 2010. 汉语中介语语料库建设的现状与对策. 语言文字应用, 000(3):129–138.

李娟, 谭晓平, and 杨丽姣. 2016. 汉语中介语语料库应用及发展对策研究. 曲靖师范学院学报, 35(2):86–91.

王兴全and 方忠. 2017. 现代出版物语言文字使用规范. 电子科技大学出版社.

肖丹, 杨尔弘, 张明慧, 陆天荧, and 杨麟儿. 2019. 面向汉语中介语的依存句法标注规范. 第十八届中国计算语言学大会（CCL 2019）.

谭晓平, 杨丽姣, and 苏靖杰. 2015. 面向汉语(二语)教学的语法点知识库构建及语法点标注研究. 中文信息学报, 29(6):54.

郭丽娟. 2019. 汉语依存句法分析树库构建与应用研究. Ph.D. thesis, 苏州大学.

鲁健骥. 1984. 中介语理论与外国人学习汉语的语音偏误分析. 语言教学与研究, (03):44–56.

鲁健骥. 1994. 外国人学汉语的语法偏误分析. 语言教学与研究, (01):49–64.

黄昌宁and 靳光瑾. 2013. 从宾州中文树库观察三个汉语语法问题. 语言科学, 12(2):178–192.

CDCPP：跨领域中文标点符号预测

刘鹏远　　王伟康　　邱立坤　　杜冰洁
北京语言大学信息科学学院
国家语言资源监测与研究平面媒体中心
闽江学院计算机与控制工程学院
liupengyuan@pku.edu.cn　978955719wwk@gmail.com
qiulikun@pku.edu.cn　blcudbj@gmail.com

摘要

标点符号对文本理解起很大作用。但目前，在中文文本特别是在社交媒体及问答领域文本中的标点符号使用存在非常多的错误或缺失的情况，这严重影响对其进行语义分析及机器翻译等各项自然语言处理的效果。当前对标点符号进行预测的相关研究多集中于英文对话的语音转写文本，缺少对社交媒体及问答领域文本进行标点预测的相关研究，也没有这些领域公开的数据集。本文首先提出跨领域中文标点符号预测任务，该任务是要利用标点符号基本规范正确的大规模新闻领域文本，建立标点符号预测模型，然后在标点符号标注不规范的社交媒体及问答领域，进行跨领域标点符号预测。随后构建了新闻、社交媒体及问答三个领域的相应数据集。最后还实现了一个基于BERT的标点符号预测基线模型并在该数据集上进行了实验与分析。实验结果表明，直接利用新闻领域训练的模型，在社交媒体及问答领域上进行标点符号预测的性能均有所下降，在问答领域下降较小，在微博领域下降较大，超过20%，跨领域标点符号预测任务具有一定的挑战性。

关键词： 中文标点符号预测任务 ； 跨领域 ； 数据集

CDCPP: Cross-Domain Chinese Punctuation Prediction

PengYuan Liu　　WeiKang Wang　　LiKun Qiu　　BingJie Du
Beijing Language and Culture University, School of Information Science
Language Resources Monitoring and Reserch Center
Minjiang University,School of Computer and Control Engineering
liupengyuan@pku.edu.cn　978955719wwk@gmail.com
qiulikun@pku.edu.cn　blcudbj@gmail.com

Abstract

Punctuation marks play a important role in text understanding. But at present, there are many errors or lacks in Chinese texts, especially in social media and Q&A texts, which seriously affects the effect of various natural language processing such as semantic analysis and machine translation. The current research on punctuation prediction is mostly focused on the speech transcribed text of English conversations. There is a lack of research on punctuation prediction in social media and Q&A texts, and there is no public dataset in these fields. This paper first proposes a cross-domain Chinese punctuation prediction task. The task is to build a punctuation prediction model using a large-scale news domain text that is basically standardized and correct punctuation, and then conduct punctuation predict in the social media and Q&A fields where punctuation is not standardized. Subsequently, corresponding datasets in the three fields of news, social media and Q&A were constructed. Finally, a BERT-based punctuation prediction baseline model was implemented and experiments were

performed on this dataset. The experimental results show that directly using the model trained in the news field, the performance of punctuation prediction in the social media and Q&A fields has decreased, and the decrease in the Q&A field is small, and the decrease in the Weibo field is large, exceeding 20% . Cross-domain punctuation prediction task has certain challenges.

Keywords: Chinese Punctuation Prediction Task , Cross-Domain , Dataset

1 引言

汉语书面语中，标点符号有着不可或缺的地位。《辞海》[0]中把标点解释为“书面语里用来表示停顿、语调以及语词的性质和作用的符号，是书面语的有机组成部分”。它可以帮助人们确切地表达思想感情和理解书面语言。近年来，随着社交媒体领域（如微博）及问答领域（如百度知道）及应用（如问答机器人）的活跃兴起，对社交媒体及问答领域文本的处理变得愈来愈重要。但这两类文本常常出现标点符号使用错误、缺失甚至完全不使用标点符号的情况。图1是标点符号标注错误及正确实例对语义分析[1]及机器翻译[2]影响的对比。其中：c及c’分别为百度知道[3]中的实例及人工对其进行标点重新标注后的文本；e及e’是分别对c及c’用谷歌翻译的结果；s及s’是对两个中文文本分别利用LTP平台进行语义依存标注的结果（限于篇幅，仅截取了部分）。对比英文译文，可见标点符号错误不但引起局部翻译错误，也影响整句的翻译质量。对比语义依存自动分析的结果，出现标点错误的地方，均导致自动语义分析标注的结果产生错误。我们还随机抽取了新浪微博文本100条并对其中的标点符号进行了人工排查，发现其中有82条标点符号缺失或使用错误。这将对NLP任务的处理如句法分析、语义分析及机器翻译等各项自然语言处理任务的效果带来产生很大影响。为社交媒体及问答等领域中的文本标注正确的标点符号，具有较高的意义和应用价值。

标点符号预测（Punctuation Prediction，PP）或标点符号恢复（Punctuation Restoration，PR）指利用计算机对无标点文本进行标点预测，使得预测之后的文本符合自身语义和标点使用规范。因为语音识别出的序列中没有标点符号，故而标点符号预测相关研究工作集中在语音识别领域，主要是面向对话领域的语音转写文本进行标点符号预测(Beeferman et al., 1998; Liu et al., 2006; Lu and Ng, 2010; Peitz et al., 2011; Tilk and Alumäe, 2015)。目前常用公开的数据集为IWSLT(Federico et al., 2012)，是针对语音领域的英文文本。迄今为止，尚无公开的社交媒体领域相关数据集，这对在该领域进行标点符号预测进行研究产生很大限制。由于社交媒体及问答领域文本中标点符号缺失或使用错误较多，直接利用社交媒体及问答领域文本进行模型训练再进行自动标点符号预测意义不大，而人工标注一个大规模社交媒体及问答领域标点符号预测数据集又非常费时费力。与此同时，新闻领域中的文本，标点符号用法基本规范，可认为是标点符号使用正确的实例，建立PP/PR 任务的数据集较为容易，但面向新闻领域文本进行标点符号预测，应用价值较低。

基于以上现状，本文提出跨领域中文标点符号预测任务，该任务是要利用标点符号基本规范正确的大规模新闻领域文本，建立标点符号预测模型，然后在标点符号标注不规范的社交媒体及问答领域，进行跨领域标点符号预测。本文构建了新闻、社交媒体及问答三个领域的数据集[4]。在新闻领域，提供了测试集，共10000条。新闻领域的文本较容易获得，且标点符号使用非常规范，因此在大规模新闻文本上进行训练并进行标点符号预测，可视为各种方法在其他领域预测性能的上限（upper bound）。在社交媒体和问答领域，本文分别提供了人工标注的测试集各1200条。除测试集外，本任务没有提供（但不禁止使用）社交媒体和问答领域这两个目标领域内的任何数据。

为对本任务及数据集进行初步评估，鉴于近年来预训练语言模型BRET（Bidirectional Encoder Representation from Transformers）(Devlin et al., 2018)在NLP领域各项任务中的优良

[0]http://chlb.cishu.com.cn/
[1]采用哈工大语言技术LTP平台：http://ltp.ai/index.html
[2]采用谷歌翻译：https://translate.google.cn
[3]http://zhidao.baidu.com
[4]https://github.com/NLPBLCU/Cross-Domain-Chinese-Punctuation-Prediction

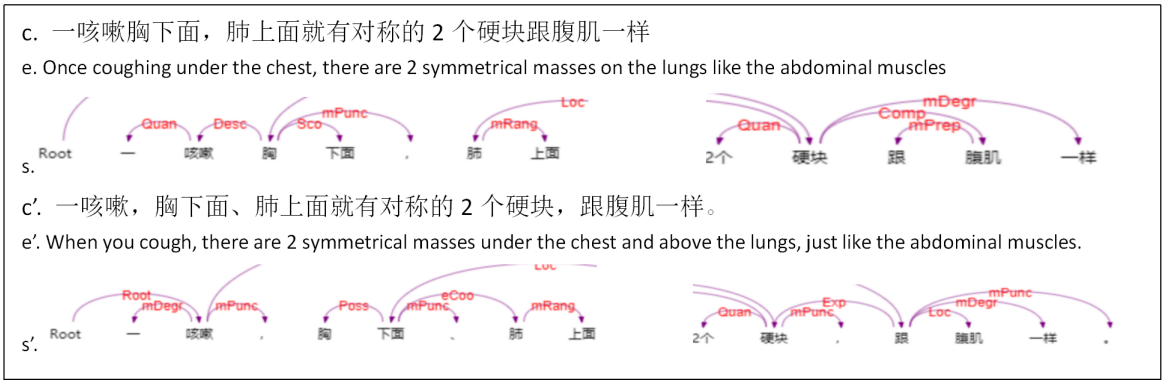


Figure 1: 标点符号标注错误对依存语义分析及机器翻译结果的影响。其中c及c'分别为百度知道中的实例及人工对其进行标点重新标注后的文本。e及e'是分别对c及c'用谷歌翻译的结果。s及s'是分别对c及c'利用LTP平台进行语义依存标注的结果，限于篇幅，仅截取了部分。

表现，我们实现了一个基于BERT的序列标注模型作为数据集的基线模型，同时还使用了Focal Loss(Lin et al., 2017)作为训练过程中的损失函数来缓解类别不平衡问题。在本数据集上的实验结果表明，直接利用新闻领域训练的模型，在社交媒体及问答领域上进行标点符号预测的性能均有有所下降，特别是在微博领域下降较大，跨领域标点符号预测任务具有一定的挑战性。

2 数据集构建

2.1 数据准备

新闻领域选择人民日报2018年全年语料。该语料共574332条文本。我们首先按照本文标点符号标签表（见第三小节中的表3）对该文本进行处理：去除无关的一些噪音标点和符号，对标点进行全半角和重复的符号归一化处理，再将冒号替换为逗号，将感叹号和省略号替换为句号。然后，在该语料中随机抽取10000条作为测试集，另随机抽取100000条作为训练集。

社交媒体领域选择新浪微博[5]，新浪微博是目前中文影响力最大的社交媒体，基于新浪微博进行自然语言处理的研究非常广泛(谢丽星，周明，孙茂松, 2012; 古万荣，董守斌，曾之肇，何锦潮，刘崇, 2016; 贺敏，刘玮，刘悦，王丽宏，白硕，程学旗, 2017; 王志宏，过弋, 2019)。本文随机爬取微博语料共120250条，语料经数据预处理之后统计得平均文本长度（含标点）为66.41，标准差为55.64，分布不太均衡，为考察不同文本长度对模型的影响，我们随机选取文本长度在65-67之间（中等长度文本）及文本长度在100-110之间（长文本）的两类，去掉其中所有标点及空格，作为社交媒体领域测试集待标注语料，并分别记为"微博（中）及微博（长）。

问答领域语料来源于中文问答匹配数据集LCQMC(Liu et al., 2018)，该数据集被广泛用于中文问答特别是问句匹配（question matching ）中。该数据集中的语料来源于百度知道，共260068对句子，经过去重后，统计得平均文本长度为10.73，标准差为4.0，平均文本长度较微博更短，文本长度分布也相对均衡，因此考察不同文本长度对模型的影响意义较小。通过对语料的观察我们发现，文本长度在15以内的文本中，标点符号相对较少，因此我们随机抽取文本长度在15以上的文本，去掉原语料中所有标点及空格，作为问答领域测试集待标注语料。

2.2 标注规范

标注规范采用2011年由中国国家标准化管理委员会发布，2012年实施的《标点符号用法》（以下简称《用法》）文件[6]中的标准。《用法》将标点符号分为点号和标号。其中点号的作用是点断，主要表示停顿和语气。而标号的作用是标示某些成分的特定性质和作用。本次主要标注点号，即句号、问号、叹号、逗号、顿号、分号、冒号。在这些点号中，叹号主要用在句末表达情感，句子的情感时常因人而异，有时可用句号替代。冒号用于句中，表示语段中提示下

[5]http://weibo.com
[6]http://openstd.samr.gov.cn/bzgk/gb/newGbInfo?hcno=22EA6D162E4110E752259661E1A0D0A8

文或总结上文的停顿，其部分功能与逗号类似。由本文任务的目的出发，我们选择最终标注的标点符号只有五种：逗号、顿号、分号、句号、问号。

2.3 标注过程

整个标注由3名语言学在读硕士生作为标注员共同完成。首先是研读规范并进行试标注，对不一致的地方进行讨论以及规范参考。待3名标注员均熟悉规范和标注后再进行正式标注。

对给定的问答领域及微博领域的测试集待标注语料，由两名标注员分别进行标注。在标注过程中，如果遇到难以理解的句子，由于很难对其进行正确的标点符号标注，标注员就直接抛弃该条文本。语料中出现的其他非标点符号，如表情符号等手动删除。对新浪微博语料，我们控制标注每种文本长度的语料各600条，共1200条。对百度知道语料，我们控制标注共1200条。标注完成后，由第三名标注员进行审核。审核的标准是排除标点符号使用不符合《用法》的原则性错误。审核后的文本，如2名标注员标注结果一致，则作为金标准文本保留，对于标注不一致的文本，由第三名标注员进行仲裁。

由于《用法》并没有对标点符号进行严格的标注规定，特别是句中的成分之间是否停顿因使用场景、使用习惯等存在差异，如：

a.突击停产后，企业为了抢回时间满足订单生产，往往会匆忙复工。这一停一开，可能危机四伏。

b.突击停产后，企业为了抢回时间满足订单生产往往会匆忙复工，这一停一开可能危机四伏。

两段文本的标注都不存在原则性错误，只是各人语感不同，语块切分的大小不同，这样，对同一文本进行标点符号标注，可能会出现多种正确的标注。因此，在仲裁时，首先由第三名标注员保留仲裁得到的标注结果，作为金标准；然后，三名标注员对本条标注的不同结果进行讨论，对三人加以讨论后确定符合《用法》的结果也加以保留，作为可选标准文本，附加在金标准文本后，以空格分开。

2.4 统计分析

最终形成的数据集共包含问答领域、微博中等及微博长文本数据各1200、600及600条。由于每条可能包含有多个正确的标注结果，实际共包含问答领域、微博（中）及微博（长）数据各1328、803及779句。问答领域、微博（中）及微博（长）数据集的标注一致率分别为：0.9751，0.9572及0.9731[7]。

数据集的基本统计情况及标点符号的分布情况分别见表1及表2。此处仅对金标准文本而没有将可选标准文本包含在内进行统计。实际上，由于备选的正确标注文本不多，因此在平均长度、平均标点个数，平均文本长度与标点个数之比及标点符号分布几个方面，所得到的的结果差异不大。

数据集	条目数	平均长度(字)/条	平均标点个数/条	句长/标点数
新闻	10000	70.16	5.61	**12.51**
问答	1200	26.28	2.64	9.95
微博（中）	600	58.14	6.18	9.41
微博（长）	600	**94.14**	**9.04**	10.41

Table 1: 数据集的基本统计情况

从表1可知，微博（长）的平均长度最长，包含标点个数最多，问答领域的平均文本长最短，包含标点个数最少。从平均文本长与平均标点个数之比可知，微博（中）的标点符号“密度”最大，平均9.41个字就有一个标点，而新闻领域标点符号“密度”最小，平均12.51个字才有一个标点。

从表2可知，各领域中，逗号都是最常用标点符号，分号都是最不常用的标点符号。根据《用法》的规定，顿号常用于重复的词语或成分之间，而分号则多用于并列的分句之间，因此在层级关系上分号的层级要大于顿号，因此顿号的使用比分号多。在问答领域，由于文本普遍较短，没有出现分号。问号与句号在各领域中使用的情况比较复杂，新闻领域中问号比例较低

[7]计算标注一致率时包含标点符号类型的一致和断句的一致，所以五个标点符号以及空符号都包含在内。

标点符号	新闻	问答	微博（中）	微博（长）
逗号	**55.13**	49.13	52.62	55.01
问号	0.61	23.48	8.22	**23.73**
句号	**26.06**	25.21	25.99	15.37
顿号	**17.00**	2.16	12.83	4.96
分号	**1.19**	0.00	0.32	0.91

Table 2: 数据集中各种标点分布情况(%)

符合新闻文体特点，这符合我们的认知。在问答领域，提问较多，因此问号的比例总体高于微博领域。句号在新闻、问答及微博（中）的分布相对接近，且在新闻中的使用相对更多。

针对微博长句种问号使用比句号使用更频繁的现象，我们进一步分析文本的内容，发现微博长句中存在不少连续发问的现象，如：

“渣完基三的直接后果就是，为毛我没有轻功？为毛我没有内力？为毛我只是想去个我想个地方要做交通工具神行不了？为毛下个楼还要等电梯或走楼梯？不能轻功跳过去？桑不起。于是我要睡觉了，梦里好好调戏内功[8]。”

3 任务

3.1 形式化

标点符号预测可视为一个序列标注任务，即给定一个文本输入序列:$X = \{x_1, x_2, x_i, \ldots x_n\}$，需要得到一个标点符号标签序列$Y = \{y_1, y_2, y_i, \ldots y_n\}$。模型所需要预测的标签集合如下表所示。标签集合中，0为无标点（space），1为逗号，2为句号，3为问号，4为顿号，5为分号。

标点	标签	标点	标签
space	0	？	3
，	1	、	4
。	2	；	5

Table 3: 标点标签表

3.2 设置

严格设置。仅以数据集每个条目的金标准文本作为正确的标注结果，即每个待预测文本，其正确的标点符号唯一。此种设置下，各数据集分别命名为：问答-严格，微博（中）-严格，微博（长）-严格。

宽松设置。将数据集中每个条目的所有标注文本作为正确的标注结果，即包含金标准文本也包含可选标准文本，对对每个待遇测文本，部分标注的结果可以不唯一，但都视为正确的标注结果。此种设置下，各数据集分别命名为：问答-宽松，微博（中）-宽松，微博（长）-宽松。

此外，由于数据集中每个条目均可以由多句组成，因此在本文的标点标签体系中，句中标点标签有6种可能（含无标点），但句末标点仅有两种可能：问号/句号，模型会相对容易地学到句末标点的信息。考虑到这个影响，针对句末，使用以下两个设置：

1）**含句末。**即在测试时，将文本中所有标点符号考虑在内（包含句末标点）。

2）**无句末。**即在测试时，不将文本中的句末标点考虑在内。

4 实验

4.1 模型

基于预训练语言模型BRET的方法在NLP领域各项任务上均取得了很好的性能，文本也基于BERT建立了一个简单的标点符号预测基线模型。如下图所示，模型输入一段文本，$X = \{x_1, x_2, x_i, \ldots x_n\}$,BERT首先把模型的输入转化为词嵌入矩阵，再经过一个线性变换层将最后

[8]选自微博（长）标准数据集。

一维的词嵌入维度转换为标签，最后经过softmax层输出序列Y，$Y = \{y_1, y_2, y_i, \ldots y_n\}$，代表每个字后面的标签序列。

从本文第2节表2可知，本数据集中标点符号的分布很不平衡，实际上，文本中的标点符号数量比其中字的数量少得多，因此上述模型输出大部分的标签都是无标点的"0"标签，直接采用交叉熵作为损失函数会导致模型在训练时更倾向于输出无标点类别，使得模型学习不到足够的标点特征。为解决这个问题，我们将原来的交叉熵损失改为现在的Focal Loss损失(Lin et al., 2017)，该损失函数调整了样本在训练中所占的权重。原本的Focal Loss是在二分类中实现的，这里将它扩展到多分类问题当中。原Focal Loss 公式如下：

$$L_{fl} = \begin{cases} -\alpha\,(1-y')^{\gamma}\, log y' & , y = 1 \\ -(1-\alpha) y'^{\gamma} log(1-y') & , y = 0 \end{cases} \tag{1}$$

其中 α 和 γ 是两个可以调节的超参数。

我们将二分类的Focal Loss拓展到多分类中：

$$L_{fl} = -\alpha_i (1-y_i)^{\gamma} log(y^i) \tag{2}$$

其中，α_i 代表第i个标签的调节因子，y_i代表第i个标签的预测概率。

4.2 参数设置

本文使用的BERT模型是Google公开的bert-base[9]，模型是由12层的Transformer encoder预训练而成，自注意力头数为12，隐藏层维度为768，总参数量为110MB。训练时，我们设置的学习率大小是5e-5，批大小是64，Dropout设置为0.25，训练轮数为15轮。

4.3 评价指标

在本文的中文标点符号预测任务中，使用分类问题的评价指标精确度P（Precision）、召回率R（Recall）和F_1 值来评价模型整体性能，以F_1 值作为主要评价指标，具体公式如下图所示：

$$F_1 = \frac{2 * Precison * Recall}{Precison + Recall} \tag{3}$$

4.4 实验结果

首先利用在本文第2小节中整理好的人民日报训练语料进行训练。然后，在本文建立数据集的问答及微博领域下进行测试。实验结果详见表4。表4列出了本文各领域数据集在本文任务设置下的模型性能。基线模型在新闻领域中的性能并无宽松/严格设置，列在最后一行。

可以发现模型在问答领域的性能较好，明显高于微博领域的性能，因为问答领域文本较短，微博领域的文本更不规范，因此这也比较符合实际情况与我们的预期。同时，宽松设置均优于严格设置，含句末设置均优于不含句末设置。

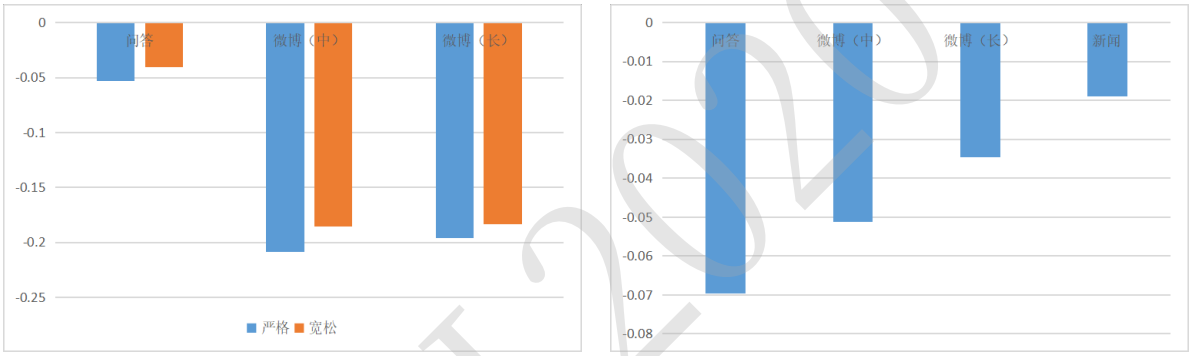
图2（a）是对比新闻领域，严格-宽松两种任务设置下性能下降的柱状图。可以看出，对比新闻领域，基线模型迁移到问答及微博领域后，标点符号预测的性能在所有设置下，均有不同程度的下降，微博领域的下降更多，微博（中）下降的幅度大于微博（长），微博（中）-严格下降的最为明显，超过了20%。跨领域标点标注任务具有一定挑战性，模型性能还有较大提升空间。

图2（b）是无句末设置比含句末设置时模型性能下降的柱状图。在所有领域，无句末均较含句末均有有不同程度下降，其中问答领域下降幅度最高（近7%），微博次之，新闻领域下降最少。问答领域下降幅度最高的原因在于，问答领域中的问句（对应句末问号）或答句（对应句末句号），比较典型，模型相对容易判断。

[9]https://github.com/google-research/bert

数据集	P		R		F_1	
	含句末	无句末	含句末	无句末	含句末	无句末
问答-严格	0.8223	0.7483	0.8348	0.7697	0.8285	0.7589
微博（中）-严格	0.6799	0.6295	0.6661	0.6138	0.6729	0.6216
微博（长）-严格	0.6823	0.6478	0.6889	0.6543	0.6856	0.6510
问答-宽松	0.8350	0.7603	0.8474	0.7884	0.8412	0.7741
微博（中）-宽松	0.6981	0.6509	0.6935	0.6447	0.6958	0.6478
微博（长）-宽松	0.6930	0.6592	0.7029	0.6695	0.6979	0.6643
新闻	0.8834	0.8647	0.8794	0.8600	0.8814	0.8624

Table 4: 基线模型在本文任务设置下各领域数据集上的性能结果



(a) 各领域较新闻领域性能下降柱状图　　(b) 无句末比含句末性能下降柱状图

Figure 2: 各模型性能下降柱状图

5 错误分析与讨论

标点符号的预测错误可分为三种类型：1）漏标，即本应有标点的地方，预测为无标点；2）多标，即本应没有标点的地方，预测出现标点；3）错标，即预测的标点类型与数据集人工标注的标点不一致。前两种类型是断句错误，最后一种类型是标点使用错误。表5为两个领域的标注错误类型分布情况[10]。可以看出，严格和宽松两个领域错误类型分布差别不大。问答领域的漏标错误要明显高于微博领域，相反多标的情况要少于微博领域。受到领域特征的影响，问答领域的文本长度普遍较短，而基线模型是在新闻领域中训练的，新闻领域中平均12.51个字才有一个标点（见2.4节表1），因此，问答领域中的文本，模型在预测时可能会完全不标注。因此问答领域的漏标情况会比微博领域更多。在微博领域中，多标的错误要比其他两种错误要多，这是因为微博领域的文本更加口语化，通常还会有一些专有名词、缩略语、新词新语等，因此在自动标注的过程中，会出现固定搭配之间被插入标点符号的情况，最终导致微博领域的多标错误所占比例较大。

以下是不同领域中的错误案例，各种类型各三组文本，每组文本包含两条文本，第一条文本为自动标注得到的错句，句前有错号，后一句为标准数据集中的正确文本，句前有对号。三组文本分别来自问答、微博（中）及微博（长）：

[10]需要说明的是，由于标点使用不具有唯一性，所以实验得出的错误文本只是相对于标准数据集而言的，机器标注的某些文本虽然不在标准数据集中但其标注也可能是正确的，但这种情况非常少。

数据集	模型预测		
	漏标	多标	错标
问答-严格	39.02	24.97	36.00
微博（中）-严格	25.73	44.03	30.23
微博（长）-严格	29.74	42.74	27.79
问答-宽松	39.16	24.43	36.41
微博（中）-宽松	26.25	42.97	30.79
微博（长）-宽松	29.46	42.35	28.19

Table 5: 基线模型预测错误类型分布及原始标注错误类型分布（%）。由于句末标点只影响错标的分布，且对整体预测错误的分布影响较小，此处没有列出不包含句末标点的情况。

a.漏标

✗巴金的激流三部曲爱情三部曲分别是什么?

✓巴金的激流三部曲、爱情三部曲分别是什么?

✗余华、马原、李耳格非不敢和他们谈文学，但是私下找余华老师看了下牙，他说牙龈是可以自我恢复的，解我多年心头大惑。

✓余华、马原、李耳、格非，不敢和他们谈文学，但是私下找余华老师看了下牙，他说牙龈是可以自我恢复的，解我多年心头大惑。

✗...[11]会上，李公平局长强调，要严肃工作纪律，切实加强对填报志愿工作的督查和管理，确保今年网上填报志愿工作平稳有序顺利进行。

✓...会上，李公平局长强调，要严肃工作纪律，切实加强对填报志愿工作的督查和管理，确保今年网上填报志愿工作平稳、有序、顺利进行。

b.多标

✗可以修改邮箱，请您提供其他邮箱，谢谢，您了。

✓可以修改邮箱，请您提供其他邮箱，谢谢您了。

✗光明能挺住吗? 带不带? 这么恐吓老百姓的? 这个是不是夸张了点啊? ...

✓光明能挺住吗? 带不带这么恐吓老百姓的? 这个是不是夸张了点啊? ...

✗...人是衰，到了何种境界才能发生百年难遇的事? 思绪凌乱了。

✓...人是衰到了何种境界才能发生百年难遇的事? 思绪凌乱了。

c.错标

✗八年级上学期期末考试，地理考不考下学期的内容。

✓八年级上学期期末考试，地理考不考下学期的内容?

✗体育满分，是我的体质变好了，能跑了，还是上大学的女生都颓废了，我五十米第一，排球第一。这是怎么了? 我神灵附体了。

✓体育满分。是我的体质变好了，能跑了，还是上大学的女生都颓废了? 我五十米第一，排球第一，这是怎么了? 我神灵附体了?

✗犯了一个错，需要另外十个错误来掩盖啊，他们小区的监控镜头能证明他在家睡觉，还有这样负责的物管啊。...

✓犯了一个错，需要另外十个错误来掩盖啊。他们小区的监控镜头能证明他在家睡觉? 还有这样负责的物管啊? ...

综合两个领域的错误案例发现，在断句错误中，漏标常见于文本中的并列成分之间，漏标的标点多为顿号。多标常会造成语义内涵错误，多标的符号多是逗号或者句号。而标点符号类

[11]以下省略号并非文本之中的符号，因文本较长，在此省略上下文。

型错误则多见于句号和问号之间，造成疑问和陈述语气混淆。有些自动预测的错误是OOV识别造成的，如例a中的第二组；有些预测错误较为明显，可能是人民日报语料中没有出现类似“考不考”这样的上下文，如例c中的第1组；但更多的预测错误难以分析具体原因，通常需要对句子意义的精细把握与理解。

6 相关工作

国际上标点符号预测或标点符号恢复任务的相关研究主要在语音识别领域。主要是基于机器学习或深度学习的方法，输入数据为听觉信息，文本信息或两者的结合。标点符号预测或任务的目前主流研究可按目标问题分为以下两类：

第一类是将该任务视为序列标注问题(Ueffing et al., 2013; Żelasko et al., 2018)，模型要为每一个位置指定一个标点符号（或无）。一些研究(Lu and Ng, 2010; Ueffing et al., 2013; Hasan et al., 2015)表明条件随机场（CRF）在标点符号预测任务上是比较有效的。近年来，随着神经网络的兴起，Che et al. (2016)首先提出了一种基于卷积神经网络的模型来进行标点预测，结果表明，基于神经网络的方法优于之前基于CRF的方法。Tilk and Alumäe (2015)基于长短时记忆网络(LSTM)及带注意力机制的双向反馈神经网络模型(T-BRNN)进一步提高了标的符号预测的性能。Yi et al. (2017)利用双向LSTM结合CRF模型（BiLSTM-CRF）以及一个其上的集成模型取得了基于序列标注方法目前的最佳性能。

第二类是将其视为单语机器翻译问题，源语言为不含有标点符号的文本，目标语为带标点符号的文本(Peitz et al., 2011; Driesen et al., 2014; Cho et al., 2012)，或目标语为标点符号序列如(Klejch et al., 2016; Klejch et al., 2017)，提出了一个带注意力机制的编码器解码器架构来解决标点符号预测。Kim (2019)提出一种带逐层多头注意力的RNN网络进行标点符号预测，并取得了仅使用词汇特征方法的最好性能。受自注意力机制Vaswani et al. (2017)在NLP任务中有效性，Yi and Tao (2019)提出了一个利用自注意力机制的神经网络模型，可同时在文本和声学的嵌入基础上利用自注意力来获得更好的表示。

还有学者引入其他相关任务来提升标点符号预测的性能，Zhang et al. (2013)提出一种联合句法分析的标点符号预测方法，该方法能利用丰富的句法标注信息，取得了很好的效果。此外，在训练CRF与神经网络时，由于词性信息可以作为有效提升标点符号标注性能的特征Cho et al. (2015)，Yi et al. (2020)提出了一种基于BERT的对抗多任务学习方法，在标点符号预测任务外，额外训练词性标注任务，两者进行对抗，最终在标点符号预测任务上取得了很好的性能。

虽然有少数对越南语(Pham et al., 2019)及中文(Zhao et al., 2012)的相关研究，以及针对中文古文的古文断句研究如黄建年，侯汉清 (2008)；张开旭，夏云庆，宇航 (2009)；王博立，史晓东，苏劲松 (2017)及俞敬松，魏一，张永伟 (2019),。但大多数研究基本都集中在英语上。

绝大多数研究基于IWSLT数据集(Federico et al., 2012)。该数据集语料来源于TED公开演讲，主题十分广泛，转录质量很高。这个数据集经Che et al. (2016)重新组织整理，训练数据集来源于IWSLT2012英文翻译track，约210万个单词，14.4万个文本。开发集约29.6万个单词，2.1万个文本。有两个测试集及Ref和ASR，来源于IWSLT2011，包含约1.3万个单词，860个文本。数据集中有逗号、句号和问号三种标点符号，以及一个非标点标记“O”。

7 结论

本文提出一个领域迁移的标点恢复/标注任务，标注了一个包含问答领域，微博短文本和微博长文本领域的测试集集合，并给定人民日报语料作为验证集。我们给出一个基于预训练语言模型bert的基线模型，并使用focal loss来缓解标签不均衡问题。该模型在人民日报上进行训练并在本数据集上进行了验证。在此基础上，向问答及微博两个领域进行迁移。实验结果表明，向问答领域的迁移效果较好，但是向社交媒体（微博）领域的迁移效果较差，且比源领域下降了20%。我们进一步对模型自动标注的结果进行了分析，发现漏标、多标与类型错误这几种错误的分布较为均衡；从领域比较来看，微博更容易多标，问答更容易漏标。有些自动标注的错误确实需要比较敏感的语感才能辨别。总体来说，跨领域标点符号迁移任务具有一定挑战性，特别是向微博领域迁移，各模型在这个任务上还有较大的提升空间，未来可以利用各种迁移学习或多任务学习的方法来尝试解决。

致谢

本文受北京市自然科学基金资助项目（4192057）资助。感谢匿名评阅人的建议。

参考文献

Doug Beeferman, Adam Berger, and John Lafferty. 1998. Cyberpunc: A lightweight punctuation annotation system for speech. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, volume 2, pages 689–692. IEEE.

Xiaoyin Che, Cheng Wang, Haojin Yang, and Christoph Meinel. 2016. Punctuation prediction for unsegmented transcript based on word vector. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 654–658.

Eunah Cho, Jan Niehues, and Alex Waibel. 2012. Segmentation and punctuation prediction in speech language translation using a monolingual translation system. In *International Workshop on Spoken Language Translation (IWSLT) 2012*.

Eunah Cho, Kevin Kilgour, Jan Niehues, and Alex Waibel. 2015. Combination of nn and crf models for joint detection of punctuation and disfluencies. In *Sixteenth annual conference of the international speech communication association*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Joris Driesen, Alexandra Birch, Simon Grimsey, Saeid Safarfashandi, Juliet Gauthier, Matt Simpson, and Steve Renals. 2014. Automated production of true-cased punctuated subtitles for weather and news broadcasts. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Marcello Federico, Mauro Cettolo, Luisa Bentivogli, Paul Michael, and Stüker Sebastian. 2012. Overview of the iwslt 2012 evaluation campaign. In *IWSLT-International Workshop on Spoken Language Translation*, pages 12–33.

Madina Hasan, Rama Doddipatla, and Thomas Hain. 2015. Noise-matched training of crf based sentence end detection models. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Seokhwan Kim. 2019. Deep recurrent neural networks with layer-wise multi-head attentions for punctuation restoration. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7280–7284. IEEE.

Ondřej Klejch, Peter Bell, and Steve Renals. 2016. Punctuated transcription of multi-genre broadcasts using acoustic and lexical approaches. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 433–440. IEEE.

Ondřej Klejch, Peter Bell, and Steve Renals. 2017. Sequence-to-sequence models for punctuated transcription combining lexical and acoustic features. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5700–5704. IEEE.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal Loss for Dense Object Detection. *arXiv e-prints*, page arXiv:1708.02002, August.

Yang Liu, Elizabeth Shriberg, Andreas Stolcke, Dustin Hillard, Mari Ostendorf, and Mary Harper. 2006. Enriching speech recognition with automatic detection of sentence boundaries and disfluencies. *IEEE Transactions on audio, speech, and language processing*, 14(5):1526–1540.

Xin Liu, Qingcai Chen, Chong Deng, Huajun Zeng, Jing Chen, Dongfang Li, and Buzhou Tang. 2018. LCQMC:a large-scale Chinese question matching corpus. In *Proceedings of the 27th International Conference on Computational Linguistics*, August.

Wei Lu and Hwee Tou Ng. 2010. Better punctuation prediction with dynamic conditional random fields. In *Proceedings of the 2010 conference on empirical methods in natural language processing*, pages 177–186.

Stephan Peitz, Markus Freitag, Arne Mauser, and Hermann Ney. 2011. Modeling punctuation prediction as machine translation. In *International Workshop on Spoken Language Translation (IWSLT) 2011*.

Quang H Pham, Binh T Nguyen, and Nguyen Viet Cuong. 2019. Punctuation prediction for vietnamese texts using conditional random fields. In *Proceedings of the Tenth International Symposium on Information and Communication Technology*, pages 322–327.

Ottokar Tilk and Tanel Alumäe. 2015. Lstm for punctuation restoration in speech transcripts. In *Sixteenth annual conference of the international speech communication association*.

Nicola Ueffing, Maximilian Bisani, and Paul Vozila. 2013. Improved models for automatic punctuation prediction for spoken and written text. In *Interspeech*, pages 3097–3101.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Jiangyan Yi and Jianhua Tao. 2019. Self-attention based model for punctuation prediction using word and speech embeddings. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7270–7274. IEEE.

Jiangyan Yi, Jianhua Tao, Zhengqi Wen, Ya Li, et al. 2017. Distilling knowledge from an ensemble of models for punctuation prediction.

Jiangyan Yi, Jianhua Tao, Ye Bai, Zhengkun Tian, and Cunhang Fan. 2020. Adversarial transfer learning for punctuation restoration. *arXiv preprint arXiv:2004.00248*.

Piotr Żelasko, Piotr Szymański, Jan Mizgajski, Adrian Szymczak, Yishay Carmiel, and Najim Dehak. 2018. Punctuation prediction model for conversational speech. *arXiv preprint arXiv:1807.00543*.

Dongdong Zhang, Shuangzhi Wu, Nan Yang, and Mu Li. 2013. Punctuation prediction with transition-based parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 752–760.

Yanqing Zhao, Chaoyue Wang, and Guohong Fu. 2012. A crf sequence labeling approach to chinese punctuation prediction. In *Proceedings of the 26th Pacific Asia Conference on Language, Information, and Computation*, pages 508–514.

俞敬松，魏一，张永伟. 2019. 基于bert的古文断句研究与应用. 中文信息学报, 33(11):57–63.

古万荣，董守斌，曾之肇，何锦潮，刘崇. 2016. 基于微博用户模型的个性化新闻推荐. 中文信息学报, 30(1):93–100.

张开旭，夏云庆，宇航. 2009. 基于条件随机场的古汉语自动断句与标点方法. 中文信息学报, 49(10):1733–1736.

王博立，史晓东，苏劲松. 2017. 一种基于循环神经网络的古文断句方法. 中文信息学报, 53(2):255–261.

王志宏，过弋. 2019. 微博谣言事件自动检测研究. 中文信息学报, 33(6):132–139.

谢丽星，周明，孙茂松. 2012. 基于层次结构的多策略中文微博情感分析和特征抽取. 中文信息学报, 26(1):73–83.

贺敏，刘玮，刘悦，王丽宏，白硕，程学旗. 2017. 基于特征驱动的微博话题检测方法. 中文信息学报, 31(3):101–107.

黄建年，侯汉清. 2008. 农业古籍断句标点模式研究. 中文信息学报, 22(4):31–38.

多目标情感分类中文数据集构建及分析研究

刘鹏远　　田永胜　　杜成玉　　邱立坤
北京语言大学信息科学学院
国家语言资源监测与研究平面媒体中心
闽江学院计算机与控制工程学院
liupengyuan@pku.edu.cn　blcutys@gmail.com　du_chengyu@163.com　qiulikun@pku.edu.cn

摘要

目标级情感分类任务是要得到句子中特定评价目标的情感倾向。一个评论句中往往存在多个目标，多个目标的情感可能一致，也可能不一致。但在已有针对目标级情感分类的评测数据集中：1）大多数是一个句子一个目标；2）在少数有多个目标的句子中，多个目标情感倾向分布很不均衡，多个目标情感一致的情形占较大优势。数据集本身的缺陷限制了模型针对多个目标进行情感分类的提升空间。针对以上问题，本文构建了一个针对多目标情感分类的中文数据集，人工标注了6339个评价目标，共2071条数据。该数据集：1）评价目标个数分布平衡；2）情感正负极性分布平衡；3）多目标情感倾向分布平衡。随后，本文利用多个目标情感分类的主流模型在该数据集上进行了实验与比较分析。结果表明现有主流模型尚不能对存在多个目标且目标情感倾向性不一致实例中的目标进行很好的分类，尤其是目标的情感倾向为中性时。多目标情感分类任务具有一定的难度与挑战性。

关键词： 目标级情感分类；中文数据集；多目标

Construction and Analysis of Chinese Multi-Target Sentiment Classification Dataset

PengYuan Liu　　YongSheng Tian　　ChengYu Du　　Likun Qiu
Beijing Language and Culture University, School of Information Science
Language Resources Monitoring and Reserch Center
Minjiang University,School of Computer and Control Engineering
liupengyuan@pku.edu.cn　blcutys@gmail.com　du_chengyu@163.com　qiulikun@pku.edu.cn

Abstract

Target-level sentiment classification task is to get the sentiment tendency of a specific evaluation target in a sentence. There are often multiple targets in a comment sentence, and the sentiments of multiple targets may be consistent or inconsistent. However, in the existing evaluation datasets for target-level sentiment classification: 1) most of them are one sentence with one target; 2) in a few sentences with multiple targets, the sentiment distribution of multiple target is very unbanlance, and the situation where the sentiments of multiple targets are consistent has a great advantage. The defect of the dataset itself limits the improvement space of the model for sentiment classification for multiple targets. In response to the above problems, this paper constructs a Chinese dataset for multi-target sentiment classification, manually annotated 6339 targets, a total of 2071 items. The data set: 1) the distribution of the number of evaluation targets is balanced; 2) the distribution of positive and negative sentiments is balanced; 3) the distribution of multi-target sentimental tendency is balanced. Subsequently,

this article uses multiple mainstream models of target-level sentiment classification to conduct experiments and comparative analysis on this dataset. Experimental results show that the existing mainstream models are still unable to well classify the targets in instances where there are multiple targets and the target's sentiment is inconsistent, especially when the target's sentiment is neutral. The task of multi-target sentiment classification is difficult and challenging.

Keywords: Target-level Sentiment Classification , Chinese Dataset , Multi-target

1 引言

社交网络、电子商务和网络新闻的发展迅速，每时每刻都有大量的评论和观点涌现。这些观点和评论文本中包含着非常重要的信息，比如通过分析某个商品的用户评论可以帮助潜在用户选择商品，也可以帮助企业改良商品等，因此情感分析成为自然语言处理领域中最活跃的研究问题之一。方面级别情感分析（Aspect-Level Sentiment Analysis）是一种细粒度的情感分析任务，关注文本针对某一实体、实体的某个部分或属性的情感倾向。目标情感分类（Aspect Term Polarity）是方面级别情感分析的核心子任务之一，目的是分析评论中的目标（Aspect Term）的情感倾向，这个目标是实体的一部分或者是实体的属性，且该目标必须明确出现在句子内。比如（见图1）：“大堂小了点儿，房间挺干净，价钱不错。”这句话中的目标词有“大堂”、“房间”和“价钱”，根据上下文“小了点”、“挺干净”及“不错”可以确定他们的情感倾向分别为负向、正向和正向。当前方面级别的情感分类的研究主要基于深度神经网络，采用端到端的方式进行情感倾向的预测或分类如(Dong et al., 2014; Vo and Zhang, 2015)。而循环神经网络因其在处理序列方面的优势得到了研究者更多的青睐，如(Tang et al., 2015; Ruder et al., 2016)等人的研究。除此之外，注意力机制(Cho et al., 2014)也常常被用来融合方面词与上下文的信息，(Wang et al., 2016a; Ma et al., 2017; Peng et al., 2017; 曾锋；曾碧卿；韩旭丽；张敏；商齐, 2019) 等利用方面词与句子进行交互以得到更好的表示。近年来，也出现了基于预训练语言模型BERT(Devlin et al., 2018)的方面级情感分类研究(Song et al., 2019; 杜成玉；刘鹏远, 2019)。

以上各类模型与方法通常会在同一个目标情感数据集上进行试验及横向比较，目前使用最广泛的目标情感分类数据集是SemEval-2014 task4(Pontiki et al., 2014)和Twitter(Jiang et al., 2011)，均为英文数据集。我们对这两个数据集中的评价目标和对应的情感倾向进行统计并发现：1）Twitter数据集及SemEval-2014 task4数据集中中含有一个以内评价目标的句子比例很高（分别为100%及73.6%）；2）SemEval-2014 task4数据集中，评价目标情感倾向不一致的句子占比仅有8.6%。而在实际应用中，一个句子包含一个以上目标词且评价倾向性不同的情况比较常见。但是，由于评测数据集中的多目标实例较少，情感不一致的实例更少，这种分布对现有模型在多目标句子上的目标情感分类评价造成一定困难，也限制了模型针对多个目标句子进行目标情感分类的提升空间。

为解决以上问题，本文构建了一个针对多目标情感分类的中文数据集[0]，人工标注了6339个评价目标，共2071条数据。该数据集：1）评价目标个数分布相对平衡；2）情感正负极分布相对平衡；3）多目标情感倾向分布相对平衡。随后，本文利用多个目标情感分类的主流模型在该数据集上进行了实验，比较了各个模型针对多目标情感分类的表现，并进行了详细分析与讨论。

2 数据集构建

2.1 数据准备

本文选取了谭松波收集整理的酒店评论语料料[1]作为原始语。该语料规模为10000篇，内

基金项目：北京市自然科学基金资助项目（4192057）资助

[0]https://github.com/NLPBLCU/Chinese-Multi-Target-Sentiment-Classification-Dataset
[1]https://languageresources.github.io/

```
<sentence id="2">
    <text>大堂小了点，房间挺干净，价钱不错。</text>
    <aspectTerms>
        <aspectTerm term="大堂" polarity="negative"  from="0" to="2"/>
        <aspectTerm term="房间" polarity="positive"  from="6" to="8"/>
        <aspectTerm term="价钱" polarity="positive"  from="12" to="14"/>
    </aspectTerms>
</sentence>
```

Figure 1: XML格式目标情感分类示例

容为携程网的评论。我们将原始语料进行去重并以此为待标注对象，共得到7767篇评论，其中5323篇是正向评论，2444篇是负向评论。

2.2 标注对象

依照目标情感分类任务研究的惯例，我们舍弃目标评价倾向为冲突的句子，仅标注为正向、负向与中性的评价句。

在语料选择方面，选取含有2个或以上目标的句子为标注对象，并且尽量控制标注数据在三种情况下的分布同时基本平衡：1）句中目标数量；2）目标情感正负倾向极性；3）句中目标情感相同或不同。

具体标注时，将每个评论中的目标词抽离出来，分别标注每一个目标词在其中的情感倾向，然后得到该句子的标注结果如："大堂"：负向；"房间"：正向；"价钱"：正向。

2.3 标注流程、数据格式与标注规范

由三名语言学及应用语言学专业的硕士生担任标注员进行标注，先进行一轮试标注，并进行讨论，在此基础上总结出标注规范。然后依据标注规范，由两名标注员独立地进行标注，对于标注不一致的情况由第三位标注员进行仲裁。

试标注。我们从待标注对象中随机抽取100条，按照标注程序进行试标注。三名标注员在标注了所有100条语料后，就标注不一致的数据进行讨论，总结并形成了标注规范[2]。

数据格式。数据用XML格式存储，如图1所示。其中：$< aspectTerms >$是目标及其情感倾向的标签，$< term >$表示目标，$< polarity >$表示情感倾向，$< from >$和$< to >$表示目标词开始位置和结束位置。

标注规范

（1）目标词：目标词是明确出现在句子中的被评价对象的具体属性，本节构建的是酒店领域的数据集，因此被评价对象是"酒店"，目标词可能是"装修风格"、"服务态度"等。只标注具有多个目标词的句子。

（2）情感倾向：包含正向、负向、中性三种情况。正向评价是评价者对某个目标词持积极的、满意的态度。负向评价是评价者对某个目标词持消极的、不满的态度。中性评价是评价者对某个目标词持中立的、客观的态度。

（3）标注单位：参照现有的英文目标情感分类数据集，本节以单个句子为单位进行标注。

（4）标注边界：只标注目标词，目标词前的形容词性修饰成分及数量短语等不在标注范围内。

（5）目标词包含名词型和动词型两种。

（6）目标词若出现多次，只标注离评价词最近的目标词。

2.4 标注结果

我们仅标注目标词为2的实例与目标词大于等于3的实例，最终标注了2071条数据，共6339个目标，平均每个句子3.06个目标。标注好的数据集基本情况如表1，2所示。句中目标词情感极性一致与不一致句子，目标词为正向情感与负向情感三者比例分别平衡。数据集整体标注一致率为78.1%。

[2]详细规范与示例将随数据集及代码一并发布。

目标词数量或情感倾向	数量	占比%
句子中目标词数量：2	954	46.1
句子中目标词数量：⩾3	1157	55.9
句子目标词情感极性一致	1009	48.7
句子目标词情感极性不一致	1062	51.3

Table 1: 目标词数量或情感倾向分布

	正向情感	负向情感	中性情感	总计
目标词数量	3001	2827	511	6339
目标词比例	47.3%	44.6%	8.1%	100%

Table 2: 所有目标词的情感倾向分布

3 模型与实验

3.1 模型

为探索和分析目标情感分类的主流方法在本文构建数据集上的表现，我们选择了5个具有代表性且已开源的主流神经网络模型，其中包括2个基于BERT的目标情感分类模型。我们还实现了1个基于BERT的基线模型BERT-SPC。

IAN(Ma et al., 2017):将上下文词与目标词通过LSTM层得到隐藏层状态序列，接着利用池化函数得到目标词的初始向量表示，该向量与上下文隐藏层状态通过注意力层得到上下文词注意力权重分布，接着计算加权后的上下文表示，将它最终的上下文向量。然后用类似的方法得到目标词表示，再与上下文表示拼接。

RAM(Peng et al., 2017)：首先通过双向LSTM层得到句子的隐藏层状态序列，接着利用位置信息构建位置加权记忆矩阵，然后构建多个注意力层，每一层的结果都是基于上一层的结果重新进行计算，以此来捕捉记忆矩阵中有用的信息。

ATAE-LSTM (Wang et al., 2016b)：对句子和给定的方面词用LSTM进行编码后，采用注意力机制对隐藏层输出进行处理，将得到的注意力向量与方面词向量拼接得到关于方面词的情感极性表达。

AEN-BERT(Song et al., 2019)：运用标签平滑化的方法来解决中性类别方面词情感模糊的问题，并运用了多个不同注意力机制对上下文和方面词进行建模。

BERT-HAN(杜成玉；刘鹏远, 2019)：建立于BERT上的基于螺旋注意力机制的神经网络模型。首先利用目标词构建句子，接着采用句子对的输入方式利用BERT预训练词向量，然后利用螺旋上下文注意力层和螺旋目标词注意力层通过多次叠加注意力层来更好地表示上下文和目标词。

BERT-SPC：使用预先训练好的BERT来生成序列的词向量，BERT有单个句子和句子对两种输入方式。本文采用句子对的输入方式，将目标词与上下文组成句子对进行输入，输入方式为“[CLS]+ target+[SEP]+context+[SEP]”，然后将得到向量送入softmax分类器。

3.2 实验

3.2.1 数据集与评价指标

数据集采用本文建立的多目标情感分类中文数据集，其中含有多个评价目标的句子共6339条。按照3:1的比例分别将两个数据集划分成训练集和测试集，具体划分见表3。评价指标采用分类准确率，即模型正确分类的样本数与模型总样本数之比。

数据集	正向目标数量/占比(%)	负面目标数量/占比(%)	中性目标数量/占比(%)	总计
训练集	2250/47.3	2120/44.5	383/8.1	4753
测试集	751/47.2	707/44.6	128/8.1	1586

Table 3: 多目标情感分类中文数据集详细信息

模型	词向量维度	隐状态维度	学习率	batch size
IAN	300	300	1e-3	16
RAM	300	300	1e-3	16
ATAE-LSTM	300	300	1e-3	16
AEN-BERT	768	N/A	2.00e-05	16
BERT-HAN	768	300	2.00e-05	16
BERT-SPC	768	N/A	2.00e-05	16

Table 4: 模型参数设置

	IAN	RAM	ATAE-lSTM	AEN-BERT	BERT-HAN	BERT-SPC
准确率(%)	74.1	82.7	79.3	75.1	81.5	**86.2**

Table 5: 模型的准确率

3.2.2 参数设置

表4是各模型实验时的参数设置情况。其中，IAN、RAM采用Stanford大学发布的GloVe词向量[3]来作为预训练词向量；BERT-SPC、AEN-BERT、BERT-HAN采用BERT BASE[4]进行预训练。

3.2.3 实验结果

实验结果如表5所示。在6个模型中，基线模型BERT-SPC表现最好，IAN表现最差。仅将目标词与实例一起输入并进行训练的BERT-SPC模型，就已经能够学到很好的目标词情感倾向信息。而其他两种进行目标词与句子进行不同注意力权重计算的模型：AEN-BERT与BERT-HAN的表现反而不如BERT-SPC。值得注意的是，两个非BERT模型RAM与ATAE-LSTM的表现分别比两个基于BERT的模型BERT-HAN与AEN-BERT要好，其可能的原因，我们将在后续分析中进一步尝试探究。

4 讨论

4.1 目标不同情感倾向对模型分类性能的影响

我们将所有模型对目标分别为正、负及中性三种情感倾向分类时的性能进行对比，结果如表6所示。其中正向及中性情感倾向最优模型为BERT-SPC，负向为BERT-HAN。所有模型在目标情感倾向不同时性能从好到坏的排序均为：目标为正向情感倾向>目标为负向情感倾向>目标为中性情感倾向。当目标情感为中性时，各个模型的表现均不尽人意。这一点很大程度上与数据集中目标情感倾向的分布有关（分布见本文第2小节中的表2）。

图1是将数据集中三种情感倾向目标的分布作为待比较的分布基准，考察所有模型性能提升绝对值的柱状图。可以看出，所有模型在目标为正/负向情感倾向时，性能较分布基准均有所提升，且提升幅度较大。目标为中性情感时，三种基于BERT的模型，不但性能较分布基准均有所提升且幅度较大（BERT-SPC提升幅度最大），这说明基于BERT的模型能在数据分布不均衡的条件下，学到一定的中性倾向的信息；而对其他三种非BERT模型，性能均低于分布基准，说明这几个模型所学到的目标中性情感倾向信息较少，甚至基本学不到（对IAN模型）。

基于BERT的模型能够比非BERT模型更好地融合中性情感目标数据的信息，我们推测可能会在一定程度上影响其在目标为正/负向情感时的性能，这可能是造成AEN-BERT与BERT-BERT在总体性能上没有RAM模型好的原因之一。

4.2 目标数对应模型分类性能的影响

为考察各模型在不同目标个数情况下的性能，根据数据集中每句含有的目标个数，分两类对各模型性能进行统计：1）含两个目标；2）含有三个目标及以上。

[3]https://nlp.stanford.edu/projects/glove/
[4]https://github.com/google-research/bert

模型	正向情感倾向	负向情感倾向	中性情感倾向
IAN	82.3	78.9	0.0
RAM	89.1	89.7	6.2
ATAE-LSTM	85.1	87.0	3.1
AEN-BERT	80.2	79.9	22.7
BERT-HAN	82.7	**92.1**	27.7
BERT-SPC	**91.3**	89.3	**39.1**

Table 6: 模型在不同情感倾向上的分类性能（准确率%）

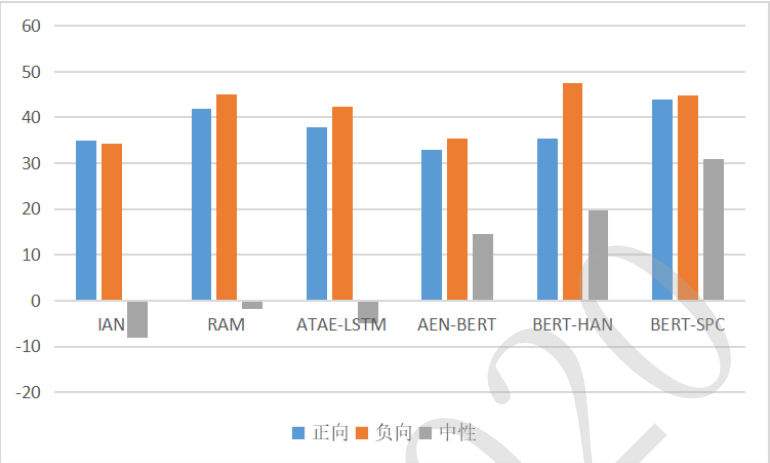


Figure 2: 各个模型分别在目标为三种情感倾向时性能较分布基准提升绝对值的柱状图（准确率%）

此外，我们还在本文第二节介绍的待标注对象语料中，按照本文建设多目标情感分类数据集基本类似的过程，额外标注了仅含有单个评价目标的例句共1046条，其情感倾向性分布与本文建立的多目标情感分类数据集基本一致。类似的，按照3:1的比例划分成训练集和测试集，分别为784条和262条。在这个单目标数据集上，所有模型重新进行了实验，参数设置与之前多目标实验相同（见表4）。

将各模型在单目标数据集及多目标数据集上的实验结果合并列于表7。其中，目标词数量为1，代表单目标数据集，其余两行代表在多目标数据集上的结果。

我们发现，虽然直觉上多目标数据相对于单目标数据更加难以分类，但在单目标数据集上的模型的性能并非均比在多目标数据集上的性能更高。同时，目标词数量分别在1,2及大于等于3时，模型的分类性能并没有较大的差距。

我们统计了在多目标数据集中，同一个条目多个目标情感一致与不一致时各个模型分类性能的表现，如表8所示。可以看出，在多个目标情感一致时，各个模型的性能均非常优异（大于90%），AEN-BERT的表现最好，其次是BERT-SPC。各个模型在多目标情感倾向不一致时，BERT-SPC的表现最好，能够达到80%，AEN-BERT表现最差，还不到60%。所有模型的性能比情感倾向一致时的性能均有大幅度的下降，降幅最大的是AEN-BERT，接近40%，降幅最小的是BERT-SPC，也达到了14.4%。

目标词数量	IAN	RAM	ATAE-lSTM	AEN-BERT	BERT-HAN	BERT-SPC
1	**81.1**	**83.7**	80.5	75.3	**82.1**	**89.3**
2	75.7	**83.7**	**80.9**	**79.0**	81.6	87.6
⩾3	73.4	82.7	79.6	73.84	81.7	85.6

Table 7: 不同目标数时模型的分类性能（准确率%）

多目标情感	IAN	RAM	ATAE-lSTM	AEN-BERT	BERT-HAN	BERT-SPC
一致	92.8	90.4	90.7	**98.1**	92.0	94.4
不一致	60.2	73.6	70.8	58.4	69.4	**80.0**
下降幅度	-32.6	-16.8	-19.9	**-39.7**	-22.6	-14.4

Table 8: 模型在多目标情感倾向一致和不一致数据上的分类性能（准确率%）

4.3 模型分类结果的相关性

图3是6个模型在本数据集上的分类结果相关性热力图，它反映的是各个模型在数据集上的预测结果的相关性。如果两个模型预测结果的相关性比较高，则一个模型预测正确时，另一个模型预测也较大可能是正确的；一个模型预测错误时，另一个模型也较大可能是预测错误的。

从图3可知，6个模型的分类结果相关性都比较高，基本都在80%以上，其中ATAE-LSTM与RAM的相关性及BERT-HAN与RAM的相关性相对较高，超过了85%。

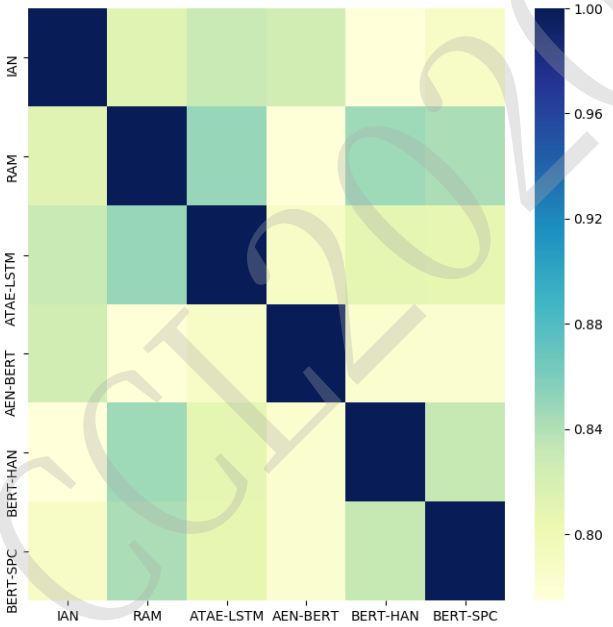


Figure 3: 模型分类结果的相关性

我们还统计了BERT-SPC模型预测错误（共219个）但其他模型能够预测正确的目标个数及比例（即与BERT-SPC模型预测错误目标个数之比），列在表9中。可见，虽然模型分类结果相关性比较高，但对BERT-SPC模型错误预测的目标，其他模型也有一定正确预测的可能，但是有部分目标（219-133=86个）所有模型均无法正确预测。

4.4 目标分类难度

数据集中目标的情感倾向可能被1或多个模型正确预测，也可能无法被任何模型正确预测。可从正确预测目标情感倾向模型个数的角度来考察目标情感倾向的预测难度。我们将各个模型对测试集中所有1586个目标实例的预测结果进行了逐个统计，对每一个目标进行模型预测正确计数，即每有一个模型对其预测正确，则该目标模型预测正确个数加一。所有目标分成从0到6共7类。在此基础上，我们进一步将所有目标分为易、中、难三个等级，结果列在表10：

易：模型预测正确个数为5及6的目标；

	IAN	RAM	ATAE-lSTM	AEN-BERT	BERT-HAN	总计
个数	61	89	84	78	86	133
比例%	27.6	40.6	38.4	35.6	39.3	60.7

Table 9: BERT-SPC模型预测错误但其他模型能够预测正确的目标个数及比例（与BERT-SPC模型预测错误目标个数之比）。其中总计是针对BERT-SPC模型预测错误而其他模型预测正确并去重后的目标总和及相应比例。

模型计数	0	1	2	3	4	5	6
目标个数	86	50	73	105	153	236	883
所占比例%	5.4	3.1	4.6	6.6	9.6	14.9	55.7
难度等级	难			中		易	
目标个数	136			331		1179	
所占比例%	8.6			20.9		70.5	

Table 10: 目标情感倾向预测难度等级分布

中：模型预测正确个数为2,3,4的目标；
难：模型预测正确个数为0,1的目标。

由表10可知，有86个目标，所有模型均没有预测正确，有883个目标所有模型均预测正确。难度等级为“难”、“中”及“易”的目标，分别占所有目标的8.6%，20.9%及70.5%。

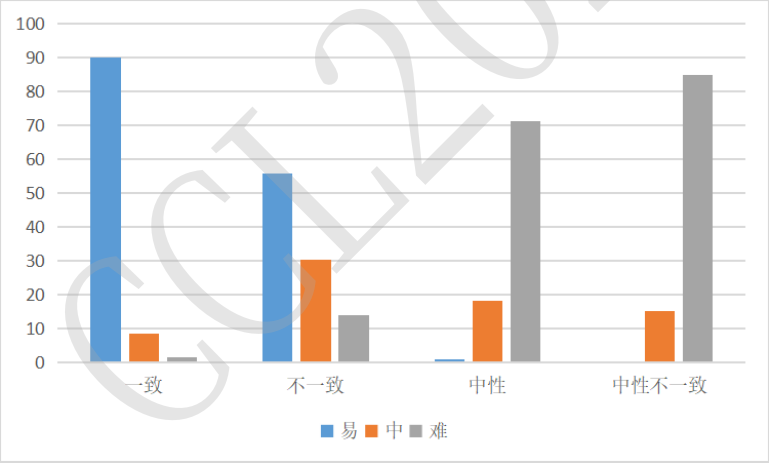


Figure 4: 目标“一致”、“不一致”，“中性”、“中性不一致”时的难度分布

在目标难度等级的基础上，我们考虑了以下四种典型情况：
1）**一致**：处在有情感倾向一致目标的句中的目标；
2）**不一致**：处在有情感倾向不一致目标的句中的目标；
3）**中性**：中性情感目标；
4）**中性不一致**：中性情感目标，且处在有情感倾向不一致目标的句中。

我们绘制了这四种典型情况下难度分布柱状图，见图4。由图4可知，当目标在“一致”时，90%的目标在等级“易”中，即容易分类，“不一致”时，仅有50%左右的目标容易分类。当目标为“中性”时，超过70%的目标在等级“难”中，即难以分类，而当目标为“中性不一致”时，有约85%的目标难以分类。这四种典型情况，目标分类难度从难到易依次为：“中性不一致”，“中性”，“不一致”，“一致”。

4.5 实例分析

为对模型难以分类的实例有一定感性认识，我们对全部模型均预测错误的目标例句进行

了仔细观察，并列举了部分在表11。表11中，其中，第一列为样例，红色字体为目标，红色字体后括号内的数字为该目标的情感倾向，1/-1/0分别表示正向/负向/中性；黑体词为本句要预测情感倾向的目标词。第二列为要预测情感倾向的目标，第三列为该目标类型，“一致”/“不一致”与4.4小节中考虑的前两种典型情况相同。第四列为要判断的目标词的真实情感倾向。最后一列是6个模型分别对目标词情感倾向的预测结果。

我们发现目标词以外的情感倾向会对目标词的情感倾向抽取造成干扰。如在句子“其实格林豪泰...还挺好，接着就是前台了，那叫什么态度啊？”中，目标词为“前台”，其情感极性很明显为负向。但是所有的模型都把他分到了正向，这可能是因为，在这个句子中短语“还挺好”离目标词更近，且情感倾向为正，但这是对目标词“大门”的描述，因此对模型的分类造成了干扰。在中性目标词的情感倾向分类上，由于目标词没有明显情感倾向，则其余目标词的情感倾向造成的干扰将会更加明显。此外，中性目标词的情感倾向含糊不清，程度较弱，不太容易判断他的情感倾向是否是中性。例如在句子“第二次入住了...冰镇饮料喝。”中，尽管目标词“冰镇饮料”的情感倾向为中性，但是句子中修饰目标词的“免费”很多情境下是表达正向的情感。在数据规模上，由于中性目标数量远远低于正向与负向目标数，故也使得模型对于中性目标的情感分类泛化能力较弱。

样例	目标	类型	情感	模型预测
其实格林豪泰给我的印象一直挺好的，（那是因为之前住的是上海的格林豪泰），所以就想换换环境，订了三天的房，首先一进大门（1），感觉还挺好，接着就是**前台**了，那叫什么服务态度（-1）呀?	前台	不一致	-1	1 1 1 1 1 1
但是在房间安排（-1）方面觉得有点欠妥，我订了两间高级房，一样的价格，但不一样的房型，一件[5]的漱洗间（-1）非常小，而且没有**阳台**，另一间的漱洗间却是这间的两倍，而且有阳台。	阳台	不一致	1	-1 -1 -1 -1 -1 -1
酒店早餐（-1）不是很丰富，房间设施（1）尚可，标准房无**矿泉水**送。	矿泉水	不一致	0	1 -1 1 1 -1 -1
第二次入住了，酒店有免费接机的服务（0），坐到车上后还有免费的**冰镇饮料**喝。	冰镇饮料	一致	0	1 1 1 1 1 1

Table 11: 全部模型均错误预测的部分实例。其中，第一列为样例，红色字体为目标，红色字体后括号内的数字为该目标的情感倾向，1/-1/0分别表示正向/负向/中性；黑体词为本句要预测情感倾向的目标词。第二列为要预测情感倾向的目标，第三列为该目标类型，“一致”/“不一致”与4.4小节中考虑的前两种典型情况相同。第四列为要判断的目标词的真实情感倾向。最后一列是6个模型分别对目标词情感倾向的预测结果。

5 相关工作

传统的基于方面词的情感分析方法包括基于规则的方法(Ding et al., 2008)和基于统计的方法(Jiang et al., 2011; Zhao et al., 2010)。这些方法侧重于将一组分类线索转化为特征向量，但这既需要费力的特征工程工作，也需要大量的额外语言资源。循环神经网络较早应用到方面级别情感分类领域(Tang et al., 2015; Ruder et al., 2016)。单纯基于RNN 的模型无法很好地捕捉到句子中与方面词与情感极性词或短语之间的关联，研究人员引入注意力机制来解决这个问题。Wang et al. (2016a)对句子和给定的方面词用LSTM进行编码后，采用注意力机制对隐藏层输出进行处理，得到关于方面词的情感极性表达。Tang et al. (2016)基于输入句子的词向量构成的外部记忆进行注意力学习，模型的每一层基于上一层输出的结果重新计算注意力分布，最终得到关于给定方面词的情感极性表达。Ma et al. (2017)不仅计算句子隐藏层输出的注意力分布，还计算方面词的注意力分布。Huang and Carley (2018)以联合的方式建模方面词和句子，

[5]因为是用户撰写的评论，因此存在一些用户自行输入的错误，为保证数据的真实性，本文未进行任何更正。

明确捕捉方面词和上下文句子之间的交互。Li et al. (2018)将位置嵌入作为输入的一部分，并用层次注意力机制来融合目标和上下文词的信息。卷积神经网络能够并行计算，在运算速度上有一定优势，于是也有学者基于参数化卷积神经网络(Huang and Carley, 2018)和基于门控卷积神经网络(Xue and Li, 2018)的相关研究。BERT提出后，一些研究在它基础上对上下文进行编码，并结合注意力机制，来更好地解决方面级别情感分类任务。Song et al. (2019)在BERT表示的基础上，采用多个不同注意力机制对上下文和方面词进行建模。类似的工作还有(Zhao et al., 2020)。杜成玉；刘鹏远 (2019)利用螺旋注意力机制，反复增强BERT编码后的方面词与句子的表示。

现有的方面级别情感分类任务的数据集主要有：SemEval-2014 task 4 Aspect Based Sentiment Analysis(Pontiki et al., 2014)、SemEval-2015 task 12 Aspect Based Sentiment Analysis(Pontiki et al., 2015)、SemEval-2016 task 5 Aspect Based Sentiment Analysis(Pontiki et al., 2016)和Twitter(Jiang et al., 2011)。

SemEval-2014 task4数据集包含两个领域，分别为laptop和restaurant。Laptop数据集摘自笔记本电脑的用户评论，包含3048个英语句子；Restaurant数据集来自于Ganu et al. (2009)标注的餐厅评论，由3044个英语句子组成。两个领域的数据集都以句子为单位人工标注了句子中的方面词及其情感倾向和位置，其中情感倾向包含正向、负向、中性，除此之外，Restaurant数据集还标注了方面词的类别的情感倾向。SemEval-2015 task 12数据集采用的原始语料与SemEval-2014 task4数据集相同，但它是以一条评论为单位进行标注的，两个领域的数据集都标注了方面词的类别及其情感倾向，但与SemEval-2014 task4数据集不同的是，它是实体和属性对，其中实体和属性属于提前规定好的实体和属性集合；Restaurant数据集还标注了观点目标词及其情感倾向及位置，它与SemEval-2014 task4数据集中的方面词的概念相同。SemEval-2016 task 5数据集的标注内容与SemEval-2015 task 12数据集相同，增加了其他语种的数据集，比如中文、法语、阿拉伯语等。Twitter数据集是使用关键字通过twitter API收集的开放域数据集，关键字包含名人、公司和产品的名称等，然后以tweet为标注单位，人工标注了tweet中出现的关键字及其情感倾向，这是目前为止人工标注的最大的针对方面情感分类任务的Twitter数据集。

6 结论

本文针对现有数据集的问题构建了一个针对多目标情感分类的中文数据集，该数据集评价目标个数、情感正负极性及多目标情感倾向均分布平衡。我们还实现了多个目标情感分类的主流模型并在该数据集上进行了实验与比较分析。结果表明：1）目标个数对各模型在数据集上的分类结果影响不大；2）在同一句中多个目标情感倾向是否一致对模型的影响较大；3）情感倾向为中性的实例较难进行预测，一方面是由于中性目标实例较少，另一方面是因为中性情感倾向的强度一般较低。多目标情感分类的模型应考虑如何对目标情感倾向性不一致，尤其是目标情感倾向性不一致且同时目标的情感倾向有中性情感的情况下，进行有针对性的改进。

致谢

本文受北京市自然科学基金资助项目（4192057）资助。感谢匿名评阅人的建议。

参考文献

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Xiaowen Ding, Bing Liu, and Philip S. Yu. 2008. A holistic lexicon-based approach to opinion mining. In Marc Najork, Andrei Z. Broder, and Soumen Chakrabarti, editors, *Proceedings of the International Conference on Web Search and Web Data Mining, WSDM 2008, Palo Alto, California, USA, February 11-12, 2008*, pages 231–240. ACM.

Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive recursive neural network for target-dependent twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 49–54, Baltimore, Maryland, June. Association for Computational Linguistics.

Gayatree Ganu, Noemie Elhadad, and Amélie Marian. 2009. Beyond the stars: Improving rating predictions using review text content. In *12th International Workshop on the Web and Databases, WebDB 2009, Providence, Rhode Island, USA, June 28, 2009*.

Binxuan Huang and Kathleen M. Carley. 2018. Parameterized convolutional neural networks for aspect level sentiment classification. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1091–1096. Association for Computational Linguistics.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea, editors, *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 151–160. The Association for Computer Linguistics.

Lishuang Li, Yang Liu, and Anqiao Zhou. 2018. Hierarchical attention based position-aware network for aspect-level sentiment analysis. In Anna Korhonen and Ivan Titov, editors, *Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018, Brussels, Belgium, October 31 - November 1, 2018*, pages 181–189. Association for Computational Linguistics.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Twenty-Sixth International Joint Conference on Artificial Intelligence*.

Chen Peng, Zhongqian Sun, Lidong Bing, and Yang Wei. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In Preslav Nakov and Torsten Zesch, editors, *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014*, pages 27–35. The Association for Computer Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In Daniel M. Cer, David Jurgens, Preslav Nakov, and Torsten Zesch, editors, *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2015, Denver, Colorado, USA, June 4-5, 2015*, pages 486–495. The Association for Computer Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, San Diego, California, June. Association for Computational Linguistics.

Sebastian Ruder, Parsa Ghaffari, and John G Breslin. 2016. A hierarchical model of reviews for aspect-based sentiment analysis.

Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Attentional encoder network for targeted sentiment classification. *CoRR*, abs/1902.09314.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2015. Effective lstms for target-dependent sentiment classification. *Computer Science*.

Duyu Tang, Bing Qin, and Ting Liu. 2016. Aspect level sentiment classification with deep memory network. In Jian Su, Xavier Carreras, and Kevin Duh, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 214–224. The Association for Computational Linguistics.

Duy Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *International Conference on Artificial Intelligence*.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Zhao Li. 2016a. Attention-based lstm for aspect-level sentiment classification. In *Conference on Empirical Methods in Natural Language Processing*.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016b. Attention-based LSTM for aspect-level sentiment classification. In Jian Su, Xavier Carreras, and Kevin Duh, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 606–615. The Association for Computational Linguistics.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2514–2523. Association for Computational Linguistics.

Wayne Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. 2010. Jointly modeling aspects and opinions with a maxent-lda hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, EMNLP 2010, 9-11 October 2010, MIT Stata Center, Massachusetts, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 56–65. ACL.

Pinlong Zhao, Linlin Hou, and Ou Wu. 2020. Modeling sentiment dependencies with graph convolutional networks for aspect-level sentiment classification. *Knowl. Based Syst.*, 193:105443.

曾锋；曾碧卿；韩旭丽；张敏；商齐. 2019. 基于双层注意力循环神经网络的方面级情感分析. 中文信息学报, 33(6):108.

杜成玉；刘鹏远. 2019. 基于螺旋注意力网络的方面级别情感分析模型. In 第十八届全国计算语言学学术会议.

基于Self-Attention的句法感知汉语框架语义角色标注*

王晓晖[1,¶], 李茹[1,2,‡,†], 王智强[3,‡], 柴清华[4,‡], 韩孝奇[1,¶]
[1]计算机与信息技术学院,山西大学
[2]计算智能与中文信息处理教育部重点实验室,山西大学
[3]智能信息处理研究所,山西大学
[4]外国语学院,山西大学
[¶]{1640051575,419432571}@qq.com [‡]{liru,wangzq,charles}@sxu.edu.cn

摘要

框架语义角色标注（Frame Semantic Role Labeling, FSRL）是基于FrameNet标注体系的语义分析任务。语义角色标注通常对句法有很强的依赖性，目前的语义角色标注模型大多基于双向长短时记忆网络Bi-LSTM，虽然可以获取句子中的长距离依赖信息，但无法很好获取句子中的句法信息。因此，引入self-attention机制来捕获句子中每个词的句法信息。实验结果表明，该模型在CFN（Chinese FrameNet，汉语框架网）数据集上的F1达到83.77%，提升了近11%。

关键词： 语义角色标注 ； self-attention机制 ； Bi-LSTM ； 汉语框架网

Syntax-Aware Chinese Frame Semantic Role Labeling Based on Self-Attention

Xiaohui Wang[1,¶], Ru Li[1,2,‡], Zhiqiang Wang[3,‡], Qinghua Chai[4,‡], Xiaoqi Han[1,‡]
[1]School of Computer and Information Technology,Shanxi University
[2]Key Laboratory of Ministry of Education for Computational Intelligence and Chinese Information Processing,Shanxi University
[3]Institute of Intelligent Information Processing,Shanxi University
[4]School of Foreign Languages,Shanxi University
[¶]{1640051575,419432571}@qq.com [‡]{liru,wangzq,charles}@sxu.edu.cn

Abstract

Frame semantic role labeling is a semantic analysis task based on the FrameNet. Semantic role labeling usually relies heavily on syntax. Most of the existing semantic role labeling models are based on Bi-LSTM. Although they can obtain the long-distance dependency information in sentences, they can not get the syntactic information well. Therefore, the self attention mechanism is introduced to capture the syntactic information of each word in the sentence. Experimental results show that the F1 of our model on Chinese FrameNet dataset reaches 83.77%, which is increased by nearly 11%.

Keywords: Semantic role labeling , Self-Attention , Bi-LSTM , Chinese FrameNet

1 引言

框架语义角色标注的主要任务是识别出句子中给定目标词所激起框架对应的框架元素，这些框架元素被赋予特定的含义，如施动者、受动者时间、地点等。简单来说，就是“谁”在“什

* 基金项目：国家自然科学基金(No.61772324;No.61936012)
† 通讯作者

么时间”、“什么地点”对“谁”做了“什么”。图1为汉语框架语义角色标注示例，其中，例句下面为句法依赖标签，例句上面的为语义角色标签。例句的目标词为“参观”，该目标词激起的为框架为“拜访”，该框架有两个核心框架元素agt（施动者）和ent（实体），分别对应例句中的“我们”和“吴店镇”，而例句中的“实地”则标为manr（方式）。语义角色标注的应用非常广泛，如在信息抽取 (Surdeanu et al., 2003)、问答系统 (Yih et al., 2003)等领域取得了一定研究成果。

传统的语义角色标注方法多采用机器学习与特征工程相结合的方法。在这类方法中，通常依赖于人工抽取的特征，并且会带来模型复杂、特征稀疏等问题 (Zhang et al., 2019)。语义角色标注对句法有着较强的依赖性。如图1所示，对于目标词“参观”来说，通常其主语都被标注为“施动者”，宾语被标注为“实体”，而“方式”则通常为副词或介词短语作状语。因此，句法信息在一定程度上有助于语义角色标注。

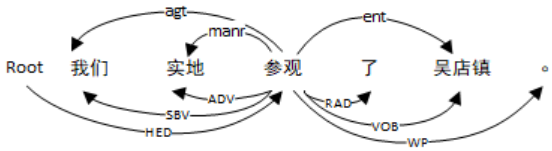


图 1. 汉语框架语义角色标注示例

近年来，深度学习方法逐渐被应用到自然语言处理任务中，它能自动学习文本中的特征，从而大大减少了特征工程中的工作量。特别地，(Zhou and Xu, 2015; He et al., 2017; Marcheggiani et al., 2017)提出了无句法的语义角色标注模型，并且取得了较好的效果。这似乎和句法信息是高性能语义角色标注的前提条件的观点相冲突。然而，句法信息被认为与语义关系密切，在语义角色标注任务中起着至关重要的作用 (Punyakanok et al., 2008)。(Marcheggiani and Titov, 2017)提出了图卷积网络模型，以相对较好的句法分析器作为输入，进一步提高了语义角色标注的性能。

由于Bi-LSTM可以有效地获取句子中的长距离依赖信息，在序列标注任务中有着天然优势。因此，现有的语义角色标注模型大多基于Bi-LSTM模型。但Bi-LSTM却无法很好的获取句子的句法信息，而近年来同样在自然语言处理领域广泛应用的self-attention机制却可以处理这个问题。因此，本文在Bi-LSTM的基础上引入self-attention机制。同时，在序列标注任务中标签之间是有依赖关系的，如在BIO标注模式中，标签I应出现在标签B之后，而不应该出现在O之后，所以本文利用CRF进行全局标签优化预测出最优标签序列。

2 相关工作

语义角色标注是由 (Gildea and Jurafsky, 2002)提出的，同时在人工标注的FrameNet语料上提出了基于统计的分类器。已有的汉语语义角色标注方法有两类，分别是基于特征工程的方法和基于神经网络的方法。

在早期的语义角色标注工作中，大多研究都致力于特征工程。(Gildea and Jurafsky, 2002)在没有大规模语义标注语料库的情况下对汉语语义角色标注进行了初步的研究，取得了很好的效果。(Xue, 2008)在中文PropBank（CPB）上将最大熵分类器和特征工程相结合，同时将标准的句法分析和自动句法分析分别加入到特征工程中，实验表明汉语句法分析的性能是实现高效语义角色标注的关键；(Li et al., 2010)基于CRF在CFN数据集上进行语义角色标注研究，分别取得了63.65%和61.62%的F1值；(Wang, 2010)基于最大熵模型分别使用词层面和块层面特征实现了的汉语框架语义角色标注；(Tu et al., 2016)提出了一种基于主动学习的方法，当数据规模相同时，实验结果最高提升了4.83%，同时，达到同等F1值时最高可减少30%的人工标注量。

随着深度神经网络模型在自然语言处理领域的诸多任务上得到成功应用，一系列基于神经网络的语义角色标注模型被提出。(Ronan and Jason, 2008)等提出了使用单一的卷积神经网络结构进行包括语义角色标注任务在内的多任务学习模型，整个网络共享权重，代表了一种新的共享任务半监督学习形式；(Wang et al., 2014)将予语义角色标注分为角色识别和角色分类两个步骤，利用分层输出的神经网络模型取得了64.19%的F1值。(Wang et al., 2015)提出了基于异构数据的Bi-LSTM模型，可以更方便地缓解单个标注语料库的可扩展性问题，

在CPB数据集上取得了77.59%的F1值； (Dang, 2015)基于词分布的汉语框架语义角色标注模型实现了72.89%的F1值； (Wang et al., 2017)基于分布式表示，提出了一种多特征融合的神经网络结构，同时使用Dropout正则化技术有效地缓解了模型的过拟合现象，使得模型的F1值提高了近7%。

最近，人们尝试构建基于span的无句法输入的端到端语义角色标注模型 (Zhou and Xu, 2015; He et al., 2017; Tan et al., 2018; Ouchi et al., 2018)。尽管无句法信息的的模型取得了成功，但是仍有许多研究致力于如何将句法优势应用到语义角色标注中。 (Roth and Lapata, 2016)将复杂的句法结构和句法相关现象看作是句法依赖路径的子序列，并将深度Bi-LSTM应用到语义角色标注中，在PropBank数据集上取得较好效果； (Qian et al., 2017)提出了SA-LSTM，以一种结构工程的方式对整个句法依赖树结构进行建模。

目前，大部分性能较好的语义角色标注模型都是基于Bi-LSTM的，但Bi-LSTM无法很好的获取句子的句法信息。近年来，注意力机制在自然语言处理领域广泛应用，受到 (He et al., 2018)和 (Zhang et al., 2019)的启发，本文引入self-attention机制，将其加入到词表示和Bi-LSTM编码器之间，同时使用条件随机场进行标签预测。

3 基于Self-Attention的语义角色标注

本文将语义角色标注任务转换成序列标注问题，给定一个句子序列及其目标词，在目标词所激起框架已知的条件下，识别出句子中与目标词所搭配的语义角色。即：给定句子$X = (w_1, w_2, w_3, ...w_n)$，和其目标词$w_{target}(1 \leq target \leq n)$，输出角色标签序列$Y^* = \arg\max p(Y|X, w_{target}, frame)$。图2为本文的模型结构，它由三个模块组成：（1）self-attention层：对句子中各词的语义重要性进行建模；（2）Bi-LSTM编码层；（3）CRF标签预测层。

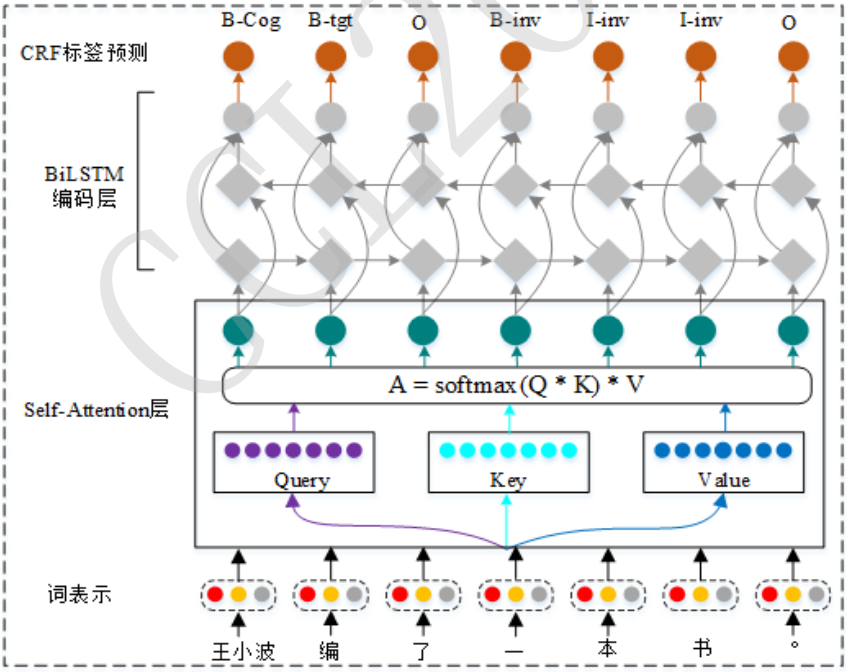


图 2. 语义角色标注模型图

3.1 词特征向量表示

本文为句子中的每个词w_i生成一个词表示x_i, 其中，i表示词在句子中的位置。每个x_i是由几个特征拼接起来的：预训练的词向量x_i^{pe}、随机初始化的词性表示x_i^{pos}、随机初始化的框架表示x_i^{frame}、随机初始化的目标词表示x_i^{target}、位置表示x_i^{loc}、以及谓词指示器$x_i^{indicator}$。所以，最终的词表示为$x_i = [x_i^{pe}, x_i^{pos}, x_i^{frame}, x_i^{target}, x_i^{loc}, x_i^{indicator}]$。

3.2 Self-Attention层

在语义角色标注任务中，通常对句法的依赖性较强，但是Bi-LSTM模型擅长获取句子中的长距离依赖信息，却不能很好地提取句法信息。因此本文在词表示和Bi-LSTM编码层之间引入self-attention机制捕获句子中每个词的句法信息，同时还可以进一步增强Bi-LSTM获取长距离依赖信息的能力。

将词表示矩阵X映射为不同的表示K，Q，V，首先对Q和K执行点积操作，并对其进行缩放操作，如公式（1）所示。再对其执行softmax操作进行归一化，得到句子中每个词之间的attention权重a_i。最后，将attention权重a_i点乘V并求和，得到每个词的表示矩阵A。

$$f(Q,K_i)=\frac{QK_i^T}{\sqrt{d_k}} \tag{1}$$

$$a_i=soft\max(f(Q,K_i))=\frac{exp(f(Q,K_i))}{\sum_j exp(f(Q,K_j))} \tag{2}$$

$$A=\sum_i a_iV_i \tag{3}$$

3.3 Bi-LSTM编码层

在汉语框架语义角色标注任务中，目标词与语义角色块并不会总是相邻的。因此，为了更加有效地获取长距离信息，在self-attention编码层之后，我们加入一个多层Bi-LSTM编码层，以取得更加丰富的表示。

在LSTM网络中，每个细胞单元有$\widetilde{C}$，g_i，g_f，g_o，C_t，h_t，其中$\widetilde{C}$是是当前细胞状态的候选值；g是控制信息流动的门；C_t是当前细胞状态；h_t是细胞隐状态，具体如下：

$$\widetilde{C}=\tanh(W_cz_t+U_ch_{t-1}+b_c) \tag{4}$$

$$g_j=sigmoid(W_jz_t+U_jh_{t-1}+b_j),j\in\{i,f,o\} \tag{5}$$

$$C_t=g_j\odot\widetilde{C}+g_f\odot C_{t-1} \tag{6}$$

$$h_t=g_o\odot f(C_{t-1}) \tag{7}$$

其中，$\odot$代表按元素乘。最后将前向序列$\overrightarrow{h_t}$和反向序列$\overleftarrow{h_t}$拼接得到Bi-LSTM编码层的输出$h_t=[\overrightarrow{h_t};\overleftarrow{h_t}]$。

3.4 标签预测

我们在Bi-LSTM编码层之后，加上一层CRF进行标签预测。因为，在序列标注问题中，相邻词的标签间存在很强的依赖关系，单独考虑每个词的标签得分是不合适的。所以使用CRF对整个序列进行全局归一化，得到概率最大的最优序列。

假设输入序列为$X=(x_1,x_2,\cdots,x_n)$，输出序列为$y=(y_1,y_2,\cdots,y_n)$，约定H是Bi-LSTM层的输出矩阵，而$H_{i,j}$对应于句子中第i个单词的第j个标签的得分。那么，它的得分定义如式（8）所示，其中，S是标签的转移得分矩阵，$S_{i,j}$表示从标签i到标签j的转移得分。

$$s(X,y)=\sum_{i=0}^{n}S_{y_i,y_{i+1}}+\sum_{i=1}^{n}H_{i,y_i} \tag{8}$$

对于输出序列y所有可能的标签序列的概率的softmax形式如式（9），其中，Y_X表示输入序列X的所有可能的标签序列。在解码时，我们通过式（10）预测输出序列。

$$P(y|X)=e^{s(X,y)}/\sum_{\widetilde{C}\in Y_X}e^{s(X,\widetilde{y})} \tag{9}$$

$$y^*=\arg\max_{\widetilde{y}\in Y_X}s(X,\widetilde{y})) \tag{10}$$

4 实验设计及分析

4.1 实验数据

本文实验数据来自山西大学的汉语框架网的例句库，该例句库的标注数据由专业人士人工标注。本文选取其中的25个框架658个词元共19355条标注数据。本实验将上述数据按框架分割成训练集、验证集、测试集，分割比例为6:2:2。

4.2 评价指标

本文使用准确率（Precision）、召回率（Recall）、$F1$值作为评价指标。具体如下：

$$P = CC/PC \tag{11}$$

$$R = CC/DC \tag{12}$$

$$F1 = 2PR/(P+R) \tag{13}$$

其中，CC为模型识别出的正确的标签数；PC为模型识别出的标签数；DC为数据集中的标签数。

4.3 参数设置

使用Glove在CFN例句库上预训练的词向量，词向量维度为100。词性、句法路径特征等特征的维度为10，其中，使用LTP（语言技术平台）进行句法分析。此外，其他超参数设置为学习率$learning_rate = 0.015$，丢弃率$dropout = 0.5$，隐藏层维度$hidden_dim = 200$，优化函数为SGD，正则化系数$L2 = 1e-8$。

4.4 实验结果与实验分析

表1为本文模型在CFN数据集上实验结果和已有模型的对比。表中第一栏为已有模型中基于神经网络模型的汉语框架语义角色标注模型，其中最好结果为72.89%。本文在基于Bi-LSTM的语义角色标注模型中融入self-attention机制，在CFN数据集上实现了83.77%的F1值，比已有模型中最好结果提高10.88个百分点，比未引入self-attention机制之前的实验结果提升了5.94个百分点。

模型	F1/%
(Lv, 2014)	60.51
(Yang, 2015)	69.91
(Wang et al., 2017)	70.54
(Dang, 2015)	72.89
ABLC	83.77

表 1. 语义角色标注在CFN数据集上的结果对比

为了验证self-attention机制的效果，设置了两个对比实验：BLC和BLAC，前者未引入self-attention机制，后者将self-attention机制加入到Bi-LSTM编码器之后，实验结果如表2所示。首先，无句法路径特征时，本文模型ABLC和BLAC较BLC分别提高了7.39%、4.49%，而加入路径特征时，ABLC、BLAC比BLC分别提高了5.94%、3.98%；其次，本文的ABLC模型比BLAC的性能分别高了2.9%和1.96%。因此，验证了本文模型引入self-attention机制在汉语框架语义角色标注任务中的有效性。此外，引入self-attention机制后，即BLAC和ABLC，加入句法路径特征对结果提升不是很明显，分别提高了0.92%和-0.02%，表明本文模型可以获取一定句法信息。

为了分析不同路径特征对实验结果的影响，本文分别使用了三种不同的路径特征进行实验，结果如表3所示。总体而言，使用一级路径特征的实验结果相对较好。另外，从表中可以看出，不同路径特征对BLC模型的影响较大，而对引入self-attention机制的BLAC和ABLC的影响则相对较小。进一步证明了本文模型可以提取一定的句法信息。

模型	无句法特征						有句法特征					
	dev			test			dev			test		
	P	*R*	*F*1	*P*	*R*	*F*1	*P*	*R*	*F*1	*P*	*R*	*F*1
BLC	81.67	68.60	74.12	83.56	71.70	76.40	81.20	70.03	74.46	83.83	73.30	77.83
BLAC	83.67	72.49	77.55	89.24	73.72	80.89	83.63	74.02	78.58	88.84	76.17	81.81
ABLC	86.16	77.88	81.53	89.21	79.35	83.79	83.03	74.53	78.15	89.93	79.00	83.77

表 2. 对比实验结果

模型	F1		
	全路径	二级路径	一级路径
BLC	77.83	76.91	81.41
BLAC	81.81	81.75	82.25
ABLC	83.77	84.32	83.96

表 3. 不同路径特征的实验结果对比

由于数据稀疏对实验的影响，本文对利用框架关系对实验数据的扩充进行初步探索。本文使用三种方式进行数据扩充。将目标框架的数据与关系数据的train、dev分别合并后作为扩充数据的train、dev，扩充后的数据称为Data1；为了验证扩充数据在无例句的框架的性能，将关系数据的train、dev作为扩充数据的train、dev，称为Data2；因为非核心框架元素具有稀疏性，会影响标注模型的性能，因此，将关系数据的train、dev去掉核心框架框架元素作为扩充数据的train、dev，称为Data3。

表4为本文模型在扩充数据上实验结果。从表2和表4可以得出，在Data1和Data3上的实验结果都比在目标框架的原始数据上的实验结果高；同时，在Data2上的实验结果可以达到与目标框架原始数据相当水平。结果表明，利用框架关系扩充数据可以提高汉语框架语义角色标注的性能。

模型	句法特征			有句法特征		
	*Data*1	*Data*2	*Data*3	*Data*1	*Data*2	*Data*3
BLC	81.79	77.36	84.79	82.43	76.58	85.83
BLAC	79.02	75.52	83.66	79.08	76.10	85.58
ABLC	82.48	75.05	83.11	87.71	76.40	84.38

表 4. 扩充数据上的实验结果

为了分析本文模型对句子中长距离依赖的影响，记录了不同的角色与目标词的距离集合的F1值，如图3所示。从图中可以看出，所有距离的F1值都得到了改进。这表明了我们的模型在引入self-attention机制后可以增强模型捕获长距离信息的能力，进而提升汉语框架语义角色标注性能。

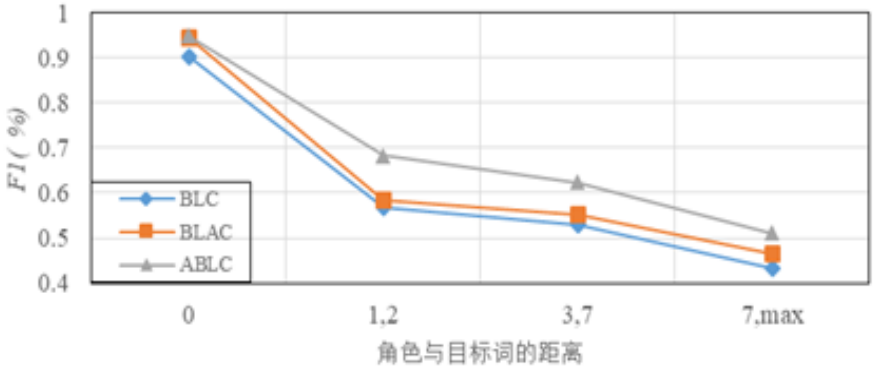


图 3. 角色与目标词距离和F1值

5 总结

本文提出了一种基于self-attention机制的句法感知汉语框架语义角色标注模型，同时，本文利用框架关系对实验数据进行扩充，以减少数据稀疏对语义角色标注的影响。实验结果表明，本文模型的实验结果比已有模型中的最好提高了超过10个百分点，并且验证了本文模型可以获取一定句法信息；同时，利用框架关系对数据进行扩充，有助于缓解SRL中的数据稀疏问题，尤其对无例句框架的SRL的帮助更大。此外，在未来工作中，在更好的融入句法信息以及数据稀疏方面还有很多工作需要完成。

参考文献

Surdeanu Sanda, Williams John and Aarseth Paul. 2003. *Using Predicate-Argument Structures for Information Extraction, Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*,pages:8-15. Sapporo, Japan.

Yih Wen-tau, Richardson Matthew, Meek Chris, Chang Ming-Wei and Suh Jina. 2016. *The Value of Semantic Parse Labeling for Knowledge Base Question Answering, Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*,pages:201-206. Berlin, Germany.

张苗苗, 张玉洁, 刘明童, 徐金安, 陈钰枫. 2018. 基于*Gate*机制与*Bi-LSTM-CRF*的汉语语义角色标注, 计算机与现代化,(4):1-6+31.

Zhou Jie and Xu Wei. 2015. *End-to-end learning of semantic role labeling using recurrent neural networks, Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*,pages:1127-1137. Beijing, China.

He Luheng, Lee Kenton, Lewis Mike, Zettlemoyer Luke. 2017. *Deep Semantic Role Labeling: What Works and What's Next, Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*,pages:473-483. Vancouver, Canada.

Marcheggiani Diego, Frolov Anton, Titov Ivan. 2017. *A Simple and Accurate Syntax-Agnostic Neural Model for Dependency-based Semantic Role, Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*,pages:411-420. Vancouver, Canada.

Punyakanok Vasin, Roth Dan, Yih Wen-tau. 2008. *The Importance of Syntactic Parsing and Inference in Semantic Role Labeling, Computational Linguistics*,34(2):257-287.

Marcheggiani Diego and Titov Ivanu. 2017. *Encoding Sentences with Graph Convolutional Networks for Semantic Role Labeling, Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*,pages:1506-1515. Copenhagen, Denmark.

Gildea Daniel and Jurafsky Daniel. 2002. *Automatic Labeling of Semantic Roles, Computational Linguistics*,28(3):245-288.

Sun Honglin and Jurafsky Daniel. 2004. *Shallow Semantic Parsing of Chinese, Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*,pages:249-256. Boston, Massachusetts, USA.

Xue Nianwen. 2008. *Labeling Chinese Predicates with Semantic Roles, Computational Linguistics*,34(2):225-255.

李济洪, 王瑞波, 王蔚林, 李国臣. 2010. 汉语框架语义角色的自动标注, 软件学报,21(4):597-611.

王蔚林. 基于最大熵模型的汉语框架语义角色自动标注, 山西大学,2010.

屠寒非, 李茹, 王智强, 周铁峰. 2016. 一种基于主动学习的框架元素标注, 中文信息学报,30(4):44–55.

Ronan Collobert and Jason Weston. 2008. *A unified architecture for natural language processing: Deep neural networks with multitask learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008)*,pages:249-256. Helsinki, Finland.

王臻, 常宝宝, 穗志方. 2014. 基于分层输出神经网络的汉语语义角色标注, 中文信息学报,28(6):51-61.

Wang Zhen, Jiang Tingsong, Chang Baobao, Sui Zhifang. 2015. *Chinese Semantic Role Labeling with Bidirectional Recurrent Neural Networks=, Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*,pages:1626-1631. Lisbon, Portugal.

党帅兵. 基于词分布表征的汉语框架语义角色识别研究, 山西大学,2015.

王瑞波, 李济洪, 李国臣, 杨耀文. 2017. 基于Dropout正则化的汉语框架语义角色识别, 中文信息学报,31(1):147-154.

Tan Zhixing , Wang Mingxuan, Xie Jun, Chen Yidong, Shi Xiaodong . 2018. *Deep Semantic Role Labeling With Self-Attention, Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18)*,pages:4929–4936. New Orleans, Louisiana, USA, February.

Ouchi Hiroki, Shindo Hiroyuki, Matsumoto Yuji. 2018. *A Span Selection Model for Semantic Role Labeling, Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*,pages:1630-1642. Brussels, Belgium.

Qian Feng, Sha Lei, Chang Baobao, Liu Luchen, Zhang Ming. 2017. *Syntax Aware LSTM model for Semantic Role Labeling, Proceedings of the 2nd Workshop on Structured Prediction for Natural Language Processing.*

Ouchi Hiroki, Shindo Hiroyuki, Matsumoto Yuji. 2016. *Neural Semantic Role Labeling with Dependency Path Embeddings, Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*,pages:1192-1202. Berlin, Germany.

He Luheng, Lee Kenton, Levy Omer, Zettlemoyer Luke. 2018. *Jointly Predicting Predicates and Arguments in Neural Semantic Role Labeling, Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*,pages:364-369. Melbourne, Australia.

He Luheng, Lee Kenton, Levy Omer, Zettlemoyer Luke. 2019. *Syntax-Enhanced Self-Attention-Based Semantic Role Labeling, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*,pages:616-626. Hong Kong, China.

吕雷. 汉语框架语义角色自动标注研究, 山西大学,2014.

杨耀文. 基于神经网络模型的汉语框架语义角色识别, 山西大学,2016.

基于词语聚类的汉语口语教材自动推送素材研究[1]

杨冰冰
北京语言大学汉语学院
942249583@qq.com

赵慧周
信息科学学院
zhaohuizhou@blcu.edu.cn

王治敏*
汉语国际教育研究院
wangzm000@qq.com

摘要

新冠肺炎的蔓延使得线上移动教学成为教育发展的必然趋势，本文以适合汉语教材自动推送的口语素材为研究对象，基于10341条生活类口语语料，对词汇的整体特点进行计量分析，在此基础上根据腾讯AL LAB 公开的中文词向量数据，使用Kmeans算法对口语词汇进行词语聚类，参考词语聚类结果及对口语语料话题和场景的考察，构建了一个包含15个一级话题、102个二级话题及81个交际场景的汉语口语话题-场景素材库。同时对各级话题常用词进行了总结。本文可为教材自动定制的素材库提供资源支持。

关键词： 汉语资源建设；词语聚类；教材自动推送；汉语口语

Study on Automatic Push Material of Oral Chinese Textbook Based on Word Clustering

Yang Bingbing
Beijing Language and Culture University Chinese Language Institute
942249583@qq.com

Zhao Huizhou
Beijing Language and Culture University School of Information Science
zhaohuizhou@blcu.edu.cn

Wang Zhimin
Beijing Language and Culture University Chinese International Education Research Institute
wangzm000@qq.com

Abstract

The spread of Coronavirus has made online teaching an inevitable trend. This paper takes oral materials suitable for automatic push of Chinese textbooks as the study object. Based on 10341 daily oral sentence, the overall characteristics of the vocabulary are quantitatively analyzed. On this basis, according to the Chinese word vector data published by Tencent AL LAB, the Kmeans algorithm is used to analyze Spoken language vocabulary is clustered, referring to the results of word clustering and the investigation of the topics and scenes of the spoken language corpus, a Chinese spoken topic-scene material library containing 15 first-level topics, 102 second-level topics and 81 communication scenarios is constructed. At the same time, it summarizes the commonly used words on topics at all levels. This paper can provide resource support for the material library automatically customized for teaching materials.

Keywords: Chinese resource construction , word clustering , textbook automatic push , spoken Chinese

[1]本研究得到国家社科基金重大项目“基于‘互联网+’的国际汉语教学资源与智慧教育平台研究”（18ZDA295），中央高校基本科研业务费（18YBT03；19PT03）的资助。

1 引言

新冠肺炎在全世界的蔓延使得在线学习成为一种趋势。在线学习资源如何满足学习者的学习需求，如何为学习者提供丰富的、可供选择的个性化学习素材，如何联通汉语学习者与教师这两部分群体，是目前特殊大环境下亟待解决的问题，也将是语言教育创新发展，适应互联网时代发展面临的问题。

本文面向在线学习中的汉语口语教材自动定制，对其所需要的学习素材进行研究。教材自动定制需要联通词汇、场景、话题等模块，构建相应的推理模式。关于词汇资源的研究，目前很多词汇知识库的研究主要是基于规则和基于统计的，也有学者利用文本分类中特征提取的方法在大规模分类语料中自动获取领域词语（刘华，2007）。基于词语聚类，一些学者对不同话题领域的词表进行了研究（吕荣兰2011；喻雪玲2013；陈钰铭2015）。关于对外汉语中的话题分类，《国际汉语教学通用教程大纲》（2008）列出了包括22个话题以及其下的小话题与常用表达，一些学者构建了汉语话题库（吕荣兰2011；苏新春2011；方沁2014）。目前对汉语交际场景的研究有杨寄洲（2000）在《对外汉语教学初级阶段教学大纲》中列举的36个交际场景，其后刘华（2014）根据影视片段，总结了63个交际场景。现有研究对我们的研究有一定的参考作用，但是目前对生活类话题下的二级话题的研究还不够深入，且缺少对交际场景和话题关联的相关研究，对汉语口语的特点也缺少分析。

因此本文以适用于汉语教材自动推送的教材素材为研究对象，通过对口语词汇特点的分析以及词语聚类，总结口语话题及场景，构建话题-场景对应的口语素材库，为在线口语教材自动推送提供资源支持。

2 汉语口语词汇特点分析

本文以汉语生活类口语为切入点，收集了10341句生活类语料。语料来源于《英语会话8000句》中生活类话题下的句子以及真实场景的对话录音。《英语会话8000句》作为英语口语的代表教材，经过长期的使用可以证明其话题覆盖面较宽，句子实用性较强，且与汉语口语教材有一定的区分，经过筛选与改写后可以作为汉语口语教材的原始语料。关于口语录音，本文对生活中常见话题、场景进行了真实场景的录音，通过机器转写与人工校对形成本文的另一部分基础语料。笔者所处语言环境为北京的高校，录音场所为北京内的交际场所。通过对全部语料进行机器分词和人工校对[2]，去除数字、英文字母后共提取出词汇6803个。

2.1 词类分布特点分析

在对全部词汇进行分词与词性标注的基础上，我们发现在口语会话中各类词汇分布如图1：

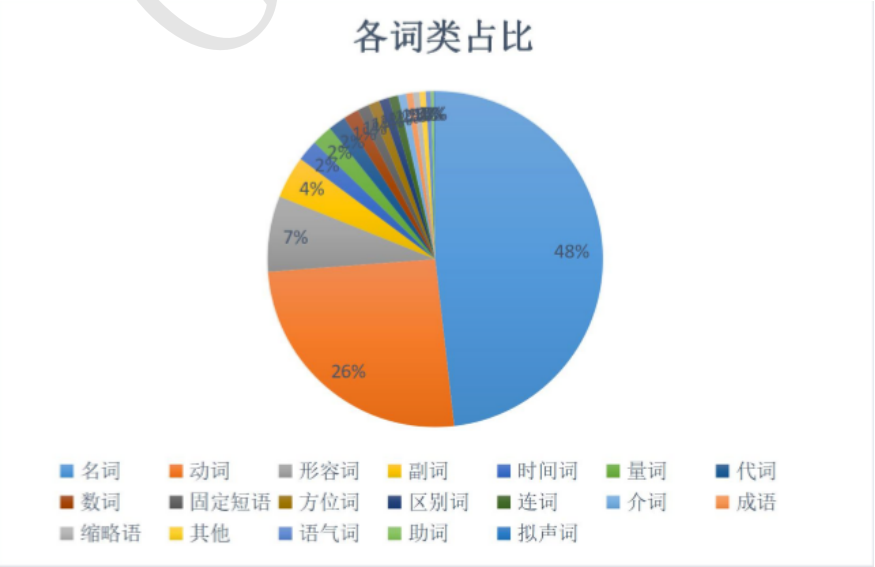


图 1 各词类分布图

[2]分词标准参考《现代汉语语法信息词典》

图1表示在所有口语语料中各词类丰富度情况，总体来看，在口语会话中名词种类最多，数量为3279个，占全部词汇的48%。其次为动词，数量为1741个，占比26%，再者为形容词，数量为496个，占比7%，副词数量为278个，占比4%。这四种词类包含的不同词语种类最多，共占全部词汇的85%，与汉语词类构成相符。时间词、量词、代词数量丰富，数量分别为139个、135个、115个，体现出口语对话中用词的特点。在口语对话中存在一部分固定短语，如“不好意思、没关系、可不”等，数量为78个。另外，在口语语料中也有一部分成语，但是数量不多，为44个。另外有一些无法判定词类的词语我们将其分为“其他”类。从各词类丰富度来看，名词、动词、形容词依旧是口语学习的重点，语气词、助词等属于封闭类词类因此数量不多，但是重点词语较多，如语气词中的“吧、呢”；助词中的“的、了”等。

词频反映一定范围内词语的常用度，因此下文将对口语对话中各词类的词频进行统计分析。

图 2　各词类词频图

由图2可知，各词类词频与各词类词语数量不一致，在口语语料中，实词的出现频率高于虚词。在所有词类中动词的词频最高，总词频为19718次，由此可知在口语对话中动词的常用度在所有词类中占据优势。其次为代词，代词虽数量不多，但是出现的频次很高，口语对话大多是两人之间的对话，因此进行自我阐述与询问对方使用的人称代词出现的频率很高。词频第三为名词，总频次为10909次，名词数量为3191个，这反映出大部分名词出现频次较低，常用度较高的名词不多。词频第四为语气词，频次为6132次，语气词是虚词中频次最高的词类。语气词虽数量不多，但是出现频率很高，语气词的高频使用反映了汉语口语的特点。其次为助词，频次为4598次，使用频率较高。另外在口语语料中，成语、拟声词等词频较低，常用度较低。

2.2　高频词词汇特点分析

口语中的高频词可以体现口语词汇特点，在口语中词频最高的词为：“我、的、你、了、是、好、吗、不、吧、您”等。在口语中代词“我”的频次最高，这与《中国语言生活研究报告》中“的”词频第一不一致，因为在口语对话中多为表达自己观点的句子，因此代词“我”的词频比书面语等语体中高。另外其他人称代词“你”、“您”、“我们”的词频也很高。

从音节数来看，在词频为前50的词中，大多为单音节词，数量为41个，双音节词数量较少，仅为9个。单音节词的数量远多于双音节词及多音节词，且频率越高，单音节词优势越强。从难易程度上看，高频口语词汇中大部分属于汉语低水平词汇，通俗性较强。

为了对高频词的词类分布进行下一步的研究，我们对词频前50各词类数量进行了统计，如图3所示。

由图3可知在高频词中动词数量最多，共15个，其次为代词，数量为10个，副词和语气词的数量也较多。助词、介词、数词中各有2个高频词，形容词、方位词、名词、量词、时间词中各有一个高频词，高频词数量较少。其他没有列出的词类没有频次在前50的高频词。

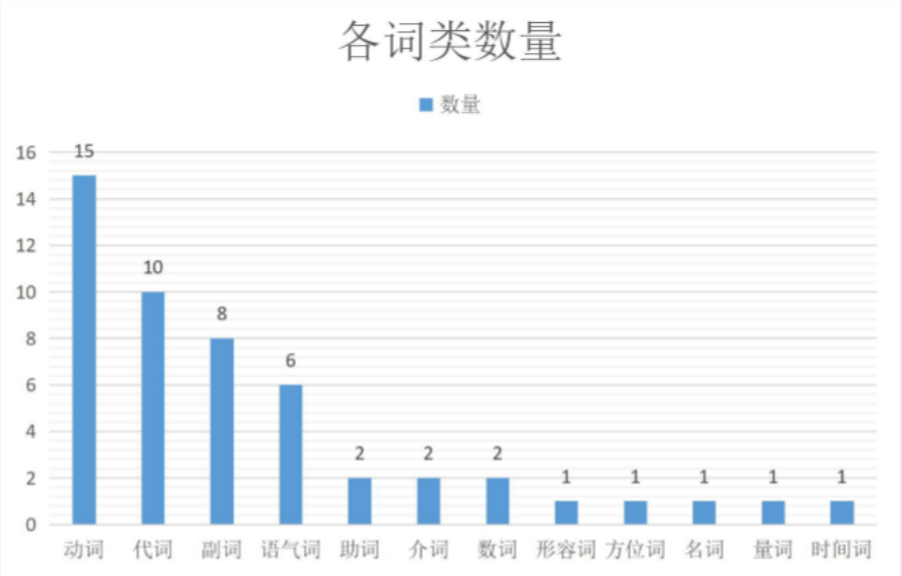


图 3　频率前50词类统计图

3　基于词向量的词语聚类

词语聚类可以把语义特征相似的词语聚在一起，自动定制的汉语教材需要提供各话题、场景的常用词及常用句，词语聚类结果可为话题和场景下常用词的汇总提供参考。本文对全部10341条语料进行话题的场景的标注，在词语聚类和标注的基础上归纳生活类话题和场景。

3.1　词向量模型

文本的向量化即把文本转化为n维的向量。本研究使用腾讯AL LAB 公开的中文词向量数据，包含800万词汇，每个词对应一个200维的词向量，该词向量数据包含很多现有公开的词向量数据所欠缺的短语，所计算的语义相似度较高，且采用了Directional Skip-Gram(DSG)算法作为词向量的训练算法，DSG算法基于广泛采用的词向量训练算法Skip-Gram(SG),在文本窗口中词对共现关系的基础上，额外考虑了词对的相对位置，所生成的词向量能够更好地表达词之间的语义关系。

3.2　Kmeans聚类算法

Kmeans 算法是最经典的基于划分的聚类方法，首先确定想要经过聚类得到若干集合的k值，我们将k值设置为500、800和1000。从全部的词汇向量中随机选择k个数据点作为质心。计算所有向量数据与聚类中心的相似度，距离离质心越近，相似度越高，计算公式如下：

$$|AB| = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

计算完成后将数据分配给与其最相似的聚类中心代表的类，并计算下一轮聚类的中心。如果新计算出来的质心和原来的质心之间的距离小于某一个设置的阈值，我们可以认为聚类已达到期望的效果，算法终止。如果新质心和原质心距离变化很大，需要继续迭代。聚类为500类、800类及1000类的实验结果如下，以生病就医类词语为例。

由表1可知，聚类总数越多，同一话题内词汇的聚类数也越多，分类越细致，每类之间的区分度越弱。在聚类数为500类的词语聚类中，对药物名称、药物效果及大夫等的划分包含在了一个类里，而在聚类数为800类的聚类中，药物名称单独为一类，在聚类数为1000的聚类中，药物名称根据种类又分为了两个类。因此对于口语话题来说，对于大话题的划分参考聚类数为500的词语聚类，对于下级话题和场景的划分参考聚类数为1000和800的词语聚类结果。

聚类数	平均词语数量	词语
500类（10类）	10.80	医院、出院、报销、住院、护士、看病、家属、病房、医保、转院、陪床、动手术、查房、住院处、住院费、主刀、药费
		医生、手术、病、治疗、病人、化疗、疗程、患者、微创、会诊、切除、预后、术后、病情、病理、不治之症、康复、确诊、疗效、治愈率、癌症、就诊、矫正、输血
		感冒、发烧、生病、咳嗽、流感、重感冒、鼻塞、麻疹、传染、退烧、上火、发高烧、感染、着凉
		血、伤口、伤、受伤、绷带、清创、切口、外伤、愈合、流血
		药、大夫、药方、治、中药、药店、处方、安眠药、药物、口服、盘尼西林、见效、药效、药膏、阿司匹林、送服、药品、体温计、方子、药水
		吃药、忌口、牙疼、打针、戒烟、服药
		药房、配药、化验室
		挂号、挂号费、挂号单、专家号、血压、血糖、心率、心电图
		眼科、诊室、内科、中医科、科室、牙科、北医三院
		账单、收据、收付款、票据、预约单、化验单
800类（15类）	7.87	医院、出院、住院、护士、救护车、眼科、会诊、转院、牙科、内科、化验室、科室
		病房、诊室、查房、住院处、中医科
		医生、大夫、病人、家属、患者、肺、病理
		病、感冒、发烧、照顾、生病、吃药、病倒、体质、打针、静养、重感冒、陪床、卧床、养病、发高烧
		嗓子、咳嗽、闷、喘
		流感、传染、感染
		治疗、化疗、治、病情、康复、确诊、退烧、靶向、疗效、治愈率、矫正
		感康、止痛片、布洛芬、感冒药、头孢
		伤、受伤、后遗症、伤筋动骨
		疼、肿、痛、疼痛、痒、剧痛、蛰、血压、血糖、心率、心电图
		手术、微创、切除、开刀、拆线、术后、动手术、输血、主刀
		药、中药、西药、中成药、药物、药方、含片
		服用、疗程、口服、阿司匹林、送服、服药
		看病、挂号、挂号费、预约单、挂号单、化验单、转诊单、就诊、专家号
		报销、医保、住院费、药费、医保卡
1000类（21类）	5.62	医生、大夫、医院、病人、护士、看病、家属、会诊、诊室、动手术、查房、就诊、主刀
		眼科、牙科、病理、内科、中医科、科室
		感冒、发烧、生病、病倒、重感冒、发高烧
		流感、传染、麻疹、感染
		嗓子、咳嗽、鼻涕、闷、咳、喘、憋、呛
		疼、酸痛、肿、痛、疼痛、剧痛
		伤、受伤、事故、崴脚
		伤口、绷带、清创、外伤、愈合、流血

聚类数	平均词语数量	词语
		治疗、护理、化疗、患者、预后、术后、病情、康复、确诊、治愈率、矫正
		吃药、打针、拆线、麻药
		药物、副作用、药品、疗效、激素、麻醉剂
		药、中药、配药、西药、中成药、安眠药、止痛片、药方、含片、退烧、止咳
		布洛芬、感冒药
		感康、头孢
		药房、药店、器械公司
		手术、微创、切除、切口、阑尾、活检
		手术、微创、切除、切口、阑尾、活检
		静养、养病
		出院、住院、病房、转院陪床
		挂号、挂号费、预约单、挂号单、住院处
		报销、医保、住院费、药费

表 1　500类、800类、1000类词语聚类情况

4　话题—场景库的构建

4.1　话题的确定

基于词语聚类结果，在对12本常用口语教材、《国际汉语教学通用教程大纲》（2008）、吕荣兰（2011）、方沁（2014）等话题库以及英语教材的话题进行总结的基础上，我们总结出生活类一级话题15个，分别为："个人信息、日常交际、居家生活、运动健身、生病就医、交通出行、购物、饮食就餐、天气、日期时间、资金管理、生活服务、休闲娱乐、意外与事故和住宿"。每个一级话题下又有若干个二级话题，根据语料的场景标注，每个二级话题都有其对应的场景。以"生病就医"话题为例：

一级话题	二级话题	场景
生病就医（I）	不适	家（JA）、宿舍（SS）、户外（HW）、路上（LS）、办公室（BG）
	预约	电话（TP）、预约平台（YP）、医院（HP）
	挂号	医院（HP）、预约平台（YP）
	就诊	医院（HP）、诊室（ZS）、诊所（ZS）、病房（BF）
	费用	医院（HP）
	买药	药房（YF）
	手术	手术室外（OR）、诊室（ZS）、病房（BF）
	住院	病房（BF）、医院（HP）、住院部（ZY）
	探病	病房（BF）、住院部（ZY）、家（JA）

表 2　"生病就医"二级话题、场景

参考词语聚类结果，我们总结出上表所示在一级话题"生病"中有二级话题9个，二级话题排列顺序基本遵循去医院看病流程。总体来看，关于"生病就医"话题的对话场景较为固定，几乎所有的二级话题中涉及到的场景都有"医院"。

通过对全部一级话题对话及词语聚类的考察，本文总结出15个一级话题下共102个二级话题，每个话题下有其对应的交际场景；根据对话内容，总结出每个话题和场景下的口语常用句。

4.2　各级话题词汇特点及常用词分析

不同话题下词汇具有不同的特点，教材自动推送资源需要提供各个话题、场景下常用度高、代表性强的词汇。为提高话题词汇相关性，突出各话题词汇特点，本文对各级话题词汇特

点及常用词的研究去除总词频中频次最高的前15个词，分别是：我、的、你、了（语气词）、是、好、吗、不、吧、您、这、有、就、我们、在，且去除一些无意义的虚词，分别是助词、介词、连词和语气词。我们对每个一级话题下的词汇特点进行了分析（以“生病就医”为例）：

词类	词频	词语及频次
动词	645	要（69）可以（56）去（54）做（48）吃（42）能（36）看（32）来（32）需要（30）想（29）休息（24）说（24）开（23）拿（22）得（20）手术（18）感冒（16）治疗（13）出院（11）预约（11）挂号（11）发烧（9）服用（8）住院（7）
名词	185	药（42）医生（34）时候（23）时间（23）大夫（18）医院（16）药房（11）体温（10）病人（8）
形容词	63	疼（27）多（16）舒服（12）严重（8）

表 3　“生病就医”话题高频词统计

通过统计发现在“生病就医”话题中高频动词数量很多，其中“休息、开、手术、感冒”等话题相关性较强，名词中“药、医生、大夫”话题相关性较强，形容词中“疼、舒服、严重”具有话题代表性。可作为“生病就医”一级话题下的常用词参考。

二级话题	场景	常用词
不适	家、宿舍、户外、路上、办公室	怎么了、哪里、疼、有点、药、医院、看病、感冒、难受、发烧、头疼、脸色、生病、医生、着凉、肚子、受伤、舒服、咳嗽、胃口
预约	电话、预约平台、医院	预约、时间、有空、合适、上班、满、网上、公众号
挂号	医院、预约平台	挂号、专家号、普通号、预约、网上、科、交费、挂号费、排队、内科、外科、口腔科
就诊	医院、诊室、诊所、病房	怎么了、哪里、问题、疼、药、医生、大夫、治疗、做、检查、严重、开药、休息、服用、情况、打针、输液、症状、伤口、医院、拍片、化验、量、注意、饮食
费用	医院、交费处	报销、医保卡、刷卡、单子、交费、交费处、排队、现金、自助
买药	药房	中药、药方、消炎、退烧药、过敏、配药、次、片、粒、过敏、症状、头孢、感冒、感冒药、止疼片
手术	手术室外、诊室、病房	手术、签、术前、家属、安排、术后、体质、风险、肿块、同意书、营养、恢复、开刀、部位、麻醉、麻药、输血、微创
住院	病房、医院、住院部	住院、出院、注意、休息、办、手续、陪床、检查
探病	病房、住院部、家	休息、怎么样、放心、情况、担心、营养、出院、照顾、保重、早日康复

表 4　“生病就医”二级话题常用词

在对每个一级话题进行词频统计的基础上，我们提取出二级话题中的话题高频词，并参考高频词的词语聚类结果，总结各二级话题下的常用词：

4.3　生活类场景统计分析

通过对语料场景的标注以及对话题的考察，我们统计出81个生活类场景，相比较《对外汉语教学初级阶段情景大纲》（杨寄洲，2000），涵盖的生活范围进一步增加，场景是口语交际发生的场所，培养学习者根据场景进行交际是培养口语交际能力的关键。口语素材库需要尽可能多的交际场景来满足学习者的交际需求。具体生活类场景如表5：

[3]括号内数字表示该场景在多少个一级话题中出现。

场景类型	场景
居住场所	家（12）[3]宿舍（10）小区（3）厨房（1）宾馆（1）
交通工具	出租车（2）地铁（1）地铁站（1）公交车（1）公交站（1）火车（1）火车站（1）飞机（1）机场（1）
就餐场所	饭店（3）咖啡厅（3）食堂（2）酒吧（2）快餐店（1）西餐厅（1）火锅店（1）奶茶店（1）小吃街（1）甜品店（1）
运动场所	操场（3）体育馆（2）篮球场（2）足球场（2）健身房（1）瑜伽馆（1）游泳馆（1）
购物场所	商场（3）市场（2）超市（1）商店（1）服装店（1）家具店（1）书店（1）
就医场所	医院（3）诊所（1）医院诊室（1）病房（1）住院部（1）手术室外（1）药房（1）
休闲娱乐场所	电影院（1）剧院（1）KTV（1）音乐厅（1）游乐场（1）景区（1）赛车场（1）滑雪场（1）美容院（1）美甲店（1）游戏厅（1）售票处（1）
学习办公场所	校园（7）办公室（5）公司（4）教室（3）图书馆（2）
服务场所	维修店（1）洗衣店（1）理发店（1）快递点（1）打印店（1）
公共场所	路上（8）户外（4）公共场所（1）
手机平台	电话（10）微信（2）网购平台（1）外卖平台（1）预约平台（1）
其他	中介公司（1）证券公司（1）银行（1）ATM机（1）旅行社（1）警察局（1）

表 5　生活类场景表

以上是根据话题所总结出的场景，共81个生活类场景。对于一些常见场景我们进行了更深一层的细分，如“医院”是一个大的场景，主要是“生病就医”一级话题的交际场景，“生病就医”一级话题下又有多个二级话题，那么本文在“医院”这一大场景下又细分了“诊室、病房”这些常见的小场景。另外，本文增加了手机平台场景，如“购物”话题中的网店，“外卖”话题中涉及到的外卖平台。目前网络平台在我们日常生活中占据了很重要的位置，这些平台上的对话虽不是常规的口语，但是具备口语色彩。对留学生来说，有必要学习常用的网络平台的对话进行交际。

通过对话题下场景的分析，我们发现一些场景几乎包含了所有话题，如“家、宿舍”这些固定生活场所，而大部分场景下话题受限制较大，在我们收集到的语料中一般只包含一个话题，如“医院诊室、病房、手术室外”场景一般只涉及“生病就医”话题、“音乐厅、游乐场”等场景一般只涉及“休闲娱乐”话题。这些场景内人们的对话通常是针对某一话题展开，话题延展性不强。由此可知大多话题和场景的关联性较强。

5　口语素材库在教材自动定制中的应用分析

基于上文口语话题-场景素材库的研究，本文对素材库的实现进行了前期的测试，可根据场景类型进行场景的选择，如图4:

图 4　场景选择模式测试

在场景选择模式中，用户可根据场景关键词进行学习场景的选择，图4所示场所类别选择为“医”，场景即为“生病就医”相关场景，为“药店、医院”等。当用户选择场景为“医院”时，学

习内容推送如图5：

场景	一级话题	二级话题	三级话题	句子	句子组序号	话轮中的句子序号
医院	日常交际	常用语	安慰	我这病怕是好不了了。	1	1
医院	日常交际	常用语	安慰	别担心，会好的。	2	1
医院	日常交际	常用语	劝阻	老刘的病很严重，咱们应该把病情告诉他。	1	1
医院	日常交际	常用语	劝阻	最好不要吧，他会受不了的。	2	1
医院	生病	费用	报销	你好，请问我的医保卡可以报销吗？	1	1
医院	生病	费用	报销	我看一下你的医保卡。	2	1
医院	生病	费用	报销	哦，你这个是只能住院才可以报销的，平时的买药是不能报销的。	3	1
医院	生病	费用	报销	哦，那好吧。	4	1
医院	生病	费用	报销	你知道咱们学校的医保可以报销多少吗？	5	1
医院	生病	费用	报销	据说是可以报销90%。	6	1
医院	生病	费用	报销	吓，那么多呀？	7	1
医院	生病	费用	报销	对呀，所以说在咱们校医院看病是很划算的。	8	1
医院	生病	费用	报销	我正好感冒了，我要去拿点药。	9	1
医院	生病	费用	报销	你看本来这些药，如果在药店买的话需要100多块钱，但是咱们校医院报销过以后只用花十几块钱。	10	1
医院	生病	费用	报销	咱们学校报销的真的挺多的。	11	1
医院	生病	费用	报销	对。	12	1
医院	生病	费用	缴费	好的，你拿着这个单子交一下费。	1	1
医院	生病	费用	缴费	出了门有一个自助的交费机器，你就在那上面交就可以。	2	1
医院	生病	费用	缴费	好的这是你的病例单，下一次过来的时候拿着这个和预约单。	3	1
医院	生病	费用	缴费	那我还用下面去，下面挂号吗？	4	1
医院	生病	费用	缴费	不用去了，你就直接来上面排队就可以。	5	1
医院	生病	费用	缴费	好的谢谢医生。	6	1
医院	生病	挂号		您得先填写这张挂号表。	1	1
医院	生病	挂号		我应该挂哪科呢？	2	1
医院	生病	挂号		你最好挂内科，坐电梯到3楼左拐沿着走道走，你会看到一块牌子在你右手边儿，在那	3	1

图 5　场景为“医院”的学习内容测试

在场景“医院”下，涉及到一级话题“日常交际”、“生病就医”及其下的二级、三级话题，图5所示句子组序号表示该句为所在话轮的第几句。此测试可以根据学习者的场景特定需求为其推送学习素材。

另外我们对用户的学习界面进行了设计，以学习者的就医需求为例进行口语素材推送分析，学习内容包括话题常用词、对话及常用句：

图 6　学习模块页面设计

在学习内容界面，常用词是口语素材库中归纳出的各低级话题的常用词，对常用词的学习配备相应的拼音、词语发音、词语朗读纠音以及图片展示。口语会话学习内容也保留了口语会

话的特点，在会话范读中保留口语的语音语调，但语速不宜过快。最后常用句的学习中，红色词语是可以被替换的内容，如第二句中，“头疼、嗓子疼”中的“头、嗓子”可以替换为“牙、肚子”等身体部位。常用句是对口语会话的总结，来源于上文口语素材研究中的资源。

6 结语

本文基于10341条口语语料，对适合在线教材自动推送的汉语口语素材进行了分析。本文在分词与词频统计的基础上对汉语口语词汇特征进行了深入描写，总结各词类分布特点，同时根据腾讯AL LAB 公开的中文词向量数据，使用Kmeans算法对口语词汇进行词语聚类，将全部词语分别聚类为500类、800类和1000类，通过对聚类结果的分析，发现聚类总数越多，同一话题内词汇的聚类数也越多，分类越细致，每类之间的区分度越弱，因此本文参考500类的分类结果对大话题进行划分，参考800类和1000类的结果对下级话题和场景进行归纳。其次，基于词语聚类，在对语料场景话题及现有话题库的考察下，本文构建了一个包含15个一级话题、102个二级话题以及81个交际场景的话题-场景框架体系，并对各级话题下的常用词及常用句进行了总结。本文最后对口语素材库在教材自动推送中的实现进行了前期测试，并且对用户的学习界面进行了设计，口语素材库可以实现根据学习者的场景特定需求为其推送学习素材。下一阶段的研究将针对素材库的推送进行更深层次的分析，制作具有实用价值的汉语教材自动推送产品，服务于汉语学习者与汉语教师。

参考文献

陈珏铭. 2015. 基于话题及交际图式的汉语会话常用词和常用句研究. 暨南大学.

崔彩霞. 2008. 停用词的选取对文本分类效果的影响研究. 太原师范学院学报:自然科学版,7(04):91–93.

方沁. 2013. 基于话题分类的汉语教学影视片段资源库构建. 暨南大学

官琴,邓三鸿,王昊. 2017. 中文文本聚类常用停用词表对比研究. 数据分析与知识发现,1(03):72–80.

国家汉语国际推广领导小组办公室. 2008. 国际汉语教学通用课程大纲. 北京:外语教学与研究出版社.

刘华. 2019. 基于文本分类中特征提取的领域词语聚类. 语言文字应用,2007(01):139–144.

吕荣兰. 2011. 基于语料库的对外汉语口语话题及话题词表构建. 暨南大学.

苏新春,唐诗瑶,周娟. 2011. 话题分析模块及七套海外汉语教材的话题分析. 江西科技师范学院学报,2011(06):58–65.

杨继洲. 2000. 对外汉语初级阶段功能大纲. 北京:北京语言大学出版社.

俞士汶等. 2003. 现代汉语语法信息词典详解（第二版）. 清华大学出版社.

喻雪玲. 2013. 基于语料库的商务汉语话题库及话题词表构建. 暨南大学.

郑艳群. 2012 . 多属性标注的汉语口语教学多媒体素材库建设及应用. 语言教学与研究,2012(05):34–39.

周小兵. 2010. 建设数字化国际汉语教学资源库. 华文教学与研究,2010(01):1.

Song Yan,Shi Shuming,Li Jing,Zhang Haisong. 2018. *Directional Skip-Gram: Explicitly Distinguishing Left and Right Context for Word Embeddings.* Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers),175–180..

基于半监督学习的中文社交文本事件聚类方法*

郭恒睿[1]，王中卿[1]，李培峰[12*]，朱巧明[12]
[1]苏州大学计算机科学与技术学院，苏州，中国
[2]苏州大学人工智能研究院，苏州，中国
hengruig@outlook.com, {wangzq, pfli, qmzhu}@suda.edu.cn

摘要

面向社交媒体的事件聚类旨在根据事件特征对短文本聚类。目前，事件聚类模型主要分为无监督模型和有监督模型。无监督模型聚类效果较差，有监督模型依赖大量标注数据。基于此，本文提出了一种半监督事件聚类模型(SemiEC)，该模型在小规模标注数据的基础上，利用LSTM表征事件，利用线性模型计算文本相似度，进行增量聚类，利用增量聚类产生的标注数据对模型再训练，结束后对不确定样本再聚类。实验表明，SemiEC的性能相比其他模型均有所提高。

关键词： 社交媒体事件聚类；增量聚类；文本相似度

Semi-supervised Method to Cluster Chinese Events on Social Streams

Hengrui Guo[1], Zhongqing Wang[1], Peifeng Li[12*], Qiaoming Zhu[12]
[1]School of Computer Science and Technology, Soochow University, Suzhou, China
[2]AI Research Institute, Soochow University, Suzhou, China
hengruig@outlook.com, {wangzq, pfli, qmzhu}@suda.edu.cn

Abstract

Event clustering on social streams aims to cluster short texts according to event contents. Event clustering models can be divided into unsupervised learning or supervised learning at present. The unsupervised models suffer from poor performance, while the supervised models require lots of labeling data. To address the above issues, this paper proposes a semi-supervised incremental event clustering model SemiEC based on a small-scale annotated dataset. This model encodes the events by LSTM and calculates text similarity by a linear model, and then clusters short texts on social streams. In particular, it uses the samples generated by incremental clustering to retrain the model and redistribute the uncertain samples. Experimental results show that this model SemiEC outperforms the traditional clustering algorithms.

Keywords: event clustering on social media , incremental clustering , text similarity

基金项目：国家自然科学基金(61772354，61836007)，国家自然科学基金青年基金项目(61806137)，江苏高校优势学科建设工程资助项目。

1 引言

在如今的网络时代，随着移动互联网的发展，信息交互变得前所未有的简便快捷。QQ、微信、微博、抖音、快手等社交媒体深入走进了人们的生活，改变了人们的生活习惯。研究表明，社交媒体对于新事件的反应要比传统媒体更加敏锐(Petrović et al., 2010)。因此，对社交媒体中的文本进行数据分析有着非常重要的意义。其中，事件聚类是社交媒体中事件检测的重要步骤(Aggarwal and Subbian, 2012)。

事件聚类旨在根据文本事件特征的不同对社交媒体中的文本进行聚类。社交媒体中多为短文本，且文本内容具有多样性、随意性，包含较多的干扰词。传统的无监督聚类模型难以准确提取社交文本的事件特征，因而得到的事件聚类结果一般准确度较低。Wang and Zhang (2017)采用了有监督的深度神经网络模型对社交文本进行聚类，增强了聚类效果，但面对海量的社交文本，该方法需要大量的文本标注工作。

基于此，本文提出了一种半监督的中文社交文本增量事件聚类模型SemiEC(Semi-supervised Chinese incremental Event Clustering model)。SemiEC模型利用LSTM(Hochreiter and Schmidhuber, 1997)提取文本特征，利用线性模型计算两个文本属于同一事件的概率，在此基础上进行增量聚类。该模型利用增量聚类过程产生的标注样本对模型进行再训练。在无需额外数据标注工作的同时，帮助模型学习更多的事件信息，提高聚类效果。同时，对于聚类过程中的不确定样本，暂时不进行聚类，在结束后用再训练后的模型进行重新聚类，可以防止较差样本影响簇心表征，影响模型再训练，提高对不确定样本的聚类准确度。实验表明，SemiEC模型与基准模型相比在各项聚类指标上均得到了提高。

2 相关工作

目前，大部分事件聚类研究主要基于词的特征。与长文本不同，短文本聚类存在高维稀疏的问题(Aggarwal and Subbian, 2012)，因此一些学者考虑引入外部特征。其中Mathioudakis and Koudas (2010)以及Saeed et al. (2019)利用突发性关键词来预测短文本的重要性，并通过这些重要的短文本来检测事件进行聚类。Nguyen and Jung (2015)通过考虑事件的发布时间、扩散程度和扩散敏感性，采用时间特征来检测事件，并进行聚类。除此之外，Li et al. (2012)探索了用户在社交媒体数据中的影响，利用文本内容特征、用户特征和使用特征来检测事件并进行聚类。Mcminn and Jose (2015)借助了文本的命名实体特征来加强事件检测的效果。

为了解决传统方法高维稀疏的问题，Cai et al. (2005)通过局部保存索引(LPI)将高维文本投影到低维语意空间，同时使语意相关的文本在低维空间中也彼此接近。Qimin et al. (2015)使用Kmeans聚类算法对特征词集进行聚类得到特征簇，用特征簇表示句向量，从而解决了向量空间模型维度爆炸的问题，同时提高了聚类效果。Zhou et al. (2018)以word2vec词向量为基础，结合时序关系，提出了JS-ID'F顺序来进行文本嵌入。Arora et al. (2019)提出了对文本的SIF Embedding，通过对词向量进行加权平均，再用PCA和SVD对其进行一些修改，得到文本的低维向量表示。Xu et al. (2015)(Xu et al., 2017)采用了基于DCNN的深度神经网络学习文本的深度特征表示。该模型首先通过现有无监督降维方法得到文本的二进制编码，然后将文本通过Word Embedding输入卷积神经网络，将文本的二进制编码作为模型的训练目标。将卷积层与输出层之间的中间特征向量作为文本的深度特征表示，是一种基于自训练的无监督模型。

对于社交媒体中的流式数据，常用的聚类算法有singlepass增量算法和局部敏感哈希(LSH)算法。对于到来的新样本，singlepass聚类算法首先需要计算新来文本与已有事件的相似度，若相似度超过阈值，则将其加入相似度最大的已有事件，否则将其设为新事件(Allan et al., 1998)。该算法的关键步骤在于计算文本相似度，目前最为常用的是余弦相似度cosine。局部敏感哈希(LSH)聚类算法主要基于新事件检测模型(FSD)，其思路是通过LSH算法找到新来文本在已聚类文本中的近邻文本集合，从该集合中找到新来文本的最近邻文本，若两者最大相似度大于设定阈值，则为已有事件，否则为新事件(Petrović et al., 2010)。在局部敏感哈希(LSH)聚类算法中，其关键步骤在于通过LSH算法尽快找到新来文本的近邻文本。Wurzer et al. (2015)对寻找最近邻文本的哈希算法做了改进，提高了效率，但准确率与Petrović et al. (2010)相当。Xie et al. (2016)提出了一种基于自训练的深度嵌入式聚类模型。该模型使用深度神经网络同时学习特征表示和聚类分配，是一种基于划分的聚类模型，不适合处理流式数

据。Hadifar et al. (2019)使用SIF Embedding进行句子表征，采用自编码器提取文本的低维特征表示，采用类似Xie et al. (2016)的聚类算法通过自训练的神经网络模型同时学习文本特征表示和聚类分配，得到聚类结果。Finley and Joachims (2005)利用有监督的SVM模型判断两个文本是否相关，通过Bansal et al. (2004)的方法进行文本聚类，该聚类方法将样本分布视为一个图模型，通过最大化簇内的样本对相似性实现聚类。Haponchyk et al. (2018)将Finley and Joachims (2005)的方法进行改进，用于对话系统中用户问题的聚类，从而分析用户意图。Wang and Zhang (2017)采用了有监督的LSTM模型提取文本特征，计算文本相似度，采用增量聚类算法进行社交文本聚类，相比此前的聚类算法有所提高，但需要大量的数据对模型进行训练。目前，在面向社交媒体的事件聚类方法中，还没有采用半监督的方法。

3 半监督的社交媒体事件增量事件聚类模型（SemiEC）

有监督聚类算法虽然聚类效果较好，但需要大量的数据标注工作来训练模型提高效果。无监督方法的性能又往往不能满足实用的需求。基于此，本文提出了一种半监督的中文社交文本增量事件聚类模型（SemiEC），相比Wang and Zhang (2017)采用的模型在相同训练的情况下进一步提高聚类效果。

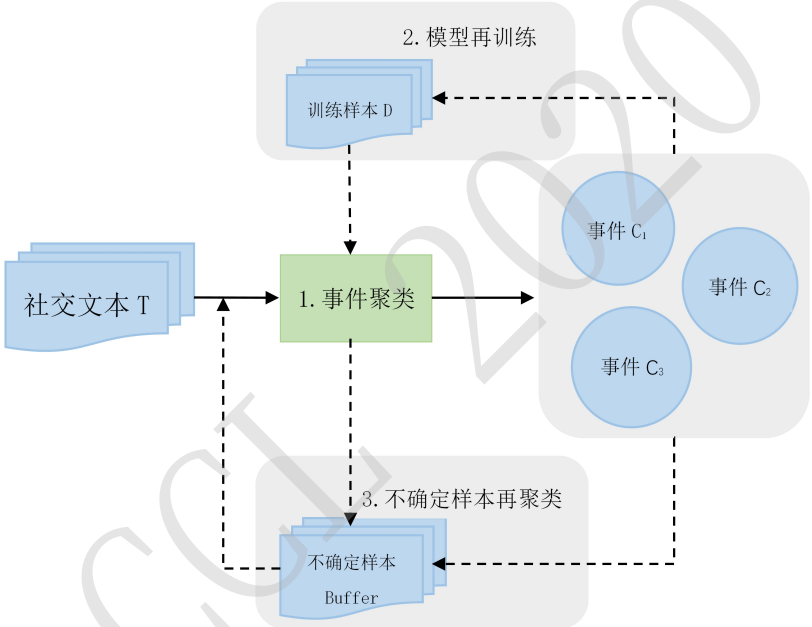


图 1: 基于数据增强的增量聚类过程

聚类过程如图1，对于输入的社交媒体文本t_i，首先对其进行事件聚类，判断t_i是属于已有事件还是新生事件，或是无法确定。若t_i属于已有事件，则将其加入该簇；若为新事件，则基于t_i建立新的簇，否则设t_i为不确定样本，加入Buffer。该过程中使用LSTM提取文本特征，利用线性模型计算文本相似度。由于增量聚类算法可以得到实时的聚类结果，因此每次对部分数据进行聚类后，从聚类结果中抽取样本组成训练集对模型进行再训练，使模型进一步学习新的事件特征，增强聚类效果。在结束后，将聚类结果中所含元素较少的簇中的样本全部设为不确定样本。然后用经过多次再训练后的模型对不确定样本进行聚类，得到最终的聚类结果。下面对聚类过程进行详细介绍。

3.1 文本表示

类似Wang and Zhang (2017)，SemiEC采用LSTM模型提取文本特征，记为$M_{encoder}$。在文本输入前，首先进行分词和去停用词。将分词后的文本用$X = \{w_1, w_2, \ldots, w_n\}$表示。其中$w_i$表示句子中第$i$个词在词表中的编号，$n$表示句子中的词数。使用由百度百科预训练的中文词向量对句子进行词嵌入，用n维词向量x_i表示词w_i，$x_i \in \mathbb{R}^{n*d}$。将文本向量$X = (x_1, x_2, \ldots, x_n)$通过LSTM模型得到一个隐藏序列$\{h_1, h_2, \ldots, h_n\}$，其中$h_t$由当前输入向量$x_t$和前一时刻的输

出h_{t-1}计算得到，$t \in (1,n)$，即$h_t = LSTM\left(x_t, h_{t-1}\right)$。训练过程中初始状态的参数均为随机生成。我们采用低维向量$H = h_n$来表示文本X。

3.2 文本相似度计算

对于两个文本X_i和X_j，首先通过$M_{encoder}$得到两个文本的特征向量H_i和H_j。将向量H_i和H_j拼接后，通过一个线性层得到向量H_c，最后通过一维线性层，由sigmoid函数得到X_i和X_j的相似度P_c，$P_c \in [0,1]$，将相似度计算模型记为M_{sim}，计算过程如下式。

$$H_c = \sigma\left(W_c^h\left(H_i \oplus H_j\right) + b_c^h\right) \tag{1}$$

$$P_c = sigmoid\left(W_c H_c + b_c\right) \tag{2}$$

其中，$\oplus$代表两个向量的拼接，W_c^h，b_c^h，W_c，b_c为模型参数。

3.3 聚类算法

SemiEC模型的聚类过程主要分为三个步骤：1）：对社交媒体文本的事件聚类；2）：对模型的再训练；3）：对不确定样本的重新聚类。

- 对社交文本的事件聚类(算法1-12行)

对于一个新到来的社交文本t_i，要判断其与已有事件的最大相似度。SemiEC模型用事件簇中的前N个文本来表示这个簇，作为簇心，若样本数不足N，则将簇中全部样本作为簇心。将t_i与代表簇心的各个文本的相似度平均值作为t_i与该簇的相似度。设定两个阈值L和H，其中$0 < L < 0.5 < H < 1$。若t_i与目前已有簇的最大相似度大于H，则将t_i加入相似度最大的簇；若t_i与目前已有簇的最大相似度小于L，则以t_i建立一个新的簇；若t_i与目前已有簇的最大相似度在L和H之间，则认为t_i分配不确定，将t_i加入缓冲区$Buffer$，暂时不进行聚类。

传统singlepass算法采用一个阈值判断文本t_i属于已有事件还是属于新生事件，相比较而言SemiEC模型加入了不确定样本这一分类，可以防止较差样本影响簇心表征和再训练的质量。

- 对模型的再训练(算法13-17行)

将特征提取模型$M_{encoder}$和相似度模型M_{sim}拼接到一起同时训练。设置一个更新阈值U，每当有U个样本完成聚类，就从U个样本中抽取D组训练数据组成训练集，对模型进行再训练。训练集中正例与负例的比为1:1。正例抽取方法为：从U个样本中随机选取一个样本p，再从U个样本中与p同簇的样本集中随机选取一个样本q，样本p与样本q组成一组正例，标签置为1。负例的抽取方法为：从U个样本中随机选取一个样本p，再从U个样本中与p不同簇的样本集中随机选取一个样本q，若U中没有与p不同簇的，则从目前所有已聚类且与p不同簇的样本中随机选取一个样本q，样本p与样本q组成一组负例，标签置为0。若目前所有已聚类的样本中都没有与p不同簇的，则本次不进行训练。

相比Wang and Zhang (2017)的方法，SemiEC增加了模型再训练的步骤，利用增量聚类过程中产生的标注数据对模型进行再训练，可以使模型学习新事件的特征，进一步提高模型的泛化能力。

- 不确定样本进行重新聚类(算法18-32行)

在增量聚类结束后，经常会出现个别样本数特别少的簇，这是增量聚类模型经常容易出现的问题。若聚类结束后，出现包含样本数少于N的事件簇，则删除这些簇，并将这些簇中的样本加入缓冲区$Buffer$，$Buffer$中包含聚类过程中的不确定样本。增量聚类结束后，再对$Buffer$中的样本进行重新聚类。计算不确定样本与已有事件的最大相似度，若样本与已有事件簇的最大相似度大于0.5，则将其加入该簇，否则以该样本建立新的簇。

相比Wang and Zhang (2017)的方法，SemiEC增加了不确定样本重新聚类的步骤，一方面，可以防止不确定样本加入簇心影响事件簇的表征以及进入训练集对模型进行错误的训练；另一方面，在聚类结束后，模型经过多次训练后效果有所增强，可以对这些不确定的样本进行更准确的聚类。

Algorithm 1 半监督社交媒体事件增量聚类算法

算法开始

Input:

社交文本$T = \{t_1, t_2, \ldots, t_n\}$
阈值L和H，表征簇的文本个数N
特征提取模型$M_{encoder}$，相似度模型M_{sim}
更新阈值U，每次训练集样本数D，缓冲区$Buffer$

Output:

事件聚类结果$C = \{C_1, C_2, \ldots, C_k\}$

Initialize: 将t_1初始化为第一个簇$C = \{C_1\}$；$Buffer = \varnothing$
for each $t_i \in \{t_2, t_3, \ldots, t_n\}$ **do**
  对于文本t_i，利用$M_{encoder}$得到其特征向量X_i
  利用M_{sim}计算X_i和C中的每个簇C_m的相似度Sim_m
  得到与t_i相似度最大的簇C_r，$Sim_r = Max\,(Sim_m)$
  if $Sim_r > H$ **then**
    将t_i加入C_r
  else if $Sim_r \geq L$ **then**
    将t_i加入$Buffer$
  else
    将t_i设为新的簇，加入C
  end if
  if 有U个样本加入C **then**
    抽取D组数据组成训练集，对模型M_{sim}和$M_{encoder}$进行训练
    通过$M_{encoder}$更新C中簇的表示
  end if
end for
for each $C_i \in C$ **do**
  if $|C_i| < N$ **then**
    将C_i中样本加入$Buffer$，删除C_i
  end if
end for
for each $t_j \in Buffer$ **do**
  对于文本t_j，利用$M_{encoder}$得到其特征向量X_j
  利用M_{sim}计算X_j和C中的每个簇C_m的相似度Sim_m
  得到与t_j相似度最大的簇C_r，$Sim_r = Max\,(Sim_m)$
  if $Sim_r > 0.5$ **then**
    将t_j加入C_r
  else
    将t_j设为新的簇，加入C
  end if
end for
输出$C = \{C_1, C_2, \ldots, C_k\}$

4 实验部分

4.1 实验数据

本次实验的数据来自微博。采用与Wang and Zhang (2017)相同的规则搜集了关于40次不同地震事件的微博，共计10828个。采用30次地震事件作为训练数据，从中随机选取样本组成训练集。其中同一地震事件中的两个文本为正例，标签为1；不同地震事件中的两个文本为负例，标签为0。训练集中正例和负例比为1:1，总共抽取了400000组样本训练模型。剩余的10次地震事件数据作为测试集，共包含2518个文本，用于事件聚类实验。

4.2 实验参数

聚类算法中N的取值为25，用事件簇中的前N个文本来表示该簇。N越大，对事件簇的代表性越强，但计算量也会增大。聚类过程中的阈值L和H的的取值范围为$0 < L < 0.5 < H < 1$，L的取值越接近0，H的取值越接近1，对不确定样本的筛选越严格，但与此同时也会增大计算量。本次实验中L的取值为0.3，H的取值为0.7。更新阈值U的取值不能太小，否则训练过于频繁，使模型容易受一些极端值的影响，产生偏离；U值过大则会使大量样本仅能使用原始模型聚类。本次实验中U值为200。每次训练集样本数D的取值越大，模型对事件特征的学习效果越明显，但同时也会增大运算量。同时D也受U的影响，实验中训练集样本数D取值为800，为U的4倍。训练轮数为5轮。

4.3 模型参数

模型基于Keras框架，后端为Tensorflow。特征提取模型$M_{encoder}$训练过程中使用了由百度百科预训练的词向量，向量维度为300，嵌入层设置为不可训练。LSTM层输出维度为128，dropout=0.1，recurrent_dropout=0.1，不返回序列，其余为默认参数。相似度模型M_{sim}中的全连接层输出维度为128，激活函数为relu，其中还包含两个Dropout层，Dropout=0.1。

神经网络N的输出为y_i，真实标签为$\bar{y_i}$，为0或1。优化器为adam，采用交叉熵损失函数binary_crossentropy，计算步骤如下：

$$loss = -\frac{1}{N}\sum_{i=1}^{N} \bar{y_i}\log(y_i) + (1-\bar{y_i})\log(1-y_i) \tag{3}$$

4.4 模型训练

将$M_{encoder}$和M_{sim}拼接到一起，两个文本通过$M_{encoder}$得到特征向量H_i和H_j，将其作为M_{sim}的输入，组成一个计算文本相似度的孪生神经网络N，最终得到两个文本的相似度。

将神经网络N在训练集上训练5轮，将训练得到的参数赋予$M_{encoder}$和M_{sim}，从而实现模型的预训练。模型再训练同样是对网络N进行再训练，训练轮数为5轮，将更新后的参数赋予$M_{encoder}$和M_{sim}，从而实现模型的再训练。

4.5 评价指标

我们采用纯度Purity，归一化互信息NMI和调整兰德系数ARI作为聚类效果的评价指标。Purity是正确计算的文本数与文本总数的比值。其定义如下：

$$Purity\left(\Omega, C\right) = \frac{1}{N}\sum_{k} \underset{j}{max}\left|w_k \cap c_j\right| \tag{4}$$

其中N表示总的样本个数，$\Omega = \{w_1, w_2, \ldots, w_K\}$表示聚类模型得到的聚类簇划分，$C = \{c_1, c_2, \ldots, c_J\}$ 表示真实类别划分。Purity取值范围为$[0, 1]$，越接近1，聚类效果越好。

NMI是一个基于熵的评价指标，其定义如下：

$$NMI\left(\Omega, C\right) = \frac{I\left(\Omega; C\right)}{\left[H\left(\Omega\right) + H\left(C\right)\right]/2} \tag{5}$$

其中I表示互信息:

$$I(\Omega;C)=\sum_k\sum_j P(w_k\cap c_j)\log\frac{P(w_k\cap c_j)}{P(w_k)P(c_j)}=\sum_k\sum_j\frac{|w_k\cap c_j|}{N}\log\frac{N|w_k\cap c_j|}{|w_k||c_j|} \tag{6}$$

H表示熵:

$$H(\Omega)=-\sum_i\frac{w_i}{N}\log\frac{w_i}{N} \tag{7}$$

NMI的取值范围为$[0,1]$，越接近1，聚类效果越好。
调整兰德系数ARI弥补了兰德系数RI惩罚力度不够的问题，其定义如下:

$$RI=\frac{a+b}{C_N^2} \tag{8}$$

$$ARI=\frac{RI-E[RI]}{max[RI]-E[RI]} \tag{9}$$

其中，a表示实际类簇与聚类预测类簇中都是同类别的元素对数，b表示实际类簇不同类别，在聚类预测类簇中也是不同类别的元素对数。$E[RI]$为RI的平均值。ARI的取值范围为$[-1,1]$，越接近1，聚类效果越好。

4.6 聚类结果

我们将提出的事件聚类模型得到的聚类结果与以下聚类方法进行对比。

1) *Singlepass*：该聚类方法采用向量空间模型表示文本，用cosine计算文本相似度，采用singlepass算法进行聚类，属于无监督聚类算法。模型由自己实现。

2) *Kmeans*：该聚类方法采用向量空间模型表示文本，用cosine计算文本相似度，采用Kmeans算法进行聚类，属于无监督聚类算法。实验中指定正确的事件个数。模型由自己实现。

3) *LSH*(Petrović et al., 2010)：这是一种基于局部敏感哈希的聚类算法，采用向量空间模型表示文本，用cosine计算文本相似度，属于无监督聚类算法。模型由自己实现。

4) *Hadifar*(Hadifar et al., 2019)：该聚类模型采用SIF Embedding表示文本，通过自编码器学习文本的低维特征表示，采用类似Xie et al. (2016)的深度聚类算法进行短文本聚类，属于无监督聚类算法。该方法在实验中采用不同领域的短文本作为测试集，本次实验中的测试集为地震领域的不同事件。实验中指定正确的事件个数。模型采用了改论文中提供的代码。

5) *Wang*(Wang and Zhang, 2017)：该聚类模型利用LSTM提取文本特征，利用线性神经网络模型计算文本相似度，通过增量聚类算法进行聚类，属于有监督聚类算法。模型由自己实现。

6) *BERT*：该方法将Wang and Zhang (2017)的模型换成了BERT词向量，属于有监督聚类算法。模型由自己实现。

由于聚类结果会受数据输入顺序的影响，因此我们将测试数据随机打乱顺序进行10次聚类，记录了各项聚类指标的平均值，聚类结果如表1。其中Wang and Zhang (2017)采用的有监督聚类模型与其他聚类模型相比聚类效果较好，因此以该模型为baseline。在经过相同训练的情况下，SemiEC模型相比baseline模型，在各项聚类指标上均有所提升，在Purity上提升3%，在NMI上提升4%，在ARI上提升10%。

社交媒体中的文本较短，表达具有随意性，即使对于同一事件的评论，其表述方式也各有不同。采用向量空间模型和词向量加权提取文本特征，都容易受干扰词的影响而导致关键特征信息无法突出，从而导致聚类效果较差。

通过对模型进行有监督的训练，相比无监督聚类模型，可以更准确的识别文本的事件特征，增强聚类效果。但基于BERT词向量模型的聚类方法相比基于word2vec词向量的方法聚类

聚类模型	Purity	NMI	ARI
Single-pass	0.54	0.56	0.39
Kmeans	0.76	0.75	0.65
LSH	0.48	0.37	0.20
Hadifar	0.50	0.45	0.34
Wang	0.78	0.79	0.60
BERT	0.63	0.60	0.47
SemiEC	**0.81**	**0.83**	**0.70**

表 1: 聚类结果对比

结果反而有所下降，原因在于BERT词向量模型考虑了更多的语义信息，而社交媒体中即使属于同一事件的文本语义描述也各有不同，因此反而导致结果较差，且BERT词向量模型相比word2vec词向量模型用时较多，对于海量的社交媒体文本而言效率偏低。

5 分析

5.1 训练数据大小对SemiEC模型的影响

有监督聚类算法对训练集的依赖较大，要使模型可以更准确地区分各种事件，其关键在于训练集中是否有足够充分的事件类型。因此，分别选取10次，15次，20次，25次，30次地震事件作为训练数据，从中抽取400000组文本对组成训练集对模型进行训练，以4.1节中的10次地震事件作为测试集，以Wang and Zhang (2017)的模型为baseline，以NMI为参考指标，将测试数据随机打乱进行10次聚类，对NMI取平均值，得到结果如图2。

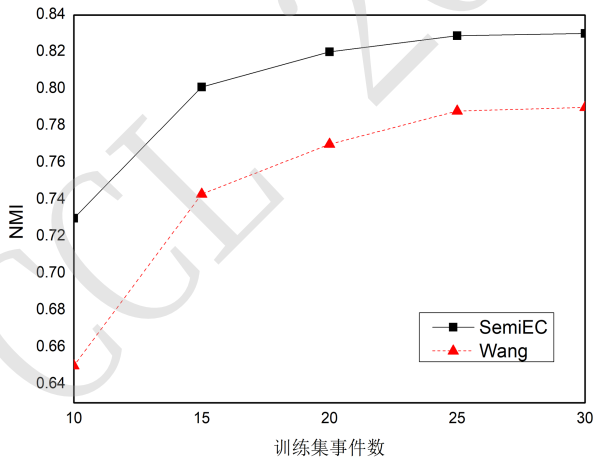


图 2: 不同训练集事件数对聚类结果的影响

由图2可以看出在不同训练集事件数的情况下，SemiEC模型相比baseline模型聚类效果有所提高，但不同训练数据的聚类效果有所差别。当训练事件数为15时，SemiEC的性能已经超过了使用更多训练数据的baseline模型。这充分说明了本文半监督方法的有效性，可以在少量标注数据的基础上，通过利用聚类过程中产生的标注数据学习新的事件特征，获得更好的性能。

5.2 参数设置对SemiEC模型的影响

模型再训练和不确定样本重聚类步骤都需要设置一些额外的参数。其中对SemiEC模型聚类效果影响最大的是更新阈值U，主要用于控制模型再训练的频率。本次实验将U分别设置为10，50，200，500，1000，每次训练集样本数D设置为U的4倍，分别为40，200，800，2000，4000。以4.1节中的10次地震事件作为测试集，以Wang and Zhang (2017)的模型为baseline，以NMI为参考指标，将测试数据随机打乱进行10次聚类，对NMI取平均值，得到结果如图3。

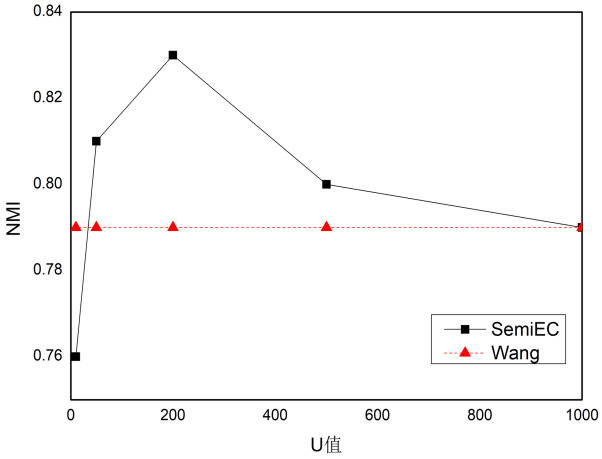


图 3: 不同U值对聚类结果的影响

由图3中结果可以看出，当U值较小，为10时，聚类效果较差，主要原因在于训练集较小，所属类别分布不均匀的概率较大，训练反而导致模型偏向于个别事件，使得聚类所得事件数比实际事件数少，聚类效果较差；当U值逐渐增大到200，样本分布趋于均匀的时候，聚类效果相对baseline会有明显提高；但不断增大U值会减少模型再训练的次数，导致大量数据仅能使用原始模型进行聚类，从而使聚类结果不断趋近但不低于baseline。因此，U的取值要在保证数据分布尽量均匀的情况下，取较小的值，此时可以使SemiEC模型达到最好的聚类效果。

5.3 模型再训练和不确定样本再聚类的有效性

为了证明模型再训练步骤和不确定样本再聚类步骤的有效性，分别对这两个步骤进行了测试。Retrain表示仅加入模型再训练步骤，Recluster表示仅加入不确定样本再聚类步骤。以4.1节中的10次地震事件作为测试集，以Wang and Zhang (2017)的模型为baseline，将测试数据随机打乱进行10次聚类，对各项聚类指标取平均值，得到结果如表2。

聚类模型	Purity	NMI	ARI
Wang	0.78	0.79	0.60
Retrain	0.79	0.82	0.69
Recluster	0.80	0.80	0.65
SemiEC	0.81	0.83	0.70

表 2: 模型再训练和不确定样本再聚类有效性对比

由表2数据可以看出，Retrain和Recluster，相比baseline在各项聚类指标上均有所提高，这充分说明了模型再训练和不确定样本再聚类步骤的有效性。其中，模型再训练步骤可以帮助模型学习新的事件特征，增强对后续样本的聚类效果。不确定样本再聚类步骤可以防止不确定样本加入簇心，减少错误样本对簇心表征的影响，从而增强聚类效果。将两者结合后的SemiEC模型，通过不确定样本再聚类减少错误样本进入训练集，增强模型再训练的效果，同时对不确定样本用再训练后的模型重新聚类，进一步增强不确定样本的聚类效果，两者相互提高，得到最好的聚类效果。

6 总结

本文提出了一种半监督增量型中文社交文本事件聚类模型SemiEC，采用LSTM提取文本特征，采用线性模型计算文本相似度，进行增量聚类。利用增量聚类过程产生的标注样本对模型进行再训练。对聚类过程中分配不确定的样本在结束后重新聚类。再训练过程可以让模型学习新的事件信息，使模型准确度随着聚类过程不断提高。对不确定样本的重新聚类可以防止不确定样本影响簇心表征，减少错误样本对模型进行再训练的概率，同时提高不确定样本的聚类准确度。SemiEC模型与经过同样预训练的有监督聚类模型相比，在各项聚类指标上均有所提高。

参考文献

Charu C Aggarwal and Karthik Subbian. 2012. Event detection in social streams. In *proceedings of the SIAM International Conference on Data Mining*, pages 624–635. SIAM.

James Allan, Jaime G Carbonell, George Doddington, Jonathan Yamron, and Yiming Yang. 1998. Topic detection and tracking pilot study final report.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2019. A simple but tough-to-beat baseline for sentence embeddings. In *proceedings of 5th International Conference on Learning Representations, ICLR 2017*.

Nikhil Bansal, Avrim Blum, and Shuchi Chawla. 2004. Correlation clustering. *Machine learning*, 56(1-3):89–113.

Deng Cai, Xiaofei He, and Jiawei Han. 2005. Document clustering using locality preserving indexing. *IEEE Transactions on Knowledge and Data Engineering*, 17(12):1624–1637.

Thomas Finley and Thorsten Joachims. 2005. Supervised clustering with support vector machines. In *proceedings of the 22nd International Conference on Machine Learning*, pages 217–224.

Amir Hadifar, Lucas Sterckx, Thomas Demeester, and Chris Develder. 2019. A self-training approach for short text clustering. In *proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 194–199.

Iryna Haponchyk, Antonio Uva, Seunghak Yu, Olga Uryupina, and Alessandro Moschitti. 2018. Supervised clustering of questions into intents for dialog system applications. In *proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2310–2321.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Rui Li, Kin Hou Lei, Ravi Khadiwala, and Kevin Chen-Chuan Chang. 2012. Tedas: A Twitter-based event detection and analysis system. In *proceedings of 2012 IEEE 28th International Conference on Data Engineering*, pages 1273–1276. IEEE.

Michael Mathioudakis and Nick Koudas. 2010. Twittermonitor: trend detection over the Twitter stream. In *proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 1155–1158.

Andrew J. Mcminn and Joemon M. Jose. 2015. Real-time entity-based event detection for Twitter. In *proceedings of International Conference of the Cross-language Evaluation Forum for European Languages*.

Duc T Nguyen and Jason J Jung. 2015. Real-time event detection on social data stream. *Mobile Networks and Applications*, 20(4):475–486.

Saša Petrović, Miles Osborne, and Victor Lavrenko. 2010. Streaming first story detection with application to Twitter. In *proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 181–189. Association for Computational Linguistics.

Cao Qimin, Guo Qiao, Wang Yongliang, and Wu Xianghua. 2015. Text clustering using VSM with feature clusters. *Neural Computing and Applications*, 26(4):995–1003.

Zafar Saeed, Rabeeh Ayaz Abbasi, Muhammad Imran Razzak, and Guandong Xu. 2019. Event detection in Twitter stream using weighted dynamic heartbeat graph approach [application notes]. *IEEE Computational Intelligence Magazine*, 14(3):29–38.

Zhongqing Wang and Yue Zhang. 2017. A neural model for joint event detection and summarization. In *proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4158–4164.

Dominik Wurzer, Victor Lavrenko, and Miles Osborne. 2015. Twitter-scale new event detection via k-term hashing. In *proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2584–2589.

Junyuan Xie, Ross Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *proceedings of International Conference on Machine Learning*, pages 478–487.

Jiaming Xu, Peng Wang, Guanhua Tian, Bo Xu, Jun Zhao, Fangyuan Wang, and Hongwei Hao. 2015. Short text clustering via convolutional neural networks. In *proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 62–69.

Jiaming Xu, Bo Xu, Peng Wang, Suncong Zheng, Guanhua Tian, and Jun Zhao. 2017. Self-taught convolutional neural networks for short text clustering. *Neural Networks*, 88:22–31.

Pengpeng Zhou, Zhen Cao, Bin Wu, Chunzi Wu, and Shuqi Yu. 2018. EDM-JBW: A novel event detection model based on JS-ID'F order and Bikmeans with word embedding for news streams. *Journal of computational science*, 28:336–342.

基于多粒度语义交互理解网络的幽默等级识别

张瑾晖[1], 张绍武[1], 樊小超[1,2], 杨亮[1], 林鸿飞[1]†
1.大连理工大学/辽宁省大连市
2.新疆师范大学/新疆自治区乌鲁木齐市
wszjh@mail.dlut.edu.cn, zhangsw@dlut.edu.cn
fxc1982@mail.dlut.edu.cn, liang@dlut.edu.cn
hflin@dlut.edu.cn

摘要

幽默在人们日常交流中发挥着重要作用。随着人工智能的快速发展，幽默等级识别成为自然语言处理领域的热点研究问题之一。已有的幽默等级识别研究往往将幽默文本看作一个整体，忽视了幽默文本内部的语义关系。本文将幽默等级识别视为自然语言推理任务，将幽默文本划分为"铺垫"和"笑点"两个部分，分别对其语义和语义关系进行建模，提出了一种多粒度语义交互理解网络，从单词和子句两个粒度捕获幽默文本中语义的关联和交互。本文在Reddit公开幽默数据集上进行了实验，相比之前最优结果，模型在语料上的准确率提升了1.3%。实验表明，引入幽默内部的语义关系信息可以提高模型幽默识别的性能，而本文提出的模型也可以很好地建模这种语义关系。

关键词： 幽默等级识别 ； 自然语言推理 ； 多粒度 ； 语义交互理解

A Multi-Granularity Semantic Interaction Understanding Network for Humor Level Recognition

Jinhui Zhang[1], Shaowu Zhang[1], Xiaochao Fan[1,2], Liang Yang[1], Hongfei Lin[1]†
1.Dalian University of Technology/Dalian, Liaoning
2.Xinjiang Normal University/Urumqi, Xinjiang Autonomous Region
wszjh@mail.dlut.edu.cn, zhangsw@dlut.edu.cn
fxc1982@mail.dlut.edu.cn, liang@dlut.edu.cn
hflin@dlut.edu.cn

Abstract

Humor plays an important role in daily communication, which makes it important problem for natural language processing. Existing works of humor level recognition tend to treat humor text as a whole, and ignore the study of the inner semantic relations of it. This paper regards humor level recognition as a kind of natural language inference task, divides humor text into two parts: "setup" and "punchline", and models the two and their relations respectively. This paper proposes a multi-granularity semantic interaction understanding network to capture semantic association and interaction in humor text from two granularity of word and clause. We conduct experiments on public humor data set Reddit, and the accuracy of the model on this corpus is improved by 1.3% compared with the previous optimal results. Our experiments show that the semantic relationship information inside humor can improve the performance of model on humor level recognition, and the model proposed by us can also represent the semantic relationship well.

Keywords: Humor Level Recognition , Natural Language Inference , Multi-Granularity , Semantic Interaction Understanding

† 通讯作者

1 引言

幽默普遍存在于人们的日常交流中，是化解尴尬、活跃气氛、促进交流的重要手段，可以对人类身心健康产生积极的影响(Morse, 2007)。随着人工智能的快速发展，如何让计算机识别幽默，并进一步识别幽默的等级成为了目前自然语言处理领域的研究热点之一。幽默识别涉及认知语言学、人工智能、心理学等多个学科，其研究能够更好地促进计算机对人类语言的理解。同时，幽默识别能够赋予计算机从更深层次理解人类情感的能力，在机器翻译和人机交互等领域有着广泛的应用。因此，幽默识别及幽默等级识别具有重要的理论研究价值和广泛的应用价值。

传统的幽默识别通常是识别一个句子或段落是否具有幽默的含义(Mihalcea and Strapparava, 2005; Zhang and Liu, 2014; Blinov et al., 2019)。许多研究表明，幽默具有连续性(Blinov et al., 2019; Weller and Seppi, 2019; Hossain et al., 2019)。幽默等级识别，作为幽默识别任务的延伸，旨在根据幽默程度的不同将幽默文本划分为不同的等级。Paulos等(1980)的研究表明，幽默文本通常能够被划分为“铺垫”和“笑点”两个部分，其中“铺垫”一般先于“笑点”表述，是对背景和前提的交代，而“笑点”则是“铺垫”的延续和反转。Weller等(2019)指出，对“铺垫”和“笑点”两部分语义及其关系的深入理解有助于幽默等级识别。表1展示了一个幽默文本及其铺垫和笑点两部分：

幽默文本	As a typical example of failure, you are so successful.
子句1：铺垫部分	As a typical example of failure
子句2：笑点部分	you are so successful

Table 1: 幽默中的铺垫和笑点

在表1中，幽默文本被划分为两个子句，子句1为“铺垫”，子句2为“笑点”。“笑点”既对铺垫中的“failure”做了补充说明，是“铺垫”的延续，又使用“successful”与“铺垫”中的“failure”形成反转。“铺垫”和“笑点”之间对立统一的关系使句子包含了一定程度的幽默。

现有的幽默识别与幽默等级识别研究通常分两步进行：首先基于幽默理论，设计并实现一系列的幽默特征；然后采用传统的机器学习方法或结合神经网络方法对幽默文本或幽默等级进行识别。Weller等(2019)采用基于Transformer的预训练模型对幽默等级进行识别并取得了较好的性能。人工构造特征耗时耗力且难以对多样性的幽默表达进行全面表征，模型的泛化能力较弱。现有的神经网络模型和预训练模型将铺垫和笑点作为整体进行建模，忽略了其独立的语义信息和交互的关联信息。此外，由于语言的细微差别可能造成幽默的程度不同，仅从单一的粒度提取幽默特征，模型的性能可能受到限制。

综上所述，为了缓解幽默等级识别中的问题，本文提出了一种基于多粒度语义交互理解网络的幽默等级识别方法。针对幽默语言多样性的问题，采用了多种词嵌入表示融合的方法对幽默文本进行表征；针对幽默语义复杂性的问题，采用了局部语义交互理解模块和全局语义交互理解模块，分别从单词粒度和子句粒度提取幽默文本的高维潜在语义特征；针对幽默中“铺垫”和“笑点”的语义关联特点，采用“交互型”的神经网络模型对二者的关联信息进行建模；最后对多粒度的语义特征和交互关联特征进行融合并对幽默等级进行识别。本文的贡献如下：

1. 本文基于多种嵌入表示融合的幽默文本表示，提出了一种基于局部和全局语义理解的神经网络模型，分别从单词级别和子句级别提取幽默文本特征。

2. 本文提出了一种基于交互语义关联特征的神经网络模型，对幽默文本中“铺垫”和“笑点”的关联信息进行建模以抽取幽默语义关联特征。

3. 本文使用基于多粒度语义交互理解网络的幽默等级识别方法，在Reddit公开幽默数据集上进行对比实验，结果表明，本文提出的方法能够有效地提升幽默等级识别的性能。

2 相关工作

作为日常生活中常见的语言现象，幽默理论研究历史久远，基于幽默理论，幽默识别也有很多的研究成果，而幽默等级识别研究则刚刚起步。本节将从幽默理论，幽默识别和幽默等级识别三个方面总结前人的工作。

2.1 幽默理论

幽默理论对幽默等级识别研究具有重要的指导意义。在众多幽默理论中，乖讹论被广泛接受且具有深远的影响。乖讹论认为幽默是人类对不协调事物的感知，当事物的发展违背人们的常识和期望时，幽默就产生了(SULS, 1972)。基于乖讹论，Raskin等(1979)提出了第一个语言学意义上的幽默理论——语义脚本理论（Script Semantic Theory of Humor, SSTH），该理论认为语义对立是幽默产生的重要原因。基于以上幽默理论，Paulos等(1980)将幽默分为“铺垫”和“笑点”，认为两部分之间存在对立统一的关系。

2.2 幽默识别

传统的机器学习方法被广泛应用于幽默识别领域。Yang等(2015)从不一致性、歧义性、语音特性和人机交互特性四个方面提取幽默的语义特征，并采用了随机森林方法识别幽默。Barbieri等(2014)根据幽默问题的语音和歧义性特点，构造了多种幽默特征。Zhang等(2014)基于幽默的语言学理论，构建50多种幽默特征并将它们划分为五个类别。Liu等(2018b)提取了对话中的情感特征及情感关联特征识别对话中的幽默。此外，他们对幽默文本中句法结构特征进行了深入的分析并指出句法结构和幽默文本具有高度的相关性(Liu et al., 2018a)。

近年来，越来越多的深度学习方法被用于识别幽默。杨勇等(2020)从音形义三个维度对幽默特征进行建模，采用层次注意力机制对幽默进行识别。Bertero等(2016a; 2016b)由《生活大爆炸》中的文本和语音内容构建幽默数据集，采用长短期记忆网络和卷积神经网络自动抽取文本语义特征，从而预测对话中的幽默。Baziotis等(2017)利用注意力机制，更好的关注到句子中的特定单词，从而提高了幽默识别的性能。Zhao等(2019)提出了一种采用张量分解的方法提取幽默的语义特征。除了英文，研究者采用深度学习方法对西班牙文(Bahdanau et al., 2014)和俄文(Blinov et al., 2019)语料进行了幽默识别。

2.3 幽默等级识别

幽默等级识别使计算机能够理解哪些语义和语义关系使句子更加有趣。Chris等(2018)对单词的幽默程度建模并对4997个单词的幽默程度进行了评分。Hossain等(2019)通过重新编辑新闻标题使其变得更加幽默，并对编辑前后文本语义的幽默程度进行了分析。Cattle等(2016)将文本划分为“铺垫”和“笑点”两个部分，并指出二者的语义相关性对文本的幽默等级具有显著影响。此外，一些国际著名评测也将幽默等级识别任务作为评测主题(Potash et al., 2017)。

综上所述，幽默理论为幽默等级识别的研究提供了理论研究基础。此外，从不同粒度提取幽默文本中“铺垫”和“笑点”的语义特征和语义关系特征有助于幽默等级识别性能的提升。

3 幽默等级识别方法

基于多粒度语义交互理解网络的幽默等级识别方法主要包括两个层次，语义的嵌入式表示层和交互特征提取层。交互语义特征提取层包括两个部分，局部语义交互理解模块和全局语义交互理解模块。

基于多粒度语义交互理解的神经网络模型如图1所示。语义的嵌入式表示层能够获取幽默文本中“铺垫”和“笑点”的高维潜在语义表示。首先为了更好地获取不同词嵌入表示的语义信息，融合多种词嵌入表示对“铺垫”和“笑点”中的单词进行表征；其次，为了获取高维潜在语义表示，采用双向长短期记忆网络（Bi-directional LSTM, Bi-LSTM）分别提取“铺垫”和“笑点”的语义信息并得到上下文表示。交互语义特征提取层将上下文表示作为输入，从局部和全局两个维度交互地提取“铺垫”和“笑点”中语义特征及两者之间的语义关联性特征。局部语义交互理解模块计算得到“铺垫”中单词语义表示和“笑点”中单词语义表示的关联信息，全局语义交互理解模块计算“铺垫”子句和“笑点”子句的关联信息。最后对局部和全局信息进行融合并对幽默等级进行识别。

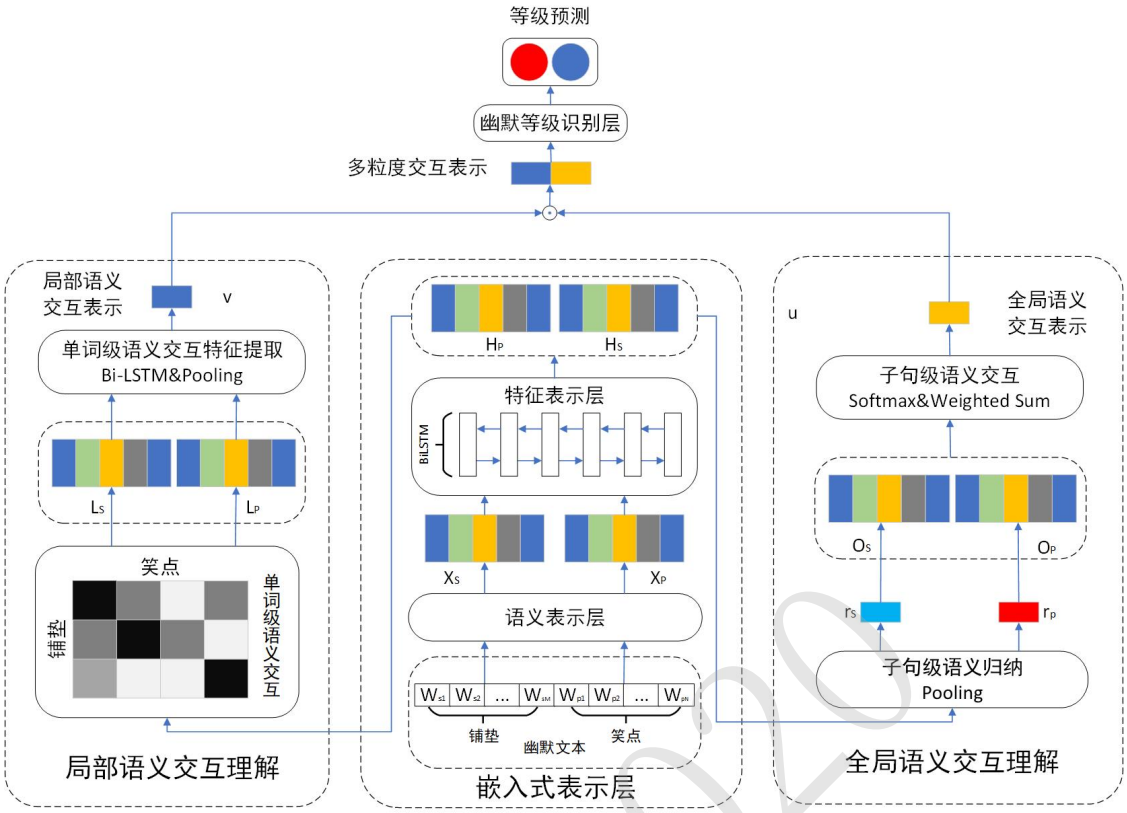


Figure 1: MSIN模型框架图

3.1 语义的嵌入式表示层

幽默是一种复杂的语言现象，一词多义等特征使得幽默特征表示和提取变得更加困难(Yang et al., 2015)。Xu等(2018)指出领域内和领域外的词嵌入表示的融合有助于文本分类模型性能的提升。目前还没有由幽默语料训练得到的词嵌入表示，而大规模的的词嵌入表示，如GloVe(Pennington et al., 2014)、BERT(Devlin et al., 2018)等，一般是利用通用语料或者新闻语料训练得到的。直接采用单一的词嵌入表示往往使得幽默等级识别的性能欠佳。此外，“铺垫”和“笑点”在幽默等级识别中起着不同的作用，将二者统一建模，不利于文本的幽默等级识别。因此，本文将幽默文本的“铺垫”和“笑点”分别建模，采用多个领域词嵌入表示进行融合，并采用Bi-LSTM提取两个部分的高维语义特征。

3.1.1 语义表示层

该层将“铺垫”和“笑点”中的每个单词映射到多个高维特征空间，并对其进行融合以获取有意义的语义表示。设幽默语句为$W = \{w_{s1}, w_{s2}, \ldots, w_{sM}, w_{p1}, \ldots, w_{pN}\}$，其“铺垫”为$W_S = \{w_{s1}, w_{s2}, \ldots, w_{sM}\}$，“笑点”为$W_P = \{w_{p1}, w_{p2}, \ldots, w_{pN}\}$，其中$w_i$为语句中的任一单词，$M + N$为句子总长度，$M$和$N$分别为“铺垫”和“笑点”的长度。将幽默语句中每个单词表示为K种低维稠密向量，并对同一单词多种向量进行拼接，得到单词的向量表示。则“铺垫”的向量表示为$X_S = \{x_{s1}, x_{s2}, \ldots, x_{sM}\} \in R^{D \times M}$，“笑点”的向量表示为$X_P = \{x_{p1}, x_{p2}, \ldots, x_{pN}\} \in R^{D \times N}$，$D$是词向量的维度，$D = D_1 + D_2 + \ldots + D_K$。

3.1.2 特征表示层

在该层中，模型利用Bi-LSTM分别提取“铺垫”和“笑点”子句的语义特征，作为幽默文本的“铺垫”和“笑点”的特征表示。LSTM(Hochreiter and Schmidhuber, 1997)能够对文本语义上的长距离依赖关系进行建模，而Bi-LSTM能够从正反两个方向提取潜在语义特征，并融合两部分的语义信息。在每个时间步t，正向和反向LSTM对输入词向量x_t的处理过程可以分别形式化的表示为：

$$\overrightarrow{h}_t = LSTM\left(h_{t-1}, x_t\right) \quad (1)$$

$$\overleftarrow{h}_t = LSTM\left(h_{t+1}, x_t\right) \tag{2}$$

其中，h_t表示t时刻的隐态向量，x_t为t时刻输入的词向量。将每个时间步正反两个方向的隐态向量拼接就得到Bi-LSTM单个时间步的输出，记作$h_t = \left[\overrightarrow{h}_t, \overleftarrow{h}_t\right] \in R^{2h}$，$h$表示隐态向量的维度。

特征表示层能够得到幽默文本的潜在语义特征，记为$H = [H_S, H_P] = [h_{s1}, h_{s2}, \ldots, h_{sM}, h_{p1}, \ldots, h_{pN}] \in R^{(M+N)\times 2h}$，其中$H_S$和$H_P$分别是铺垫和笑点的潜在语义表示。

3.2 交互语义特征提取层

Chen等(2016)指出不同粒度的语义单元及其交互信息能够有效地提高模型对文本语义的理解。铺垫和笑点作为两个语义单元，两者在不同粒度上相互作用，铺垫中单个词语及铺垫整体都会影响到笑点的语义表达，反之亦然。此外，Engelthaler等(2017)指出不同单词在句子中表现出不同的幽默程度。单词的语义信息与语句的幽默等级具有一定的相关性。

为使神经网络模型能够学习到单词和句子的语义信息，并且能够获取“铺垫”和“笑点”之间的关联信息，本文采用局部语义交互理解模块和全局语义交互理解模块对来自上层的潜在语义表示做处理。

3.2.1 局部语义交互理解模块

Yang等(2015)研究发现，在幽默文本中，不同词语的重要程度不同，当删除幽默文本中的某些词语后，文本的幽默程度下降甚至完全消失。本文采用局部语义交互理解模块从单词级别提取幽默文本的语义信息和语义关联信息。局部语义交互理解模块包括单词级语义交互层和单词级语义特征提取层。

单词级语义交互层 单词级语义交互层使用软对齐的方式获取“铺垫”和“笑点”的单词粒度语义交互表示。具体地讲，对来自特征表示层的铺垫和笑点的潜在语义表示H_S和H_P，该层首先将两者中每个单词对应向量两两之间做点乘，计算公式为：

$$e_{ij} = h_{si} * h_{pj} \tag{3}$$

可以得到铺垫和笑点的相似度矩阵$E = \{e_{ij} | i \in [1, M], j \in [1, N]\} \in R^{M\times N}$，其中$e_{ij}$表示铺垫中第$i$个单词和笑点中第$j$个单词的相似度。

然后，该层以加权求和的形式求出铺垫和笑点中每个单词对应的交互表示：

$$\widetilde{h}_{si} = \sum_{j=1}^{N} \frac{exp\left(e_{ij}\right)}{\sum_{k=1}^{N} exp\left(e_{ik}\right)} h_{pj}, \forall i \in [1, M] \tag{4}$$

$$\widetilde{h}_{pj} = \sum_{i=1}^{M} \frac{exp\left(e_{ij}\right)}{\sum_{k=1}^{M} exp\left(e_{kj}\right)} h_{si}, \forall j \in [1, N] \tag{5}$$

由上面两个式子可知，交互表示包括由铺垫表示的笑点和由笑点表示的铺垫，模型使用铺垫或者笑点中所有向量的加权和来得到对方每个单词的表示，以这种方式实现铺垫和笑点的交互。

最后，模型融合每部分文本各自的潜在语义表示及交互表示，得到两部分在该层的输出：

$$L_S = [l_{s1}, l_{s2}, \ldots, l_{sM}] \tag{6}$$

$$L_P = [l_{p1}, l_{p2}, \ldots, l_{pN}] \tag{7}$$

其中$L_S \in R^{M*8h}$、$L_P \in R^{N*8h}$分别是铺垫和笑点的单词级语义交互表示。本文使用如下方法对每个单词的潜在语义表示和交互表示进行融合：

$$l_{si} = [h_{si}, \widetilde{h}_{si}, h_{si} - \widetilde{h}_{si}, h_{si} * \widetilde{h}_{si}], \forall i \in [1, \ldots, M] \tag{8}$$

$$l_{pj} = [h_{pj}, \widetilde{h}_{pj}, h_{pj} - \widetilde{h}_{pj}, h_{pj} * \widetilde{h}_{pj}], \forall j \in [1, \ldots, N] \tag{9}$$

其中l_{si}、$l_{pj} \in R^{8h}$，

单词级语义交互特征提取层 该层对单词级交互信息进一步抽象，获取单词级语义交互特征。首先，该部分分别将L_s和L_p经过Bi-LSTM来提取单词级交互表示的高层特征，计算过程为：

$$G_S = [g_{s1}, g_{s2}, \ldots, g_{sM}] = Bi - LSTM(L_S) \tag{10}$$

$$G_P = [g_{p1}, g_{p2}, \ldots, g_{pM}] = Bi - LSTM(L_P) \tag{11}$$

其中$G_S \in R^{M \times 2h}$，$G_P \in R^{N \times 2h}$。分别对G_S和G_P做平均池化和最大池化，并将池化获得的四个向量拼接，最终得到幽默文本的局部语义交互特征向量v，计算过程如下：

$$v_{s,ave} = ave\_pool(G_S), v_{s,max} = max\_pool(G_S) \tag{12}$$

$$v_{p,ave} = ave\_pool(G_P), v_{p,max} = max\_pool(G_P) \tag{13}$$

$$v = [v_{s,ave}, v_{s,max}, v_{p,ave}, v_{p,max}] \tag{14}$$

3.2.2 全局语义交互理解模块

Ma等(2017)研究表明，对于文本中的不同语义单元，其单词的含义会受到其他语义单元的有影响。铺垫和笑点作为幽默文本的两个子句级语义单元，二者互相作用，对幽默等级识别产生重要影响。本文采用全局语义交互理解模块从子句级别提取幽默文本的语义信息和语义关联信息。全局语义交互理解模块包括子句级语义归纳层和子句级语义特征提取层。

子句级语义归纳层 该层分别对“铺垫”和“笑点”子句的上下文表示（H_S或者H_P）做平均池化和最大池化，将两部分拼接得到二者的归纳表示。公式如下：

$$r_{s,ave} = ave\_pool(H_S), r_{s,max} = max\_pool(H_S), r_s = [r_{s,ave}, r_{s,max}] \tag{15}$$

$$r_{p,ave} = ave\_pool(H_P), r_{p,max} = max\_pool(H_P), r_p = [r_{p,ave}, r_{p,max}] \tag{16}$$

得到的向量r_s、r_p通过全连接层把它们的维度投影到$2h$。

子句级语义交互层 该层对铺垫和笑点子句做交互，然后对交互信息进一步抽象，以获取全局的语义交互特征。首先，计算子句与各词之间的交互权重：

$$O_S = [o_{s1}, o_{s2}, ..., o_{sM}] = [h_{s1} * r_p, h_{s2} * r_p, \ldots, h_{sM} * r_p] \tag{17}$$

$$O_P = [o_{p1}, o_{p2}, ..., o_{pN}] = [h_{p1} * r_s, h_{p2} * r_s, \ldots, h_{pN} * r_s] \tag{18}$$

其中$O_S \in R^{M \times 2h}$，$O_P \in R^{N \times 2h}$。

然后，通过加权求和的方式获得两个子句的交互特征，并最终得到全局语义交互特征向量u：

$$u_s = \sum_{j=1}^{N} \frac{exp(o_{pj})}{\sum_{k=1}^{N} exp(o_{pk})} h_{pj} \tag{19}$$

$$u_p = \sum_{i=1}^{M} \frac{exp(o_{si})}{\sum_{k=1}^{M} exp(o_{sk})} h_{si} \tag{20}$$

$$u = [u_s, u_p] \tag{21}$$

其中u_s、$u_p \in R^{2h}$分别是铺垫和笑点的子句级别交互特征，将两者拼接得到u。

3.3 幽默等级识别层

该层由全连接层及softmax层组成。首先将局部和全局语义信息进行融合，然后通过全连接层和softmax层，得到幽默等级的概率分布，计算公式如下：

$$T = [v_{s,ave}, v_{s,max}, u_s, v_{p,ave}, v_{p,max}, u_p] \tag{22}$$

$$humor_{cls} = softmax\left(T\right) = \frac{e^{ti}}{\sum_{i=1}^{12h} e^{ti}} \tag{23}$$

其中$T \in R^{10h}$，$humor_{cls} \in R^C$是概率分布，C是幽默等级数量。本文采用交叉熵作为损失函数，其形式化表示如下：

$$loss = -\sum_{i=1}^{Num}\sum_{j=1}^{C} y_i^j \, log\widehat{y}_i^j + \lambda \parallel \theta \parallel^2 \tag{24}$$

其中，Num是训练集样本数，i是样本序号，j是标签序号，y_i^j是样本的真实标签类别，$\widehat{y}_i^j$是样本的预测标签类别，λ是L_2正则化项的超参数，θ是模型参数的集合。

4 实验结果

本节首先介绍了实验数据、评价指标、实验设置和基线方法，然后对比了基线方法和本文提出的MSIN方法的幽默等级识别性能，最后通过实验分析了本文提出方法的有效性。

4.1 实验数据与评价指标

Reddit数据集：该数据集由Weller等(2019)构建。幽默语句来自Reddit中带有“humor”标签的文本，采用众包方式对幽默语句的“铺垫”和“笑点”进行了标注，且对幽默语句的强弱进行了人工标注。数据集规模详见下表。

	弱幽默	强幽默	总计
训练集	9719	9719	19438
验证集	304	304	608
测试集	304	304	608

Table 2: Reddit幽默数据集统计信息

评价指标：为了便于和基线方法进行比较，本文采用了被广泛接受并应用于文本分类任务中的精确率（Acc）、准确率（P）、查全率（R）和F1 Score （F1）作为评价指标。

4.2 实验设置

词嵌入：在训练过程中，词嵌入表示分别采用了Glove以及Word2Vce(Mikolov et al., 2013)，维度均为300，词嵌入在训练的过程中固定。对未登录词使用$(-0.01, 0.01)$上的平均分布随机初始化。

超参数：在实验中，设置L_2正则化项的超参数$\lambda = 10^{-5}$，Bi-LSTM的神经元个数为128，CNN三个卷积核的尺寸分别为2、3和5，优化方法为Adam(Kingma and Ba, 2014)，Batch大小为64，dropout为0.5。为了防止过度拟合，在训练过程中使用了学习率衰减和早停机制。为了便于和基线模型对比，采用了Weller等(2019)对数据的划分。

4.3 基线方法

本文使用下述基线方法进行对比实验：

- Human(Weller and Seppi, 2019)*：人工预测结果。
- CNN(Weller and Seppi, 2019)*：采用CNN自动提取幽默语句的潜在语义特征并进行幽默等级识别。

https://nlp.stanford.edu/projects/glove/
https://code.google.com/archive/p/word2vec/

Method			Precision	Recall	F1_Score	Accuracy
Human(Weller et al.)*			-	-	-	66.30
分类任务	分类模型	CNN(Weller et al.)*	-	-	-	68.80
		CNN(Kim et al.)	68.22	68.85	68.18	68.16
		LSTM(Hochreiter et al.)	69.46	68.09	68.77	69.08
		Bi-LSTM-Attention	68.70	73.62	71.02	69.98
		Transformer(Weller et al.)*	-	-	-	72.40
		BERT(Devlin et al.)	72.06	74.67	73.34	72.86
推理任务	表示模型	CNN(Kim et al.)	69.78	69.87	69.41	69.42
		LSTM(Hochreiter et al.)	70.66	71.61	70.89	70.71
		BiLSTM-Attention	69.93	73.88	71.70	70.97
		BERT(Devlin et al.)	73.27	73.03	73.15	73.19
	交互模型	ESIM(Chen et al.)	73.38	70.72	72.03	72.53
		MSIN	**74.10**	**74.34**	**74.22**	**74.18**

Table 3: Reddit数据集实验结果

- CNN(Kim, 2014)：本文复现的基于CNN的方法，使用3种不同尺寸卷积核的CNN提取幽默文本特征进行幽默等级识别。
- LSTM(Hochreiter and Schmidhuber, 1997)：使用LSTM提取幽默特征并进行幽默等级识别。
- Bi-LSTM-Attention：使用双向LSTM和注意力机制提取幽默文本特征，并对幽默等级进行识别。
- Transformer(Weller and Seppi, 2019)*：使用基于transformer结构(Vaswani et al., 2017)的预训练模型对幽默文本整体做特征提取，以进行幽默等级识别。
- BERT(Devlin et al., 2018)：本文复现的基于BERT方法的结果，在任务语料上做微调然后进行幽默等级识别。
- ESIM(Chen et al., 2016)：只基于局部语义交互信息进行幽默等级识别。
- MSIN：本文提出的多粒度语义交互理解网络，综合使用语义嵌入、局部语义交互和全局语义交互进行幽默等级识别。

4.4 实验结果分析

本文在Reddit数据集上的实验结果见表3。表格整体分为三部分，第一部分为人工进行幽默等级识别的结果；第二部分采用之前研究的通用方法，将幽默等级识别视作文本分类任务，把幽默文本整体编码后进行分类；第三部分基于本文观点，即可将幽默等级识别任务视作自然语言推理任务，把幽默文本划分为铺垫和笑点两个语义部分，以这两部分作为模型的输入，使用表示型模型或交互型模型预识别文本蕴含的幽默等级。

在第二部分，本文的使用的CNN与Weller等(2019)的CNN结果相近，且两者均取得了明显好于人工预测的结果，证明了神经网络在幽默等级识别上的有效性。然而CNN由于卷积核尺寸固定，难以捕获长距离的语义关系，这对需要充分理解上下文的幽默等级识别任务是不利的。相比CNN，LSTM使用隐态向量捕获句子在长距离上的语义关系，可对时间序列进行有效建模，在数据集上取得了好于CNN的结果。然而LSTM是有偏倚的模型，后送入模型的信息会比先送入模型的信息拥有更大的权重，因此文本又使用Bi-LSTM+Attention进行改进。一方面，BiLSTM可以编码句子从前到后和从后到前两个方向上的信息，获取的特征更丰富，另一方面，Attention将所有时间步上的隐态向量赋予权重，让模型关注在文本分类过程中起关键作用的部分，缓解了由于LSTM的偏倚性造成的信息损失，因此模型相比LSTM取得了更好的结果。最后，本文使用BERT识别文本的幽默等级，其结果与Weller等(2019)使用Transformer的结果相近，并且两者均明显优于之前的模型。

Method	Precision	Recall	F1_Score	Accuracy
MSIN+Glove	73.20	70.07	71.60	72.20
MSIN+Word2Vec	73.33	72.37	72.85	73.03
MSIN+Both	**74.10**	**74.34**	**74.22**	**74.18**

Table 4: 不同词向量使用方式结果比较

Method	Precision	Recall	F1_Score	Accuracy
Word Level	72.64	73.36	73.00	72.86
Sub-sentence Level	70.68	73.22	71.91	71.41
MSIN	**74.10**	**74.34**	**74.22**	**74.18**

Table 5: 不同粒度实验结果比较

在第三部分，本文分别使用表示型和交互型两类模型进行幽默等级识别。

表示模型分别将铺垫和笑点编码为向量，然后将两向量与他们之间作差及点乘的结果拼接以捕获两部分的关系，最后基于拼接后的向量进行分类。为方便与第一部分的结果作比较，本文仍采用CNN、LSTM、Bi-LSTM-Attention和BERT四个模型。首先做内部比较，可以发现四个模型的结果依次递增，与第一部分的趋势保持一致；其次将表示模型与第一部分比较，发现四个模型的结果均高于第一部分中对应的模型，证明将幽默文本拆分为铺垫和笑点两部分，并让模型学习两部分之间的关系信息有助于幽默等级的识别。

在交互模型部分，本文使用ESIM与本文提出的MSIN进行比较。ESIM通过计算两部分文本之间单词的相似度矩阵来构建局部语义交互表示，并以此来推断前后文本的关系，在没有大量预训练知识的情况下，取得了略低于BERT的结果。本文提出的MSIN综合考虑交互过程中局部和全局语义信息的影响，取得了好于ESIM的最优结果。因此可以证明，相比表示模型，交互模型可以更好地捕捉到铺垫和笑点之间的关系；本文提出的多粒度语义交互理解模型融合单词和子句两个级别的交互信息，在幽默等级识别任务上取得了提升。

同时，本文进行消融实验，证明了词向量融合及多粒度交互两个结构的有效性，实验结果分别见表4和表5。表4前两行分别为只使用Glove和只使用Word2Vec的结果，第三行是使用融合词向量的结果，可以发现，融合之后效果更佳。表5前两行分别为只使用单词和子句交互的结果，第三行为融合两个粒度进行交互的结果，可以发现，多粒度交互网络取得了最优结果。

5 结论

本文将幽默文本划分为铺垫和笑点两部分，提出对两者之间的关系进行建模可以显著提升模型识别幽默等级的性能。基于这个观点，首先，本文在融合多种嵌入表示的基础上，从局部和全局两个粒度来对幽默中的语义关系进行理解和建模。其次，本文对幽默中“铺垫”和“笑点”两部分的关联信息做交互建模，从而实现充分挖掘铺垫和笑点之间的关系。最后，本文在Reddit幽默数据集上进行实验，取得了最优结果，同时结合消融实验证实了模型设计的有效性。在以后的工作中，我们将在幽默文本自动切分及基于铺垫的笑点文本生成方面做更多的探索。

参考文献

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv: Computation and Language*.

Francesco Barbieri and Horacio Saggion. 2014. Automatic detection of irony and humour in twitter. *Process Biochemistry*, 40(8):2637–2642.

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017. Datastories at semeval-2017 task 6: Siamese lstm with attention for humorous text comparison. pages 390–395.

Dario Bertero and Pascale Fung. 2016a. Deep learning of audio and language features for humor prediction. page 496.

Dario Bertero and Pascale Fung. 2016b. A long short-term memory framework for predicting humor in dialogues. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Vladislav Blinov, Valeria Bolotovabaranova, and Pavel Braslavski. 2019. Large dataset and language model fun-tuning for humor recognition. pages 4027–4032.

Andrew Cattle and Xiaojuan Ma. 2016. Effects of semantic relatedness between setups and punchlines in twitter hashtag games. pages 70–79.

Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2016. Enhanced lstm for natural language inference.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Tomas Engelthaler and Thomas T. Hills. 2017. Humor norms for 4,997 english words. *Behavior Research Methods*, 50(1):1–9.

Sepp Hochreiter and Jurgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Nabil Hossain, John Krumm, and Michael Gamon. 2019. "president vows to cut hair": Dataset and analysis of creative text editing for humorous headlines. *arXiv: Computation and Language*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. pages 1746–1751.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *Computer Science*.

Lizhen Liu, Donghai Zhang, and Wei Song. 2018a. Exploiting syntactic structures for humor recognition. pages 1875–1883.

Lizhen Liu, Donghai Zhang, and Wei Song. 2018b. Modeling sentiment association in discourse for humor recognition. 2:586–591.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Twenty-Sixth International Joint Conference on Artificial Intelligence*.

Rada Mihalcea and Carlo Strapparava. 2005. Making computers laugh: Investigations in automatic humor recognition. pages 531–538.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *Computer Science*.

Donald R. Morse. 2007. Use of humor to reduce stress and pain and enhance healing in the dental setting. *J N J Dent Assoc*, 78(4):32–36.

John Allen Paulos. 1980. *Mathematics and Humor*. University of Chicago Press.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Peter Potash, Alexey Romanov, and Anna Rumshisky. 2017. Semeval-2017 task 6: hashtagwars: Learning a sense of humor. In *International Workshop on Semantic Evaluation*.

Victor Raskin. 1979. Semantic mechanisms of humor. *Synthese Language Library*, 5(4):409–415.

J. M. SULS. 1972. A two-stage model for the appreciation of jokes and cartoons : An information-processing analysis. *Psychology of Humor*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. pages 5998–6008.

Orion Weller and Kevin D Seppi. 2019. Humor detection: A transformer gets the last laugh. pages 3619–3623.

Chris Westbury and Geoff Hollis. 2018. Wriggly, squiffy, lummox, and boobs: What makes some words funny? *Journal of Experimental Psychology General*, 148(1).

Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. 2018. Double embeddings and cnn-based sequence labeling for aspect extraction.

Diyi Yang, Alon Lavie, Chris Dyer, and Eduard Hovy. 2015. Humor recognition and humor anchor extraction. pages 2367–2376.

Renxian Zhang and Naishi Liu. 2014. Recognizing humor on twitter. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management*, pages 889–898.

Zhenjie Zhao, Andrew Cattle, Evangelos E Papalexakis, and Xiaojuan Ma. 2019. Embedding lexical features via tensor decomposition for small sample humor recognition. pages 6375–6380.

杨勇，杨亮，邹艳波，任鸽，樊小超. 2020. 基于音形义特征和层次注意力机制的幽默识别. 计算机工程, pages 1–12.

文本情感分析中重叠现象研究

娜仁图雅
首都师范大学
nrty0910@cnu.edu.cn

徐晓音
哈佛医学院
xxu@bwh.harvard.edu

摘要

汉语中的重叠现象丰富，文本情感分析任务中应该密切关注语篇空间内的重叠现象及其交互状态。本文就重叠在文本中的样态、特点及情感标记功能进行了理论探讨；重点就构词性重叠、结构性重叠的表现形式及情感语义进行了分析；据此研究的基础上，本文就重叠现象在文本情感分析上的实际应用从几个方面进行了讨论。

关键词： 重叠现象 ； 标记功能 ； 构词形式 ； 短语 ； 应用

A Study on Repetition in Text-based Sentiment Analysis

Tuya Naren
Capital Normal University
nrty0910@cnu.edu.cn

Xiaoyin Xu
Harvard Medical School
xxu@bwh.harvard.edu

Abstract

Repetition of characters and words is a rich phenomenon in Chinese and worth close attention to the repetition usage and its interactions with other parts of text in the task of text-based sentiment analysis. This work discussed and analyzed the appearance, characteristics, and sentiment marking of repetition in text, with a focus on the analysis of the repetition of words and repetition of sentence structure, and the sentiment presentation given by the above repetitions. Based on the analysis, we elaborated on the practical application of identifying and using repetitions in text to assist sentiment analysis.

Keywords: Repetition , Sentiment marking , Phrase formation , Short phrases , Application

1 引言

近年来文本情感分析获得了很大的发展, 并取得了广泛的应用。尤其是机器学习技术的发展，极大的提升了文本情感分析的准确率(赵妍妍2010)。比如Pang(2002)等人利用支持向量机在文本情感分析上达到了80%的准确率, Borbosa(2010)等人利用支持向量机分析推特上的文本情感, 获得了81%准确率。Kamps(2004)等则利用WordNet的词向量来计算新词与代名词之间的语义距离来推断文本中形容词的情感倾向。其它方法包括有朴素贝叶斯算法(梁柯2019)和随机森林法(张月梅2020)(刘勇2019)。除了传统的机器学习方法, 新技术尤其是深度学习算法也不断

被开发出来并应用在情感分析上, 比如基于长短期记忆神经网络LSTM的情感分析(任勉2018)(卢强2019)以及在LSTM上发展起来的注意力机制的情感分析(王伟2019)(卢玲2018)。在上述现有的文本情感分析中, 多数技术基于检测情感词来判断文字, 语句, 以及篇章文本中的情感极性和类别。周咏梅(2013)等构建了基于HowNet和Senti-WordNet的中文情感词典并考虑到情感词在不同语义环境下的情感倾向强度, 同时使用词典校对方法对划分的情感词倾向强度进行了优化。比如Shen(2009)等专门建立了否定词、程度副词、感叹词和情感词词典来帮助计算机识别文本的情感倾向性。王素格(2009)等提出了一种先基于五种汉语情感词词表构建一个中文情感词表, 然后利用该词表并采用线性加权组合的方法对句子情感进行分类的方法。黄高峰和周学广(2015)提出了一种基于词库权重、句式权重、修饰权重相结合的语句级细粒度情感倾向性分析算法, 在这个算法中构建了一个细粒度情感词库, 对其中的词语进行权重值的计算。张仰森(2018)等在对微博情感分析中则构建了一个包含情感词、程度副词、否定词、微博表情符号和常用网络用语的语料库帮助双重注意力模型对微博的情感分析。杨立公(2013)等提出了从文本颗粒度出发, 总结了文本情感分析方法在情感词抽取, 语料库和情感词典构建等几方面的发展。虽然情感词提供了一个明确的文字上的情感指示,但只依靠情感词来判断情感却有可能漏过其它表述情感的语言现象,进而影响情感分析的准确率。在这个文章里我们研究并讨论重叠作为一种语言现象在情感表达上的功能和如何通过检测重叠这种现象来增进情感分析的全面性和准确率。

重叠在情感意义的表达和交换过程中具有重要作用。能够使得情感信息在传递或被分析过程中，被更全面、更准确地解读出文本的情感意义。本文在计算语言的框架下, 从面向文本情感分析的角度出发，对篇章文本中存在的各种重叠现象进行分析和再认识，实际上也是对情感表达各个组成成分或语义资源的分析与再整合。索绪尔(1980)提出“在语言状态中,一切都以关系为基础的”。重叠与情感表达之间所形成的关系，在文本情感分析任务中应注意以下两点：一是，重叠与情感表达之间的关系存在并协互动与互补关系。二是，重叠内部及外部的多个不同的重叠之间的情感语义关系，均具有共同的情感语义指向。在关系认识的基础上提取规则，能够帮助机器综合运用多种模态分析手段实现情感识别的目的。重叠现象在语篇中的表现形式多样，主要体现为说话人对某一事物或现象的内涵性评价或感受的深化，具有很强的情感表达和描述能力。重叠与情感的变化有直接或间接的关系。词语及小句两者重叠所处的句法位置能够带来情感表达(及强度)的差异，进而增加了文本情感分析的难度。当前的研究，尚欠缺对重叠现象的研究，也没有将重叠作为情感分析的一个判断指标，因此，值得对重叠现象进行深入细致的研究。

2 重叠样态及标记功能

文本情感分析各个层级的文本中，当出现重叠性的语言形式时，情感类型及强度都可能发生变化，具体表现为情感类型的凸显和情感强度的加深。原因在于重叠所形成的是共指表达，能够层层推进情感，所形成的情感语义潜势都指向情感类型的确定和强度的深化。

2.1 重叠样态

重叠是形成文本情感的诸多语言表现形式之一。文本中的词汇、句子、段落及篇章这四个层级中都有可能存在重叠性的语言信息。特别是词汇重叠极为常见，在不同层级的文本中都有分布。

2.1.1 样态

文本中的语言表情形式是多样的，大体可分为内隐和显性的。汉语中的重叠现象，由于形式工整有规则可循，因此属于显性形式。重叠具有情感表达的功能，我们可以从其在文本中的普遍性和典型性样态来考察。例:

推开门一看，嗬！好大的雪啊！山川、树木、房屋，全部罩上了一层①厚厚的雪，万里江山变成了粉妆玉砌的世界。落光叶子的柳树上，挂满了②毛茸茸、③亮晶晶的银条儿；冬夏常青的松树和柏树，堆满了④蓬松松、⑤沉甸甸的雪球。一阵风吹来，树枝⑥轻轻地摇晃，银条儿和雪球儿⑦簌簌地落下来，玉屑似的雪末儿随风飘扬，映着清晨的阳光，显出⑧一道道五光十色的彩虹。《第一场雪·峻青》

文本情感分析中通常是以情感词作为主要评定指标，从该本文示例来看，含情感词“好、粉妆玉砌、五光十色”以及表情感的语气词“嗬、啊”。如果只是据此评定标准来计算该段文本的情

感的话，则显然不够充分，因为该文本中还含有8个重叠形式。这些重叠形式大体可归结为形容词的“AA式、ABB式”以及数量短语的“ABB式”。这些重叠是由几个意义相关或相近、相同或相似的结构，前后相连，经重叠，层层推进共同形成了一个与赞美、喜爱相关的情感表达链条。这些重叠形式在文本中的分布表明其实际上承担着情感表达的职能。该文本中的多个重叠与目标情感是个体与整体的关系。即文本的目标情感，是由包括重叠形式在内的诸多表情成分组织传送实现的。重叠产生的情感语义指向目标情感，当文本中出现多个重叠时，表示相互连接，共同推进和确立情感，表现为句或段落中的交集和交互状态。

2.1.2 特征

重叠形式的使用范围不限于词汇、句子、段落或篇章，其在文本中的分布可能不只局限于某个句子或某个段落当中，甚至可以跨越段际。其在文本中的使用样态来看，呈现出如下情感表达特征：(1)通常具有附加值。可使得情感表达的类型或强度在各种形式的重叠中发生变化，赋予更多的情感价值。(2)往往是临时形式。由于重叠形成了一个新的表达结构或单位，因此具备有限的能产性。(3)重叠的往往是凸显的。重叠的往往是可接受度高的语言成分，重叠后不影响对句子的理解，重叠的就是所要凸显的。(4)重叠形式有限制。重叠的形成受语言结构规则的限制，是建立在基本结构框架基础上的有规则的重叠。(5)具有共同的句法特征。重叠的目的是用以表示一个完整情感语义，句法上基本能够实现情感表达的自足。

2.2 标记功能

文本中重叠形式的使用具有情感标记功能。话语人在传递情感话语信息时，有意识地使用带有话语标记性质的重叠手段，实际上是对读者或听话人的语境假设加以限制和制约，其目的不在于改变话语所表达的语义内容或命题内容，而是要促进文本话语的理解和推进情感信息识别。一方面可以减少文本话语理解的阻力，另一方面则可推进情感信息的识别，提高读者或听话人理解文本的情感信息的效度，同时减少文本情感误读的出现，保证言语交际的正常进行。

2.2.1 情感强度加强标记

重叠所形成的情感强度潜势大于一个词或句子，是文本中情感强度被加强的指标之一。如果以零增为起点，定数每重叠或增加一次，便表示定数的增加，因此重叠具有程度加深的语义。例：

a 我懂得母亲没有说完的话。妹妹也懂。我俩在一块儿，要①好好儿活.....《秋天的怀念·史铁生》

b 母亲扑过来抓住我的手，忍住哭声说：“咱娘儿俩在一块儿，①好好儿活，②好好儿活.....”《秋天的怀念·史铁生》

上述两例来自同一作者、同一文章。a句中“好好儿”是“好”的“AA儿”式重叠。重叠后“好”具有了“殷切的要求或希望”的情感含义，形成了“好好儿活”这样一个全量的情感表达；b句中“好好儿活”出现两次，在全量的基础上实现定数的叠加，即情感语义及强度的再次叠加，因此情感强度明显高于a。进入到文本中，我们可以据此捕捉到从a到b的情感强度的变化。这是运算整合及语用推理的结果，所形成的特定的重叠性的结构，其语义特征是对原有结构特征的承继和发展(扩大)，即大于原结构的情感语义特征，其潜在的情感强度大于一个简单句或复句，带来了一定的增值效应。因重叠而形成的结构化方式和特征，能够随着具体表达式的多次强化而调节情感表达强度。因此，重叠是文本中情感强度被加强的标记。

2.2.2 情感语义向心标记

语义指向是指句法结构中甲成分与乙成分有语义联系及语义所指的方向。文本中重叠的信息存在着一定的情感语义方向。重叠形式的使用其语义指向都应是向心的。如以下两种情况：(1)共指性重叠。是指句子中的重叠成分构成了共指表达，形成了联合指意(表情)，能量增强，同时也意味着放弃其他指意可能。例：

看，①像牛毛，②像花针，③像细丝，密密地斜织着，人家屋顶上全笼着一层薄烟。《春·朱自清》“牛毛、花针、细丝”共指春天连绵的细雨。假设：X=雨；Y1=牛毛；Y2=花针；Y3=细丝；Z=细。无论是“牛毛”、“花针”还是“细丝”,都是为了说明春雨的“细”。“细”是“春雨”所具有的状态，即X、Y、Z是一种共指表达。将三个“比喻”辞格重叠

使用，使得三个喻体共享了本体的状态义素特征，同时排除其它特征，从而使得语义方向更加明确。

(2)非共指性重叠。是指句子中的重叠成分有分别由有交叉，结构上不是完全对应。非共指性的重叠同样能够将复杂的、强度高的情感展现出来。例：

它的颜色非常鲜艳。①头上的羽毛像橄榄色的头巾，绣满了翠绿色的花纹。②背上的羽毛像浅绿色的外衣。③腹部的羽毛像赤褐色的衬衫。《翠鸟·菁莽》

该例句中三个比喻小句的重复使用，不仅更具描述性，同时也附带有喜爱的情感。“头上的羽毛、背上的羽毛、腹部的羽毛”，是非共指性的本体，三个比喻小句的本体虽各有所指，却又共同指向“颜色非常艳丽”，因此非共指性的重叠仍然是具有语义向心力的。

综上，文本中的各个意义单元本身具有多种指意(表情)潜势。重叠的使用主要是出于所要表达的情感与强度形成匹配，意味着话语人力图实现表达的终极目的，具有优先实现指向核心情感的潜势，因此是语义向心标记。

2.2.3 情感动态发展标记

篇章文本中情感呈现点状分布，具有渐进性，且可跨过不同段落。经重叠形成的语义结构呈现出可操作性及整合性的特征，是一种动态性语义结构，能够充分展现人物(单个或多个人物)和事件的情感动态发展过程。例：

他们一队一队按照次序走，走过正对天安门的白石桥前，就举起灯笼火把，高声欢呼“毛主席万岁！”“毛主席万岁！”毛主席在城楼上主席台前边，向前探着身子，不断地向群众挥手，不断地高呼“人民万岁！”“同志们万岁！”《开国大典·李普》

该文本示例中，多个结构相同或近似的重叠(或反复)将欢腾的场面和不同人物的愉悦、热爱的情感不断推进，重叠的小句之间是递进关系。重叠作为一种情感信息组织的手段贯穿于文本之中，能够表达情感信息的动态发展及延续，是文本情感动态发展的标记。对其进行识别，可以据其自然段在文本进展中所处的位置为坐标(10%，20%...90%，100%)来分析每个自然段以及其中人物的情感的单独变化和各个人物相互的情感变化来分析总结文本中每个人物的情感基准、变化趋势和相对作用。

3 构形性重叠

汉语是语义型的语言，缺少形态变化，但是某些词类或个别词汇却存在构形上的变化，这种变化主要表现为重叠。“重叠之前的形式称为‘基式’，重叠之后的形式称为‘重叠式’”，我们称之为“构形性重叠”。构形性重叠形式的使用，往往能够用以表达喜爱、厌恶等情感类型，同时又与指小、增量形式指称联系在一起，以表示不同的情感类型和极性强度。

3.1 形容词重叠

有些性质形容词可以重叠，用以表示形状或程度的加深或适中，具有情感表达的功能，能够反映人的高强度的情感认识或体验。

3.1.1 单音节重叠

单音节的重叠形式主要有“AA”、”AA的”、”AA儿”三种，如“弯弯、绿绿的、好好儿”。例：

a 一位老人细细端详着毛主席，说：“这位首长，好像在哪儿见过。在哪儿呢？《毛主席在花山·翟志刚》

b 小草偷偷地从土地里钻出来，嫩嫩的，绿绿的。《春·朱自清》

c 我懂得母亲没有说完的话。妹妹也懂。我俩在一块儿，要①好好儿活.....《秋天的怀念·史铁生》

上述例句经单音节重叠后，或多或少地都具有了情感表达的功能。a句里的“细细”将认真、仔细及疑惑的程度加强；b句则具有了喜爱的情感；c句体现为一种殷切的希望。另外，两个单音节能够联合重叠，形成AABB式，这两个单音节词汇之间互为反义或近义，通常用来表达话语人的对某人某事的概括性的评价和认知。例：

a 这时，又从上边的河湾湾里漂来了好多大大小小的蒲团儿, ...。《小翠鸟打房子·王季》

b 左面山坡上高高低低一幢一幢土红色和灰色的三层楼房吸引着我的眼光。《MV旧的春天·巴金》

例句中的重叠是单音节的“大、小”, “高、低”联合形成的重叠，表达的是话语人对事物的概括性一种认识，具有指多增量的情感语义。

3.1.2 双音节重叠

双音节重叠主要有“ABB”、”AABB”、”BAA”三种，如“红彤彤、曲曲折折、红通通”。例:

a 没有月光的晚上，这路上阴森森的，有些怕人。《荷塘月色·朱自清》

b 店主人高高兴兴地捧出了笔砚纸张。《文天祥·沈起炜》

c 警卫员只好接过茶叶筒，端端正正地向毛主席敬了个礼。《毛主席在花山·翟志刚》

a句式用了ABB式重叠，渲染了周遭阴森之感，“阴森”本身就是负面情感词，重叠后在此处表达了一种更加强烈的恐惧；b句使用的是AABB式重叠，“高兴”是正面情感词，本身强度就高，重叠后其所表达的仍然是高兴的情感，但强度却达到最高；c句中使用的是“端正”的AABB式重叠，表达了警卫员对毛主席的敬重的情感。所以说，AABB 式重叠可以表达一时一地的态度，这也是双音节重叠的一个情感语义特点。

3.2 名词重叠

名词重叠式主要有“AA”, “ABAB”, “AABB”和“ABB”式。重叠之后名词一般具有赞同、喜爱、亲昵等(与认同类相关的)情感语义。例:

a 人人都有饭吃，家家都有房住。

b 这孩子猴精猴精的。

c 我爱家乡的山山水水，我就想用我自己学到的知识去把它变得更美好。《你怎么也想不到·路遥》

d 可是从此以后，每逢看见蜜蜂，感情上①疙疙瘩瘩的，总不怎么舒服。《荔枝蜜·杨朔》

e ①青线线(那个)②蓝线线，蓝格英英(的)彩，生下一个③兰花花，实实的爱死人。《兰花花·民歌》

a句的增加了“所有、每一”的含义，是对“每个人、所有家庭”的评价，一种增量指称；b句的“猴精”是名词，重叠后语义发生变化，用以表示“机灵、聪敏”的含义，是指小形式指称。表达的是与“夸赞”相关的情感类型，且情感强度很高；c句“山山水水”是单音节名词联合重叠所形成的AABB式，具有繁多的含义，凸显了对家乡热爱的情感；d句“疙瘩”一词原本没有情感，经过重叠而具有了情感表述的功能，用以表达“不快”这一情感类型，是一种增量形式指称；e句是陕北方言样本，含有多个名词的重叠，使得兰花花这一人物形象跃然纸上，表达的是“喜爱”的情感，属于增量形式指称。

3.3 动词重叠

汉语中有些动词本身就含有情感义，如“喜欢、祈盼、虐打”等；而有些则不含，如“看、听、说”等。同样的，有些动词能重叠，有些则不能。如某些用来表示动作行为的动词就能够发生重叠，并具有了表短时或轻微的含义。其重叠形式，单音节动词为AA式，如“想想”；双音节动词为AABB式，如：“研究研究”。例:

a 织云抬起眼光，看看远远的蓝天。《我们的歌·赵淑侠》

b 贾芹走进书房，只见那些下人指指点点，不知说什么。《红楼梦·第93回》

c 她也笑了，坐在我身边，絮絮叨叨地说着：“看完菊花，咱们就去‘仿膳’，你小时候最爱吃那儿的豌豆黄儿。《秋天的怀念·史铁生》

重叠后的动词会被赋予情感语义。a句中的“看”本身不具有情感表达功能，经AA式重叠具有了短时的含义；b句中的“指点”经重叠后发生了语义上的改变，具有了贬义；c句“絮叨”一词具有负向的情感，重叠形式“絮絮叨叨”不仅具有指多的含义，同时加深了反感的情感强度。此外，有些动词的AA式重叠使用场景比较固定，如“红旗飘飘、蝴蝶飞飞”等，表示的是动作的反复和延长，而不是短时或轻微的含义。

3.4 数词重叠

汉语中可进行重叠的数词非常有限，最常见的数词重叠形式是“AA”式和“AABB”式。其中“一、九、千”可重叠为“AA式”；“二、三、千、万”可重叠为“AABB式”。例：

a 心似双丝网，中有千千结。

b 九九八十一难。

c 天上五颜六色的火花结成彩，地上千千万万的灯火一片红。《开国大典·翟志刚》

d 问今是何世，乃不知有汉，无论魏晋。此人一一为具言，所闻皆叹惋。《桃花源记·陶渊明》

e 等到得不着票子，便不免有了三三两两的怨声了。《旅行杂记·朱自清》

a、b句中的数词经重叠后具有指多增量的含义，表示心结多、磨难多。句子情感强度被增强；c句则具有了指多增量的含义；d句数词“一”重叠之后意义和功能均发生改变，不仅用以指多，还具有逐一地，详细、详尽地对某人某事或某物的进行介绍的含义，多用于动词前，具有正向的情感，情感强度被增强；e句中“三、两”本身就是指少数词，重叠后具有了零散的语义。需要说明的是：AABB式中只有这一例是减量指少，情感强度被削弱。其它的“AA式”或“AABB式”都是指多增量，情感强度被增强。数词重叠带来的词义增加。重叠后的数词除了“三三两两”具有指少减量的语义之外，其他赋予“指多增量”的语义。数词重叠可使得句子的情感强度增强或削弱，因此具有情感表达功能。

3.5 量词重叠

量词重叠不仅增加了意义，还增加了功能。李宇明(1996)认为AA 式是量词重叠形式。单音节量词大多可以重叠，如“个个、件件、条条”等。量词重叠之后增加了“每一、逐一”或“多”的含义。例：

a 个个都是好样的。

b 我家洗砚池头树，朵朵花开淡墨痕。《墨梅·王冕》

c 层层的叶子中间，零星地点缀着些白花，有袅娜地开着的，有羞涩地打着朵儿的。《荷塘月色·朱自清》

上述例句的重叠之后，增加了“每”和“多”的含义。a句表达了对每一个人的赞赏；b句则增加了繁多的语义，表达的是诗人对梅花的喜爱之情；c句则增建了逐一的含义，这些量词的重叠使得被描述的对象更加形象化和可观可感。

综上所述，主要是就形容词、名词、动词、数词、量词的所形成的重叠进行了分析。重叠形式的使用在情感表现力上，增量形式指称的特点较为突显，具有程度更深、更细腻的特点；在语言使用体验上，生动而形象，能够呈现出更多的情感表达效果，且语用接受程度高。

4 结构性重叠

文本情感分析不仅要关注语言系统中稳定的、规则的、典型的语言现象，还应关注特殊的、偶发的、创新性的塑造语言结构的现象。结构性的重叠，可视为特殊的、创新性的情感表达手段，主要表现在比况短语、“的”字词组重叠、数量短语、状中短语及定中短语的重叠上。

4.1 比况短语重叠

比况，实际上是比喻的灵活多样用法之一。本体可不出现，喻词及喻体出现，重叠排列成结构相似或相同、互相映衬的平行结构(或句式)来表达话语人的思想情感。比况结构的重叠使用，将使得句子形象生动、情感类型更加明确和凸显，情感强度高。例：

a 你①像雾②像雨又③像风。

b 野花遍地是：杂样儿，有名字的，没名字的，散在草丛里，①像眼睛，②像星星，还眨呀眨的。《春·朱自清》

c 有的①像峰峦，②像河流，③像雄狮，④像奔马.....它们有时把天空点缀得很美丽，有时又把天空笼罩得很阴森。《看云识天气·朱泳》

a句是三个比况联合形成的比况结构重叠，将人比作“雾、雨、风”，表达的是话语人为之着迷的情感；b句有两个比况结构，把野花人格化，使它们具有人的情感，形成的拟人性的表述，用以表达作者强烈的喜爱的情感。c句的喻体由三个比况结构重叠构成，将云比作是“峰峦、河流、雄狮、奔马”，表达的是作者对云的形态变幻的认识，同样是一种与喜爱相关的情感。

4.2 “的”字短语重叠

“的”字短语由助词“的”附着在实词或短语后组成，用来指称人或事物，属于名词性短语，其功能相当于一个名词，如“大的、卖菜的、跑堂的”。其重叠使用通常是话语人为了进行详尽地介绍而采用的一种表达方式，是增量指称，对小句情感强度具有补强作用。例：

a 夜市上好不热闹，有①摆摊的、②卖药的和③赚吆喝的.....

b 我用手拨开草一看，原来青草下边藏着满满一层小花，①白的、②黄的、③紫的。《花的勇气·冯骥才》

c 我们继续拍掌，树上就变得热闹了，到处都是鸟声，到处都是鸟影。①大的、②水的、③花的、④黑的，有的站在树枝上叫，有的飞起来，有的在扑翅膀。《鸟的天堂·巴金》

上述例句大都是三个以上“的”字短语的连续使用，我们视此种语言表达形式为“的”字短语重叠。“的”字短语本身不具情感意义，经重叠后其作用主要是罗列，增加了指多的含义，因此具有间接性地补强句子情感的作用。其中，a句“的”字短语的重叠主要是对前文夜市的热闹程度的补足和加强；b句重叠的目的不单是罗列，同时也是希望给读者以身临其境之感，间接表达了作者对“满满一层小花儿”的喜爱；c句通过“的”字短语的重叠将鸟的种类颜色等进行了大致的介绍，目的是以此保持与“鸟的天堂”这一主题及整体情感语义上的相和谐，表达的是对奇特幽美和别具洞天的“鸟的天堂”的赞美。

4.3 数量短语重叠

通常量词与数词组成数量词组产生重叠，形成：一AA式、一A一A式，如“一件件、一本一本”。具有“繁多”和“按某种方式或状态进行”的含义。两种形式可互换使用。例：

a 月亮越升越高，穿过(一缕一缕)轻纱似的微云。《月光曲·杨爽》

b 玉屑似的雪末儿随风飘扬，映着清晨的阳光，显出(一道道)五光十色的彩虹。《第一场雪·峻青》

c 白云来了，(一大团一大团的，)从祖父的头上飘过，好像要压到了祖父的草帽上。《祖父的园子·萧红》

a、b 句中的数量词组的重叠，具有指多增量的功能，这种指多表现出的是与喜爱或认同等相关的情感。在情感表达上, 如果省略例句中括号中的重叠,比如a与原句相比较, 我们可以体会到“一缕一缕”的使用更加强调了作者轻松平静的心态。同样的, 把b句中的重叠去掉便会降低作者对雪欣赏赞许的强度。c句中的”一大团一大团”则凸显了云的状态，增强了作者对白云这样一个具正面极性事物的倾向性看法, 如果将其去掉，便觉少了轻松愉快的情绪, 虽还保有正面的极性, 却更偏近于中性了。

4.4 状中短语重叠

状中短语由状语和中心语组成。该结构的重叠形式丰富多样，能够表达认同、赞美、喜爱等情感，经重叠后的状中结构能够赋予句子更高的情感强度。例:

a 小鱼儿在荷叶下笑嘻嘻地游来游去，捧起①一朵朵②很美很美的水花。《荷叶圆圆·胡木仁》

b 昙花开放的声音是短促的，茶花开放的声音是悠长的，不管短促或悠长，都是①那么动听，②那么迷人。《花开的声音·爱玲》

c 他们正用劳力建设自己的生活，实际也是在酿蜜－－①为自己，②为别人，③也为后世子孙酿造生活的蜜。《荔枝蜜·杨朔》

a句中含有两项重叠，其中②是程度副词和形容词构成的状中结构的重叠，联合加强了“美”的程度和状态，表达的是喜爱的情感；b句是代词和形容词构成的状中结构的重叠，表达的是赞美的情感；c句是由介词短语和形容词构成的状中结构的重叠。该句当中不含情感词，假设只有一项状中结构的条件下，该句是不含情感的。该句中含有三项状中结构，其重叠使用赋予了句子颂扬的情感。

4.5 定中短语重叠

定中短语由定语和中心语构成，两者是修饰关系，修饰语一般由形容词和名词充当，中心语是被修饰的对象。修饰语和中心语之间有时用“的”来连接。定中短语是不自由短语，通常不能独立使用，只能做主语宾语。其在文本中有连续反复使用的情况，主要体现为对主语或宾语的凸显，凡是重叠的就是凸显的，因此，能够赋予句子情感。例:

a 小河是娴静的，宛如明镜一般，倒映着①红色的花、②绿色的树。《家乡的小河·佚名》

b ①青的草，②绿的叶，各色鲜艳的花，都像赶集似的聚拢来，形成了光彩夺目的春天。《燕子·郑振铎》

c ①秋的味，②秋的色，③秋的意境与姿态，总看不饱，尝不透，赏玩不到十足。《故都的秋·郁达夫》

定中短语在文本中的重叠使用，意味着修饰性词语出现的几率越高，修饰性的词语通常具有较强的形象色彩及情感色彩，因此实现情感表达的可能性就越大。a 、c两句重叠的使用，不仅限定了对象，同时具有很强的画面感。因此，a句凸显了小河倒影的美感，b句的重叠则与“光彩夺目的春天”相匹配，表达出对春的赞美；c句由三个定中结构反复使用，联合形成一个大主语，修饰语及中心语之间具有领属关系，因此是对主体对象的凸显。相比之下，用“秋天”来替换做主语也是可以的，但却丧失了主语所承担的情感功能。因此定中短语重叠具有赋予句子情感作用。

综上分析，句子中的结构性重叠现象能够用以表达情感。董秀芳(2016)认为“词组的重叠是一种句法上的重叠，实际上是同义并列。这种手段所造成的形式都有描摹状态的意味，这种描摹带上了加强程度的含义，平行于词法重叠中的增量功能，而这种增量的描摹功能也是取决于说话人的认识，因此也具有明显的主观性。” 主观性是实现情感表达的必要因素，因此，在文本情感分析中结构性重叠具有情感表达的功能。

5 重叠在情感分析上的应用

在前面几节里我们总结了重叠的几种表现形式。可以看出来，在情感表达上，重叠不仅凸显了情感阶段的出现而且加强了情感表达的强度和细腻化，另外，重叠也把一些中性词变的有了情感色彩。例如，“冷”本身不具有感情色彩，重叠后“冷冷地”则表现了负面情感。类似的，“热”作为单独一个字是中性的，当它与虚词“乎”结合成“热乎乎”就变成了一个ABB形式的正向表达，同理，“热热闹闹”就变成了一个AABB式的正面情感表达。因此，通过对重叠现象的分析，我们可以提取一些规律来帮助计算机识别情感, 从而可以从几个方面来有效地提高文本情感分析的准确性。

第一个方面就是通过检测字词尺度上的重叠。根据上述的讨论，我们看到重叠的有效长度有不同的尺度，既可能以重复单个字或单个词为单元的形式出现，如AA或ABAB的形式，亦或AABB和ABCABC形式；也可能以重复短语或短句为单元的形式出现。在以字词为尺度的重叠中，我们可以设置算法来寻找AA或ABAB形式的语言现象，如采用一个带记忆的滑动窗口(moving window)w来检测AA或ABAB。在这个过程中，假设w的记忆长度为k, 在位置n的输入为$w(n)$, 则算法检测是否

$$w(n) = w(n-1) \tag{1}$$

来检测AA式的重叠；同样的, 算法检测是否

$$\begin{aligned} w(n) &= w(n-1) &(2)\\ w(n-2) &= w(n-3) &(3)\end{aligned}$$

并

$$w(n) \neq w(n-2) \tag{4}$$

来判断AABB式的重叠；以及检测是否

$$\begin{aligned} w(n) &= w(n-2) &(5)\\ w(n-1) &= w(n-3) &(6)\end{aligned}$$

来判断ABAB式的重叠。同时我们还需要考察重叠的组合用法，如在例句“毛茸茸、亮晶晶的银条儿“出现的ABB, CDD形式的重叠，那么这个时候就需要把ABB，CDD作为一个整体考虑来判断其表述的情感极性和强度。

第二个方面则检测短语或短句尺度上的重叠。实现短语或短句的重叠检测需要设计灵活的算法在检出句式结构的同一性的基础上准确地检测语义上的重复或类似性，比如例句”漓江的水真静啊，静得让你感觉不到它在流动；漓江的水真清啊，清得可以看见江底的沙石；漓江的水真绿啊，绿得仿佛那是一块无瑕的翡翠。” 算法可以设计成在一个长度为K的记忆片段中，以标点符号为隔断定义短句，在当前的第n个短句的位置上首先分析短句的结构，如动宾结构，主谓结构，或动状结构，然后搜索第$(n-1),(n-2),\cdots,(n-j)$个短句知道第$(n-j-1)$个短句的结构与当前短句的结构不同式停止。这时我们可以判断从第$(n-j)$到第n个短句具有相同的结构。下一步我们可以设计算法判断每一个短句的情感极性和类别。

第三个方面则需要针对不同尺度的重叠进行综合整理。在检测分析字词尺度和短句尺度上的重叠现象的同时，算法还需要能够处理复杂的重叠现象，比如以下例句中出现的嵌套式的重叠，“一会儿红彤彤的，一会儿金灿灿的，一会儿半紫半黄，一会儿半灰半百合色”。在这个例句里我们既观察到短句结构上的重叠，“一会儿.....”，同时还可以看到在第一和第二个短句出现ABB式的字词重叠，即“红彤彤”“金灿灿”, 而在第三和第四个短句中出现混合ABAB式的字词重叠, 即”半紫半黄””半灰半百合色”。在这种语言现象中，我们需要设计算法使其正确解读“红彤彤”“金灿灿”隶属于短句尺度上的重叠，用来表示同一个情感场景，而不是两个分别的情感。如果我们用一个矢量来代表短句尺度的情感表达，针对上述例句我们可以用四个矢量分别代表其中的短句, 如F_1=”一会儿红彤彤的”, F_2=”一会儿金灿灿的”, F_3=”一会儿半紫半黄”, F_4=”一会儿半灰半百合色”。而其中F_1又包含一个字词尺度的情感表达f_1 = ”红彤彤”, F_2包含f_2=”金灿灿”。同样的, F_3包含了f_3=”半紫半黄”, F_4包含有f_4=”半灰半百合色”。在情感识别上, F_1到F_4的短句重叠可以指示这是一个富有情感的表述, 但仅凭短句重叠还不能判断情感极性, 当结合了f_1到f_4的字词重叠, 特别f_1和f_2, 我们就可以判断出这是一个正面的情感极性。在这个例子里我们还可以观察到重叠现象之间存在的互相修饰作用, 亦即虽然F_3, F_4中的”一会儿半紫半黄，一会儿半灰半百合色”本身是中性的表述, 但结合了F_1和F_2的正面情感极性后, 我们就可以判断F_3和F_4也变成了或参加了正面情感的表述。而这种通过短句重叠来赋予中性表述以极性的功能在依据情感词来表述情感的文句是罕见的。

第四个方面在于准确判断重叠在单独使用时赋予文本的情感含义。根据前面的分析和举例, 我们注意到在没有情感词的情况下, 重叠可以单独表达情感, 如”实际也是在酿蜜－－为自己，为别人，也为后世子孙酿造生活的蜜”中，重叠的状中结构将原本是中性的”实际也是在酿蜜”变成了正面的赞美。这种情形如果只是检测词库里的情感词很可能无法判定句子的情感属性, 而

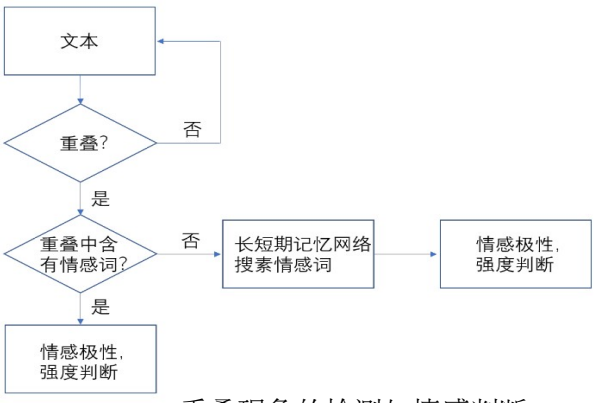


Figure 1: 重叠现象的检测与情感判断

当我们设计一个算法检测出这一重叠时, 算法便可以向前和向后搜索到"酿蜜"和"生活的蜜"来综合考虑一个重复强调"酿蜜"的句子应该是含有正面情感的。从这个方面来考虑, 现有的情感分析方法包括深度学习方法可以在情感词的基础上拓展对情感表现的定义, 将重叠作为一种表达情感的现象包括在训练和验证过程中。从计算语言的角度出发，根据重叠在情感表达上的多样性，我们可以设计类似于图1的计算流程来首先检测重叠是否出现，当检测到重叠时，须进一步判断重叠的语形本身是否为情感词，以及重叠语形长度附近是否与情感词存在共现的状况。如果含有情感词，即视为共现，可以直接据情感词的标注状况来判断它所表现的情感极性和强度。如果不含，即非共现，则可通过机器学习结合情感词来判断整个重叠现象的情感极性和强度。在这一步需要记录与重叠相关的情感词的位置，因为情感词可能出现在重叠之前或之后，所以可以利用长短期记忆网络或类似的具有记忆功能的机器学习技术来捕捉到情感词和重叠直接的关联并加以分析。例如，在长短期记忆网络中，根据重叠的表现形式我们可以用来加强不同长度记忆的强度。 综合以上几个方面的讨论, 我们可以得出结论, 重叠作为一种语言现象, 有着使用上的多样性和增进情感表达的作用, 既可以与情感词联合使用也可以单独使用。对重叠的准确检测和正确分析可以有效地提高自动化的情感分析的表现。在深度学习被广泛用于文本分析的情景下，文本中检测出的重叠现象可以用于加强对神经网络的训练。

6 总结

本文中我们从重叠的语言现象出发, 探讨和归纳整理了不同形式和尺度上的重叠现象, 既有最简单的单音节字的重叠, 也有词组的重叠, 还有复杂的短语和短句的重叠。通过分析, 我们发现重叠对文本的情感表达有着多方面的作用, 重叠既可以与情感词共现配合来显性的表达情感，也可以单独出现来隐性的表达情感。在这种情况下，通常需要结合上下文才能准确判断重叠所表现的情感。从文本情感分析视角来看，重叠现象可以作为一个情感表达的标记，可以据此设计自动分析方法来检测文本中的重叠，当检测到时可首先预期正面或负面的情感表达，通过对重叠的细致分析，再进一步细致地判断情感的极性和强度。重叠对于文本情感分析的影响，一是，既能突出情感表达的语句, 也能起到调控情感强度的作用, 同时, 重叠还能弥补仅仅依靠情感词进行情感分析的一些不足。二是，重叠, 尤其是句式结构上的重叠, 以及当重叠不依赖情感词而表达情感时，对情感分析也是一个新的挑战。如何把重叠现象和情感词相结合以开发出更好的情感分析技术还是一个新方向, 本文对此做了一些探讨。在未来的研究中，我们计划对重叠现象做进一步的分析，归纳总结如何对重叠进行自动化的计算分析，并与情感词语料库结合来设计具有针对性的算法，并于实验相结合来验证算法的表现。

致谢

本文中娜仁图雅的工作为北京市人力资源与社会保障局外派项目"口述文本的关键语义信息提取(2018PC03)"的阶段性研究成果；同时得到国家社科基金重大委托项目"语言大数据挖掘与文化价值发现"(14&ZH0036)的资助。

参考文献

赵妍妍, 秦兵和刘挺. 文本情感分析. 软件学报, *21(8):1834-48*.

Pang B, Lee L, and Vaithyanathan S. Thumbs up? Sentiment classification using machine learning techniques. *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing, 10:79-86*.

Borbosa L and Feng J. 2010. Robust sentiment detection on Twitter from biased and noisy data. *Proceedings of the 23rd International Conference on Computational Linguistics, Association for Computational Linguistics, 36-44*.

Kamps J, Marx M, Mokken RJ, and De Rijke M. 2004. Using WordNet to measure semantic orientations of adjectives. *In LREC, 4:1115-1118*.

张月梅和刘媛华. 2020 基于K近邻和随机森林的情感分类研究. 计算机与数字工程, *48(2):367-371*.

刘勇和兴艳云. 2019 基于改进随机森林算法的文本分类研究与应用. 计算机系统应用, *28(5):220-225*.

梁柯, 李健, 陈颖雪和刘志钢. 基于朴素贝叶斯的文本情感分类及实现. 智能计算机与应用, *5(34)*.

任勉和甘刚. 2018. 基于双向LSTM模型的文本情感分类. 计算机工程与设计, *39(7):2064-2068*.

卢强, 朱振方, 徐富永和国强强. 2019 融合语法规则的Bi-LSTM中文情感分类方法研究. 数据分析与知识发现, 3(11):99-107.

王伟, 孙玉霞, 齐庆杰和孟祥福. 2019. 基于BiGRU-Attention神经网络的文本情感分类模型. 计算机应用研究, *12:8*.

卢玲, 杨武, 王远伦, 雷子鉴和李莹. 2018. 结合注意力机制的长文本分类方法. 计算机应用, *38(5):1272-1277*.

周咏梅, 杨佳能和阳爱民. 2013 面向文本情感分析的中文情感词典构建方法. 山东大学学报*(*工学版*)*, *43(6):27-33*.

Shen Y, Li S, Zheng L, Ren X, and Cheng X. 2009. Emotion mining research on micro-blog. *In 2009 1st IEEE Symposium on Web Society, 71-75*.

王素格, 杨安娜和李德玉. 2009. 基于汉语情感词表的句子情感倾向分类研究. 计算机工程与应用, *45(24):153-161*.

黄高峰和周学广. 2015. 一种语句级细粒度情感倾向性分析算法研究. 计算机应用与软件, *32(4):239-42*.

张仰森, 郑佳, 黄改娟和蒋玉茹. 基于双重注意力模型的微博情感分析方法. 清华大学学报*(*自然科学版*)*, *58(2):120-130*.

杨立公, 朱俭和汤世平. 2013 文本情感分析综述. 计算机应用, *33(06):1574-607*.

索绪尔. 1980. 普通语言学教程. 北京:商务印书馆.

李宇明. 1996. 论词语重叠的意义. 世界汉语教学, *1(11):10-19*.

董秀芳. 2016. 主观性表达在汉语中的凸显性及其表现特征. 语言科学, *15(6):561-70*.

基于BiLSTM-CRF的社会突发事件研判方法

胡慧君[1,2]，王聪[1,2]，代建华[3]，刘茂福[1,2] ✉
1. 武汉科技大学计算机科学与技术学院，武汉，430065
2. 智能信息处理与实时工业系统湖北省重点实验室，武汉，430065
3. 智能计算与语言信息处理湖南省重点实验室，湖南师范大学，长沙，410081
liumaofu@wust.edu.cn

摘要

社会突发事件的分类和等级研判作为应急处置中的一环，其重要性不言而喻。然而，目前研究多数采用人工或规则的方法识别证据进行研判，由于社会突发事件的构成的复杂性和语言描述的灵活性，这对于研判证据识别有很大局限性。本文参考“事件抽取”思想，事件类型和研判证据作为事件中元素，以BiLSTM-CRF方法细粒度的识别，并将二者结合，分类结果作为等级研判的输入，识别出研判证据。最终将识别结果结合注意力机制进行等级研判，通过对研判证据的精准识别从而来增强等级研判的准确性。实验表明，相比人工或规则识别研判证据，本文提出的方法有着更好的鲁棒性，社会突发事件研判时也达到了较好的效果。

关键词： 事件分类；研判证据识别；等级研判；BiLSTM-CRF

Social Emergency Event Judgement based on BiLSTM-CRF

Huijun Hu[1,2], Cong Wang[1,2], Jianhua Dai[3], Maofu Liu[1,2] ✉
1. School of Computer Science and Technology,
Wuhan University of Science and Technology, Wuhan, 430065
2. Hubei Province Key Laboratory of Intelligent Information Processing and
Real-time Industrial System, Wuhan, 430065
3. Hunan Provincial Key Laboratory of Intelligent Computing and
Language Information Processing, Hunan Normal University, Changsha, 410081
liumaofu@wust.edu.cn

Abstract

In recent years, classification and rating of social emergency event have attracted more and more attentions in emergency management. However, most of the current studies adopt the rule-based methods to identify the evidences for event judgement, and have troubles in event judgement due to the complexity of social emergency event composition and the flexibility of language description. Inspired by the idea of event extraction, this paper has proposed the event judgement method via BiLSTM (Bidirectional Long-Short Term Memory) and CRF (Conditional Radom Fields) based on event classification and evidence extraction. The social emergency event classification is carried out firstly, and then the event evidences are extracted based on event type. In the end, the rating of social emergency event is judged with the attention mechanism and the combination of event type and evidence. Experimental results show that the proposed method is more robust than rule-based ones, and effective in the social emergency event judgement.

Keywords: Emergency Event Classification , Evidence Identification , Emergency Grade Judgement , BiLSTM-CRF

1 引言

近年来，社会突发事件频发，给人们带来了巨大影响，也给应急处置带来了巨大挑战。社会突发事件突然发生的特性和所具有的破坏性，要求应急处理必须做到时效性和准确性，从而来及时止损。而社会突发事件类型和等级研判作为应急处置的起始部分，决定着后续应急预案能否快速准确地实施。目前，国内对社会突发事件的应急处置已初具规模，但如果让应急决策者手工去处理应急处置中的所有部分，可能会缺乏效率，延误最佳的应急时机。因而，社会突发事件类型和等级的自动研判在应急处置中非常关键。

国内较早便开展了社会突发事件应急处置的相关工作，由于当时标准不明，研究多是探索性的。2006年，国家在总体应急预案中发布了有关突发事件类型和等级的标准后，相关研究便逐渐多了起来。然而，此标准更多注重的是对突发事件类型全面覆盖，对等级划分标准并未作出详细说明，导致目前大多数研究更多注重的是对应急处置的综合性评价，强调了应急处置的风险性，Ivica等 (2019)通过模糊决策的方式来确定雷电的位置对输电网的影响，Sedova等 (2018)使用模糊推理方法来对海上突发事件进行等级研判，Sun (2018)采用了直觉模糊集理论，并结合层次分析法从而实现对水利工程施工应急救援方案的定量评价。近些年来，基于机器学习方法也逐渐开始展露，Hou等 (2013)将聚类方法运用在突发事件应急物资的分类上，Qiu等 (2019)和Fu等 (2016)结合贝叶斯模型在突发事件的应急处置中进行运用，商丽媛等 (2014)融合支持向量机的方法实现对突发事件的等级研判，徐绪堪等 (2018)在商丽媛等 (2014)的基础上，提出了基于随机森林的突发事件等级研判方法。虽说现有研究有一定的效果，但还是存在一些不足，目前对于应急处置中等级研判证据大多采用人工或规则方法。人工提取研判证据，则需大量人力，且进行研判时也需人工识别研判证据，灵活性较差；采用规则方法，规则库的建立需要专家大量的时间进行总结与归纳，并且中文语言灵活多变，还会存在规则库建立不完全的问题，如例1和例2所示。

例1：2020年1月9日，武汉市出现**首例新冠肺炎病亡患者**，经核酸检测方法共检测出**新型冠状病毒阳性结果15例**。

例2：截至2020年1月15日，武汉市累计报告**新型冠状病毒感染的肺炎病例41例**，在治重症5例，**死亡2例**。

例1中和例2中，“首例新冠肺炎病亡患者”和“死亡2例”同时指向了死亡人数，而“新型冠状病毒阳性结果15例”和“感染的肺炎病例41例”同时指向了感染人数，对“死亡”和“感染”人数方面的研判证据采用了不同的描述形式，规则方法将无法保证规则集覆盖所有的研判证据描述。

上述例子中可以看出，现研究方法在研判证据提取上还存在着许多不足，这些不足直接影响了后续应急处置的精准性。本文提出的基于BiLSTM-CRF的序列标注研判方法，实现了对研判证据更加细粒度化的识别，并结合注意力机制达到对突发事件精准研判的目的，而目前的社会突发事件研判研究重点并不在此。相比较于以往研究，本文有两个优点，一是序列标注模型识别出的研判证据更加的灵活和准确，没有人工的繁琐，比规则的精确。二是以往研究中大多数只考虑了应急处置中的某一个环节，而忽略了不同环节之间的影响性，本文将分类和等级研判结合，并在进行等级研判时融合注意力机制分配不同证据之间的权重，优化社会突发事件研判任务效果。

2 基于BiLSTM-CRF的突发事件研判方法

社会突发事件研判任务是指针对某一条突发事件文本判定其具体突发事件类型和突发事件等级。受Mu等 (2019)和He等 (2018)的启发，参考“事件抽取”的思想，将社会突发事件研判任务划分为事件识别、事件分类、研判证据识别和等级研判四个子任务，通过对整个研判任务的细致划分，获取更加精准的事件语义信息，达到增强研判效果的目的。

事件识别通过识别事件触发词，判断文本中是否含有事件，来达到识别事件的目的，如：“婚庆现场发生了爆炸”，识别到了“爆炸”为事件的触发词，即可知道文本中含有突发事件，再通过对触发词进行分类，从而确定文本突发事件的类型，由此便可知事件发生了，发生

基金项目：国家社科基金重大研究计划（11&ZD189）；全军共用信息系统装备预先研究项目（31502030502）；湖南省科技计划项目(2018TP1018, 2018RS3065)

事件是什么类型。对于突发事件等级研判，其关键是研判证据的识别，但中文语言的不规格化和灵活性导致现有研判证据识别准确率低，突发事件等级研判的效果差。为了提高突发事件等级研判证据识别的准确率，本文在研判证据识别中结合事件分类任务，融入突发事件类别信息，突发事件类型不仅会加强研判证据的识别能力，并且会对突发事件等级研判有很大影响，如“台风登陆时中心附近最大风力有13级（38米/秒）”，通过识别到“台风”，可知发生了突发事件，事件类型为“气象灾害”，当知道突发事件类型为“气象灾害”时，人为便可以联想到“台风”、“暴雨”、“大雾”等具体气象灾害事件，便可知推测到“风力”、“降雨量”、“能见度”等事件元素为研判证据，由此便可看出事件类型对研判证据识别的影响较为明显。在进行等级研判时，为了避免其它因素的干扰，只将事件类型信息和研判证据作为等级研判的判别特征，整个方法框架图如图1所示。

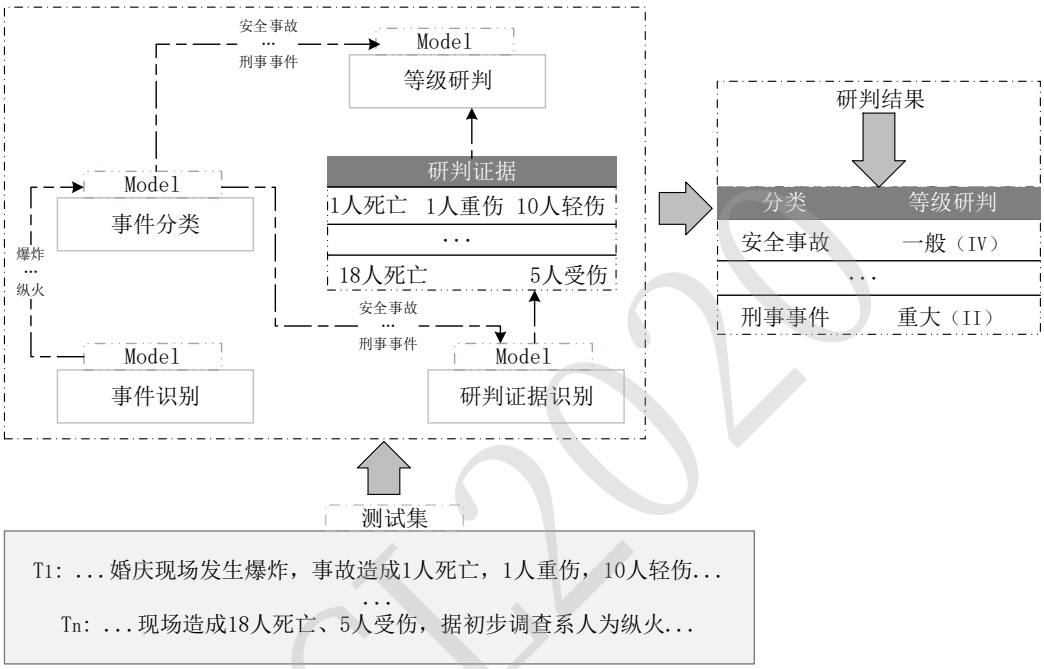


Figure 1: 方法框架图

图1中可以看到，方法流程为先进行事件识别和事件分类，再进行研判证据识别，最后完成等级研判。研判证据的识别在输入层结合了事件分类的结果，再将识别出来的研判证据和分类结果进行等级研判。对研判证据的精准识别，可在进行等级研判时减少其他因素的影响，让等级研判只跟识别出的研判证据和类别信息相关，由此研判的等级会更加的精确。

2.1 事件分类与研判证据识别

事件分类采用了标注模型实现，包括触发词的识别和分类，将识别出的触发词进行分类，即可确定突发事件的类别，与分类模型相比，标注模型对识别文本语义相似的不同类别事件有着更好的效果。事件分类的实现方法基于BiLSTM-CRF模型，BiLSTM-CRF的模型可以同时完成触发词识别和突发事件分类，减少错误传播，从而最终分类结果较好。首先文本经过字符嵌入后，输入到BiLSTM模型中，再通过CRF方法获取序列标注结果，方法过程由上述图1可以看出，测试文本经过模型识别出“爆炸”、“纵火”等为激活事件元素，再而进行事件分类，分成“安全事故”、“刑事事件”等类别，从而得到了事件类型信息。

本文提出的方法中，突发事件类型会作为等级研判环节的一部分，主要影响表现为：一是作为研判证据之一成为等级研判的输入，与其它研判证据不同的是，突发事件类型含有突发事件的类别信息，可作为单独任务提供信息，也可结合其它任务，当作为单独任务则是事件分类任务，识别事件进行分类，而不为单独任务时，则可成为等级研判的一环，从而来提升等级研判的效果。其次便是类别信息对研判证据的影响，如上述所说，当发生了“气象灾害”事件，可以推断出“风力”、“降雨量”等可能为其研判证据，由此便可看出类别信息很大程度上能帮助研

判证据的识别。现有的突发事件的等级研判方法，由于研判证据的非标准化，传统方法很难精准识别出研判证据，而BiLSTM-CRF的序列标注模型则可以细粒度地识别这些研判证据信息，再者考虑到类别信息对研判证据识别的影响性，因此本文提出了基于类型信息融合的研判证据识别方法，此方法以BiLSTM-CRF模型为基础，通过结合类别信息和原始文本，将得到的突发事件分类结果，融合到研判证据的识别中，从而提高研判证据识别准确度，研判证据识别模型的框架图由图2所示。

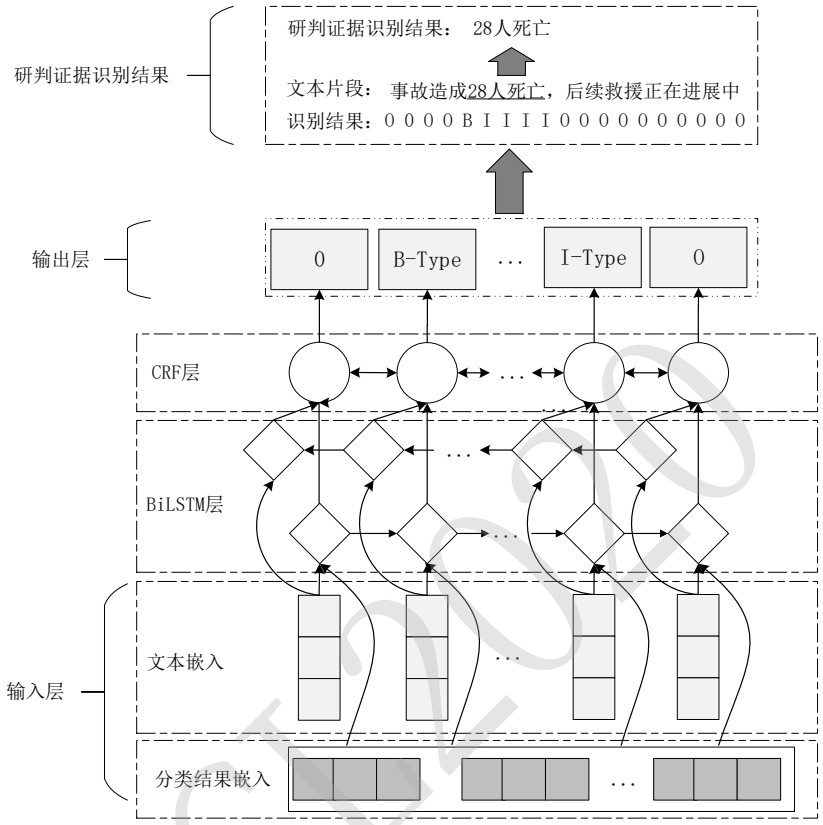


Figure 2: 研判证据识别模型

图2研判证据识别模型中，输入层中由下往上分别表示突发事件分类输出结果和文本向量表示，将其类别信息与当前输入结合经过BiLSTM层进行处理，再通过CRF层得到预测的序列标注结果，则可得到研判证据识别结果。

输入层：类别信息向量矩阵和输入文本向量矩阵的结合语义向量表示。采用了预训练模型BERT（Bidirectional Encoder Representation from Transformers）来获取文本的语义向量表示，BERT发布以来，在自然语言处理研究多项任务都取得了优异的成绩。BERT采用了双向Transformer 作为特征抽取器，可以获取更丰富的语义信息，文本经过BERT模型将进行字符级嵌入转换成向量形式。类别信息和输入文本经过BERT获取到各自的语义向量表示，结合作为输入层。

BiLSTM层：由BERT传过来的向量矩阵，经过BiLSTM层从而来得到更多的语义信息。LSTM（long short-term memory）为长短时记忆神经网络，是RNN的一个变种，用于解决RNN在时间序列中长期依赖丢失的问题。LSTM的输入为向量矩阵，经过下列步骤则可以得到隐藏层的向量表示。

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{1}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$g_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{3}$$

$$C_t = i_t * g_t + f_t * C_{t-1} \tag{4}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t * tanh(C_t) \tag{6}$$

其中σ代表着$Sigmoid$函数，f、i、o、C分别代表着输入门、遗忘门、输出门和最后的Cell。双向长短时记忆神经网络（BiLSTM）则是将同一序列分别经过前向LSTM和后向LSTM，从而得到隐藏层输出h_t 和h'_t ，将h_t 和h'_t 结合从而得到BiLSTM的输出h 。

CRF层：BiLSTM传过来的概率矩阵通过条件随机场（CRF）的方法来获取序列的最优标记。在以往研究中，已表明CRF在序列标注求解问题上有很好的效果。传入的BiLSTM的输出的概率矩阵为O_{m*l}，其中O_{ij}表示第i个字符映射到第j个标签上的概率。当已知序列$seq = \{w_1, w_2, \ldots, w_n\}$的预测的标签序列为$y = \{y_1, y_2, \ldots, y_n\}$，则从式（7）则可以得到当前序列的得分。

$$f(x, y) = \sum_{i=0}^{l} (A_{y_i, y_{i+1}} + O_{i, y_i}) \tag{7}$$

其中，A 为转移概率矩阵，该矩阵在对当前位置进行标注时可以利用之前标注信息，$A_{y_i, y_{i+1}}$表示标签y_i移到标签y_{i+1}时的概率。通过求解$f(x, y)$的最大值来获取最优的标签序列，再采用动态规划算法来得到最优标注路径。

输出层：已标注的序列文本。

通过输出层输出的已标注的序列，则可以识别出需要的研判证据，在图3中可以看出，通过模型输出后，则得到了文本中研判证据。通过研判证据才能决定突发事件的等级。

2.2 等级研判

等级研判为社会突发事件研判任务中的最后一个环节，该环节融合了分类结果和证据识别结果，当得到突发事件分类结果和研判证据后，则可进行等级研判。突发事件的等级研判时将只受类型和研判证据的影响，从而避免其他不相关因素对等级研判的影响。识别出来的研判证据结果和突发事件分类输出结果进行结合，作为等级研判任务的输入，经过BiLSTM层，再结合注意力机制，最终预测出当前文本的突发事件等级。模型的网络结构图如图3所示。

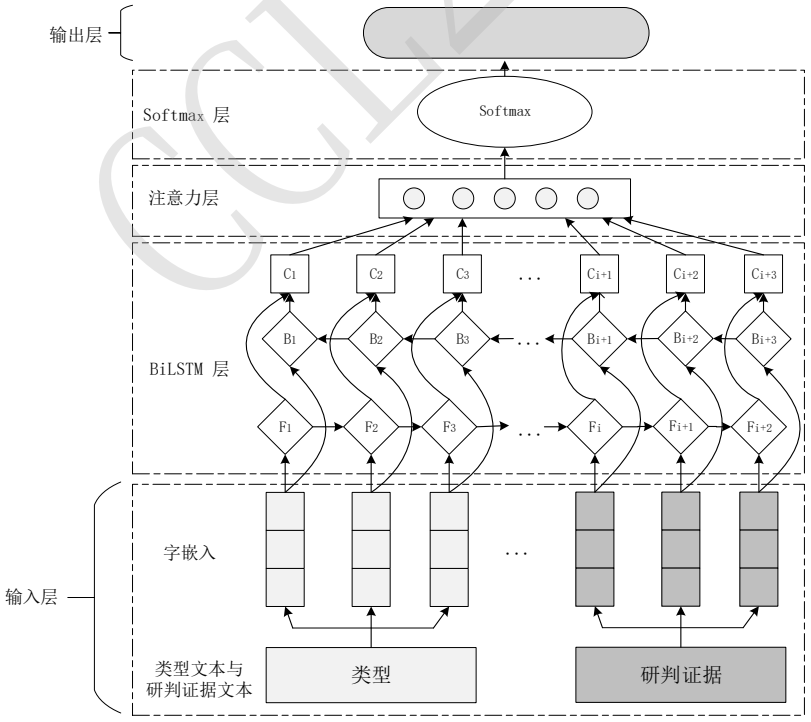


Figure 3: 等级研判模型

图3中的“类型”表示事件分类的输出结果。“研判证据”则表示研判证据的识别结果，将这两部分同时输入，进行编码，转换成向量矩阵，输入到BiLSTM层处理信息，再经过注意力层，

通过注意力机制来调节各个证据之间的权重大小。近些年，注意力机制在在各个自然语言处理任务中已取得了不错的成绩，由于等级研判时不同证据对研判的影响不同，加入注意力机制可以有效的处理这个问题，从而来提高等级研判的准确性。由BiLSTM层输出的特征向量矩阵H，经过以下步骤得到注意力层的输出γ。

$$M = tanh(H) \tag{8}$$

$$\alpha = Softmax(W^T M) \tag{9}$$

$$\beta = H\alpha^T \tag{10}$$

$$\gamma = tanh(\beta) \tag{11}$$

得到注意力层的输出后，最后经过$Softmax$层分类输出研判等级。

3 实验

3.1 数据集

社会突发事件的研判语料来源于政府网站信息公开中的突发事件通报及微博中突发事件新闻报道，标注过程中参照《国家特别重大、重大突发公共事件分级标准（试行）》、《贵州省突发事件分级标准》文件，确立标注规范，标注了突发事件的事件触发词、事件类型、研判证据和突发事件等级四部分，其中事件类型包括了：安全事故、公共卫生事件、地震灾害、气象灾害及刑事事件五大类，突发事件等级包括了：特别重大（I）、重大（II）、较大（III）和一般（IV）四个等级，按照此规范，共标注突发事件数据2000条，形成突发事件研判语料集，具体语料分布情况如表1所示。

Table 1: 语料分布表

刑事事件	公共卫生事件	气象灾害	安全事故	地震灾害	总计
450	325	281	519	425	2000
特别重大（I）	重大（II）	较大（III）	一般（IV）	–	
400	443	388	769	–	

社会突发事件研判语料按照BIO标注方式进行标注，标注出上述五个类型的事件触发词，即激活事件的事件元素和事件类型，同时标注出用于等级研判的研判证据和突发事件等级，如例4所示。

例4：2007年8月13日16时45分，湖南省湘西土家族苗族自治州凤凰县正在建设的堤溪沱江大桥发生特别重大**坍塌事故**，造成**64人死亡**、**4人重伤**、**18人轻伤**，直接**经济损失3974.7万元**。事故发生后，党中央、国务院领导同志作出重要批示，华建敏国务委员赶赴事故现场指导抢险救援工作。

在例4中，将“坍塌事故”标注为当前文本的事件触发词，并确定当前文本事件类型为“安全事故”。而“64人死亡”、“4人重伤”、“18人轻伤”和“经济损失3974.7万元”是突发事件等级研判任务中的主要因素，在此文本中，这些证据同时作为评判突发事件等级的关键特征，需都标注，并按照标注规范，确定当前文本突发事件等级为“特别重大（I）”。整个突发事件研判语料集都采用例4中标注方法进行标注，以交叉检验的方式进行评估，评估结果近似完全的正确度从而完成了整个数据集的构建。本文选取了已标注突发事件研判语料集中的1000份作为训练集，500份语料作为验证集，剩下500份语料为测试集。

3.2 实验设置

社会突发事件的研判研究，本文将其划分为四个子任务，在事件识别任务中，对于结果的判定，要求其预测的触发词与预先标注好的触发词匹配，才能判定正确，事件分类任务中，目的是得到整个的文本的突发事件类型，因此对于识别出的事件触发词，要求其类型正确则为正确分类，实验参数设置如下：输入的维度max_seq_length为256，训练集的batch_size为16，测试集的batch_size为8，训练学习率为1×10^{-5}，使用dropout来防止过拟合，值为0.5。对于突发事

件研判证据识别，同样要求预测出的研判证据与预先标注好的研判证据一致则判定识别正确，而对于突发事件等级研判，要求预测出的等级和预先标注等级一致。研判证据识别任务中，使用了事件分类任务相同的实验参数设置，对于等级研判任务，实验参数设置如下：输入的维度max_seq_length为126，训练集与测试集batch_size为256，训练学习率为1×10^{-3}，dropout值为0.5。

评价标准采用了精准度（P）、召回率（R）和F-score（F值）来评估事件识别和分类、研判证据识别和等级研判的效果，如公式（12）-公式（14）。

$$P = \frac{\text{模型正确识别的总数}}{\text{模型识别的总数}} \tag{12}$$

$$R = \frac{\text{模型正确识别的总数}}{\text{语料中标准结果的总数}} \tag{13}$$

$$F\text{值} = \frac{2PR}{P+R} \tag{14}$$

3.3 实验的结果与分析

按照上述方法和模型，将数据集划分为训练集、验证集和测试集，用训练集和验证集进行训练与验证，将训练好的模型在测试集上进行预测，根据预测结果与真实结果之间的差异进行评估，评估结果如表2所示。

Table 2: 基于序列标注模型研判结果表

模型	P（%）	R（%）	F值（%）
事件识别	87.82	88.03	87.93
事件分类	97.53	95.00	96.22
研判证据识别(+类型)	77.30	84.62	80.79
研判证据识别	72.97	83.90	78.05
等级研判	76.64	76.40	76.48

表2中可以看到对于事件识别和事件分类之间的F值相差了8.29%，造成如此大的差距是由于相近触发词导致，本文对于突发事件类型分类在于整个文本所属类别，当出现了触发词，但识别的位置和词语与预先标注好的不一致，但预测的突发事件类型一样时，也将判定为正确，如例5所示：

例5：9月13日上午，海口市东湖南里发生一起**故意伤害**致死案：一男子持刀**行凶**，造成3名男子1死2伤。

例5中，标注的事件触发词为“行凶”，事件类型为“刑事事件”，在进行预测时并未识别出“行凶”，而将“故意伤害”作为事件的触发词识别出来，类型同样也为“刑事事件”。在进行突发事件类型判定时，其对于文本而言预测出来的类型正确则为正确。

在表2中的研判证据识别(+类型)和研判证据识别分别表示在加入分类结果和不加分类结果的研判证据识别效果，从表中可以看出，再融入了分类结果后，研判证据抽取F值有明显的提升。

突发等级研判与事件分类不一样，从表2中可以看出研判证据识别的F值比等级研判的F值高，研判证据正确识别决定等级研判的准确性，若缺少研判证据，则大部分情况都会造成判定的等级过低，如例6所示：

例6：8月19日15时05分，一辆安阳市区至安阳县北郭乡的公交车上发生持刀抢劫杀人案。车上33名乘客，15人被捅伤，其中2人在救治途中死亡，1名伤者经抢救无效死亡。

例6中，通过文中模型可以识别出当前类型为“刑事事件”，但对于其他等级研判特征的识别并未识别完全，文中模型识别出了“15人被捅伤”，而未识别出死亡研判证据，因此导致了最终的研判等级变低。

社会突发事件的研判任务的最终目的在于突发事件分类和等级研判，表2中可以看到以BiLSTM-CRF的序列标注模型完成的突发事件分类和等级研判的效果，为了证明序列标注模型的有效性，本文将徐绪堪等 (2018)所使用随机森林模型的方法作为对比，同时也加入了BERT模型作为对比，如表3所示:

Table 3: 实验对比表

模型	事件分类			等级研判		
	P（%）	R（%）	F值（%）	P（%）	R（%）	F值（%）
随机森林模型	80.85	60.80	63.76	60.30	46.40	46.27
BERT模型	94.52	94.00	94.41	68.07	69.00	67.75
序列标注模型	97.53	95.00	96.22	76.64	76.40	76.48

由表3可以看出，序列标注模型的实验结果表现最好，当使用徐绪堪等 (2018)的方法时无论是事件分类或等级研判，其效果与BERT模型以及序列标注模型都相差较多，造成这种情况的可能原因便是随机森林模型在徐绪堪等 (2018)的实验中所使用的语料集规模较少，突发事件类别种类单一，当扩大语料集和突发事件类别种类时便达不到较好的效果。序列标注模型在事件分类中相比较于随机森林模型和BERT模型F值分别提高了30.46%和1.81%，在这个实验中，序列标注模型能更加有效的识别文中的类别信息，即事件的触发词，相比较于其他两个模型而言，序列标注模型能有效避免在噪点信息的干扰。等级研判中序列标注模型与随机森林模型和BERT模型相比F值分别提高了30.21%和8.73%，随机森林模型除上述提到的数据规模扩大，效果不佳的情况外，其次缺陷便是由于语言的灵活性，采用规则的方法并不能很好的识别研判证据，从而导致等级研判结果较差，而BERT模型采用了句子语义信息来作为研判证据，其证据不如采用序列模型识别出的证据精确，且二者模型都未充分使用类别信息。

例7：2008年7月21日，云南省昆明市发生了公交车连环爆炸案事件，造成2名乘客死亡和4名乘客受伤。云南省急救中心及时到达现场，对此事件导致的伤员进行了迅速、高效的紧急医疗救援，最大程度地减少人员伤亡，控制和减轻突发事件的危害，维护了社会稳定。

对于例7，除随机森林模型将其事件类型分类成“刑事事件”，其他两个模型都能正确识别事件类型为“安全事故”，造成错误分类的可能原因有数据集中如“云南”、“爆炸”等词高频率的出现在刑事事件类别中，因此随机森林模型对于该数据并不能很好的识别，而BERT模型和序列标注模型能学习到文本中更多的语义信息，因此能识别正确。对于此例子的等级研判结果，除序列标注模型识别正确以外，其余两个均识别错误。首先，对于随机森林模型，基于规则的研判证据的识别方法灵活性较差，当伤亡信息中出现了其他的词或字，便不能很好的识别。对于BERT模型，由于文本中冗余信息过多，所以也并未研判正确。

4 总结与展望

本文中提出了基于序列标注模型来完成社会突发事件的研判方法，相比较以往研究，此方法能更加灵活准确的识别出研判证据，通过使用BiLSTM-CRF的方法提高识别的准确率，同时将事件分类和等级研判结合起来，来优化等级研判的效果。

目前本文中只考虑到了单一突发事件的情况，但有时一个社会突发事件的发生可能产生其他衍生突发事件，此时若要对当前突发事件进行等级研判则需考虑到其衍生突发事件对其的影响。接下来的工作将会加入衍生突发事件，若当前突发事件产生了其他衍生突发事件，则判定其对当前突发事件联系性和影响性，再而进行突发事件的等级研判。

参考文献

Deliang Jiang. 2010. *Research on extraction of emergency event information based on rules matching.* Computer Engineering and Design. 31(14):3294-3297.

Lingxia Hou, and Kunying Li. 2013. *Classification of Emergency Supplies on Fuzzy Clustering.* logistics engineering and management. (3):74-75.

Luheng He, Lee Kenton, Omer Levy, and Luke Zettlemoyer. 2018. *Jointly Predicting Predicates and Arguments in Neural Semantic Role Labeling.* ACL. pages:364–369. doi:10.18653/v1/P18-2058.

李素建, 王厚峰, 俞士汶, 辛乘胜. 2004. 关键词自动标引的最大熵模型应用研究. 计算机学报. 27(9):1192-1197.

韩永峰, 郭志刚, 陈翰, 许旭阳. 2012. 基于领域特征词的突发事件层次分类方法. 信息工程大学学报. 13(5):593-600.

Ivica Petrović, Srete Nikolovski, Hamid Reza Baghaee, and Hrvoje Glavas. 2019. *Determining Impact of Lightning Strike Location on Failures in Transmission Network Elements using Fuzzy Decision Making.* IEEE Systems Journal. 14(2):2665-2675. doi: 10.1109/JSYST.2019.2923690

Jiangnan Qiu, Yanzhang Wang, Leilei Dong, and Xin Ye. 2011. *A Model for Predicting Emergency Event Based on Bayesian Networks.* journal of systems & management. 20(1):98-103.

贾泓昊, 罗智勇. 2019. 基于序列标注的引语识别初探. 中文信息学报. 33(2): 1-7.

Kaichang Sun, Wenjun Ma, and Quan Li. 2018. *Decision-making of IAHP-intuitionistic fuzzy set-based emergency rescue scheme.* water resources and hydropower engineering. 49(6):135-140.

Yingwei Luo, Xiaolin Wang, and Xinpeng Liu. 2008. *A Rule-based Event Handling Model.* IEEE Asia-Pacific Services Computing Conference. pages:869-875. doi: 10.1109/APSCC.2008.48.

Matthew Sims, Jong Ho Park, and David Bamman. 2019. *Literary Event Detection.* ACL.pages:3623–3634. doi:10.18653/v1/P19-1353.

Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. 2019. *Exploring Pre-trained Language Models for Event Extraction and Generation.* ACL. pages:5284–5294. doi:10.18653/v1/P19-1522.

Sedova A.Nelly, Viktor A.Sedov, and Ruslan Bazhenov. 2019. *Analysis of Emergency level at Sea Using Fuzzy Logic Approaches.* International Conference of Artificial Intelligence. doi: 10.1007/978-3-319-67349-3_30.

Shuhui Gao, and Xiao Jia. 2019. *Application of Convolutional Neural Network in Forensic Evidence Examination and Prospect of Hair Evidence Identification.* science technology and engineering. 19(23):1-9.

沈兰奔, 武志昊, 纪宇泽, 林友芳, 万怀宇. 2019. 结合注意力机制与双向*LSTM*的中文事件检测方法. 中文信息学报. 33(9): 79-87.

商丽媛, 谭清美. 2014. 基于支持向量机的突发事件分级研究. 管理工程学报. 028(001):119-123.

Xiaofeng Mu, and Aiping Xu. 2019. *A Character-Level BiLSTM-CRF Model With Multi-Representations for Chinese Event Detection.* IEEE Access 7:146524-146532.

徐绪堪, 王京. 2018. 基于随机森林的突发事件分级模型研究. 中国安全生产科学技术. 014(002):77-81.

应文豪, 李素建, 穗志方. 2017. 一种话题敏感的抽取式多文档摘要方法. 中文信息学报. 31(6):155-161.

杨健, 黄瑞章, 丁志远, 陈艳平, 秦永彬. 2020. 基于边界识别与组合的裁判文书证据抽取方法研究. 中文信息学报. 34(3): 80-87.

Zhe Fu, Hao Zhang, and Qiang Chen. 2016. *Application of Naive Bayes Classifier in Stampede Risk Early-Warning of Large-Scale Activities.* Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII). IEEE. pages:174-180. doi: 10.1109/ICIICII.2016.0051.

结合金融领域情感词典和注意力机制的细粒度情感分析

祝清麟[1,2]，梁斌[1]，刘宇瀚[1]，陈奕[1]，徐睿峰[1,2,†]，毛瑞彬[3]
1. 哈尔滨工业大学(深圳) 计算机科学与技术学院，广东 深圳 518055
2. 哈工大理光联合实验室 广东 深圳 518055
3. 深圳证券信息有限公司 广东 深圳 518028
†. 通讯作者，Email：xuruifeng@hit.edu.cn

摘要

针对在金融领域实体级情感分析任务中，往往缺乏足够的标注语料，以及通用的情感分析模型难以有效处理金融文本等问题。本文构建一个百万级别的金融领域实体情感分析语料库，并标注五千余个金融领域情感词作为金融领域情感词典。同时，基于该金融领域数据集，提出一种结合金融领域情感词典和注意力机制的金融文本细粒度情感分析模型。该模型使用两个LSTM网络分别提取词级别的语义信息和基于情感词典分类后的词类级别信息，能有效获取金融领域词语的特征信息。此外，为了让文本中金融领域情感词获得更多关注，提出一种基于金融领域情感词典的注意力机制来为不同实体获取重要的情感信息。最终在构建的金融领域实体级语料库上进行实验，取得了比对比模型更好的效果。

关键词： 细粒度情感分析 ； 金融文本 ； 情感词典

Attention-based Recurrent Network Combined with Financial Lexicon for Aspect-level Sentiment Classification

Qinglin Zhu[1,2], Bin Liang[1], Yuhan Liu[1], Yi Chen[1], Ruifeng Xu[1,2,†], Ruibin Mao[3]
1. Department of Computer Science, Harbin Institute of Technology (Shenzhen), Shenzhen, Guangdong 518055, China;
2. HIT-RICON Joint Lab, Shenzhen, Guangdong 518055, China;
3. Shenzhen Securities Information Co. Ltd., Shenzhen, Guangdong 518028 China;
†. Corresponding author, Email: xuruifeng@hit.edu.cn

Abstract

To address the lack of sufficient annotated corpus and the poor performance of common sentiment analysis models for the task of entity-level sentiment analysis of financial texts. This paper builds a multi-million level corpus of sentiment analysis of financial domain entities and labels more than five thousand financial domain sentiment words as financial domain sentiment dictionary. Based on this financial domain dataset, we propose an Attention-based Recurrent Network Combined with Financial Lexicon, called FinLexNet. FinLexNet model uses LSTM to extract category-level information based on financial domain sentiment dictionary and another LSTM to extract semantic information at the word-level, which can effectively obtain information about the characteristics of financial domain words. In addition, in order to get more attention to the financial sentiment words, an attention mechanism based on the financial domain sentiment dictionary is proposed. Finally, experiments were conducted on the dataset we constructed, which shows that our model has achieved better performance than the comparative models.

Keywords: Fine-grained sentiment analysis , Financial texts , Financial Sentiment Lexicon

1 引言

随着互联网和金融行业的快速发展，在金融领域目前不断出现大量专业的股评报告、研究报告等信息，以及个体投资者的个人看法和分析。无论新闻报道还是针对相关主题与公司的评论信息，往往都包含有对相关事件与公司的评价与态度，具有丰富的投资和监管参考价值。对这些评价信息的全面把握，有助于投资者更好的了解市场，辅助投资决策。同时，对于金融市场监管者，有助于及早从评价中发现潜藏的问题，对于掌握市场动态，消除市场风险也有着重要意义。为此，金融文本的情感分析研究正在成为当前研究和应用热点。

金融领域的文本实体级细粒度情感分析研究尚处于初级阶段，也是细粒度情感分析重要的子任务(Pang and Lee, 2008)，面临着诸多挑战。首先缺乏高质量、大规模的金融领域文本情感标注语料，导致文本处理底层技术缺乏数据支撑。现有的通用文本情感分析模型缺乏对金融领域文本特点的分析和利用，没有考虑金融领域词性特征，缺乏对金融领域情感先验知识的利用，因此在金融文本上表现欠佳。

针对现有金融领域语料库匮乏的问题，本文构建了金融领域细粒度情感分析语料库。首先对各大金融新闻网站进行数据爬取与清洗，之后按照字级别对所爬取的数据进行实体标注和实体情感标注。总共标注了5206篇新闻稿，整理出3325个实体和对应的9240条情感语句，并构建了包含5047个词的金融领域情感词典。

针对现有方法对金融领域知识利用的不足，本文设计并实现了结合金融领域情感词典和注意力机制的细粒度情感分析模型（Attention-based Recurrent Network Combined with Financial Lexicon, FinLexNet），该模型使用一个LSTM提取词级别的文本信息，并基于金融情感词典将文本中的词分成“积极”，“消极”，“中立”，“金融实体”，“其他”五个类别对文章进行表示，使用另一个LSTM提取金融领域词性特征，这样不仅让模型关注到不同类型词语的特殊性，从而更好的理解上下文的语义信息，还能作为对词级别较为细粒度的信息的补充，获取更宏观的文本信息。模型还使用了金融领域情感词典指导注意力机制，使得注意力机制更加关注金融领域情感词，在构建的数据集上达到了同类模型的最佳效果。

本文构建了一个百万级的金融领域实体级细粒度情感分析语料库，并在此基础上提出了一种结合金融情感词典和注意力机制的情感分析模型。不仅促进围绕金融领域文本的情感分析研究的深入，具有很好的科学意义，同时，可以很好服务于面向金融领域的舆情分析、市场判断和监管协调，具有突出的应用价值。

2 相关研究

细粒度情感分析是情感分析的一个热门且具有重要应用价值的领域(赵妍妍 et al., 2010)，侧重于对细粒度情感信息的挖掘。对于金融领域，实体级的细粒度情感分析是分析出金融文本中出现的金融实体的情感，常用的方法有基于情感词典，机器学习和深度学习的细粒度情感分析方法。

情感词典是识别文本情感的有效工具，有不少学者研究构建情感词典的方法(Meng et al., 2012; 梅莉莉 et al., 2016)，形成了如Word Net(Fellbaum, 2012)等具有代表性的英文情感词典和董振东等人编制的中文知网情感词典How Net。基于情感词典的细粒度情感分析方法主要是利用句式词库和情感词典去分析文本语句的特殊结构及情感倾向词，如Wu等(2006)根据情感词情感强度的不同而赋予不同的情感权重，然后进行加权求和。Lipenkova等(2015)提出了预建立的词典和通用语言规则相结合的方式，其在中文方面级情感分析任务上取得了较好的效果。

基于机器学习进行细粒度情感分析也是主流的方法之一，在早期的研究中，细粒度情感分析被当作一般情感分类任务，使用情感词典、文本语义等特征等提取文本特征来建立细粒度情感分类模型。Kiritchenko等(2014)引入了产品的总体评分和情感词库两个外部知识，并和SVM分类器相结合一起，在SemEval2014年竞赛中取得了最佳性能。Ramesh等(2015)提出使用马尔科夫随机场解决在线课程MOOC中的方面级情感分类问题。郝志峰等(2015)提出把情感对象识别看作一个序列标记问题，通过在传统的CRF序列标记模型上增加情感对象的全局节点，有效地结合上下文信息、句法依赖以及情感词典，从而可以识别出微博中的情感对象。然而传统的机器学习方法通常需要依赖大量的人工筛选特征，这需要耗费大量的时间和精力。

随着深度学习技术的发展，研究人员设计了一系列的神经网络自动生成对象和内容的低维度表示方法，并且在细粒度情感分类任务中得到了较好结果。Tang等(2016a)提出一种基于目标的长短期记忆网络（TD-LSTM），依据目标词的位置将输入的文本切分成左右两个部分并分别送入LSTM，较传统LSTM模型性能有所提升。注意力机制（Attention Mechanism）源于对人类视觉的研究，近年来，随着注意力机制的深入研究，很多学者基于注意力机制提出了一系列的方法进行细粒度情感分析。赵冬梅等(2018)提出一种利用协同过滤算法计算得到用户的情感分布矩阵，再使用注意力机制提取文本信息，从而进行实现情感分类。曾峰等(2019)提出了一种基于注意力机制的LSTM神经网络模型，模型从词级别和句子级别两个层面进行语义提取，从而获取不同词语和句子的重要性。吴小华等(2019)使用字向量对文本进行字级别的表示，并使用双向的LSTM网络和注意力机制提取上下文之间的关系。

金融领域细粒度情感分析研究较少，Cortis等(2017)讨论了SemEval-2017会议“金融微博和新闻的情感分析”任务三十余位参赛者的方法和工具，其中最多人使用的是基于传统机器学习模型SVM和SVR的方法。Do等(2019)指出金融领域数据标注需要广泛的领域专业知识，进行专业标注会很昂贵，所以构建的数据较少。Maia等(2018)发布了一个非常小的数据集（FiQA），包含了金融领域的文本实例和文本中提到的实体，并给每个实体的情感打分。Yang等(2018)基于ELMo模型提出了ULMFiT方法分析FiQA数据集上的金融实体情感。Salunkhel等(2019)提出了一种用于方面分类的迁移学习方法和一种用于金融数据的情感预测的回归方法，迁移学习方法利用了BERT，并使用了不同的回归方法，其中线性支持向量回归法的效果最好。

细粒度情感分析的方法较多，但是在金融领域实体级细粒度的情感分析研究较少，尤其是缺乏数据集的情况下使得金融领域的研究更难以开展。我们针对语料库匮乏的问题构建了金融领域细粒度情感分析语料库。针对现有模型缺乏对金融领域知识利用的问题，构建了提出了结合金融领域情感词典的细粒度情感分析方法，用金融领域情感词指导注意力机制，并结合金融领域词性特征，取得了同类模型的最佳性能。

3 金融领域实体级细粒度情感分析语料库构建

针对金融领域情感分析语料库匮乏的问题，我们设计并构建了金融领域实体级细粒度情感分析语料库。考虑到新闻文本信息丰富，更新速度快且较为正规，我们爬取了各大金融数据网站（21世纪经济报道 [0]、财新网 [1]、每经网-公司板块 [2]、生意社 [3]、人民网 [4]）作为数据来源，采用Scrapy框架共计爬取22681篇新闻文本，并对文章进行了删除特殊符号，利用正则匹配剔除一些无关信息等预处理。

首先我们进行了金融实体的标注。对于金融实体，我们标识出文本中的公司名，人名和品牌名称。实体名基于长匹配的原则进行标注，并通过天眼查辅助确定公司名、品牌名称等。

例如：“**乐融致新**和**乐视网**业务发展的颓势仍没有出现明显的好转。”

在这个文本中“乐融致新”和“乐视网”为我们标注的实体。

对于金融实体情感标注，我们将金融实体的情感极性标注为三大类：无情感、消极、积极，每一类指定的标注准则如下：

(1)积极情感

对于积极情感的标注，如果文本中出现了有利于公司经营的事实，以及一些人为的积极评价，则标注为积极。

例如：“伴随着近年来白酒行业复苏，**水井坊**业绩也水涨船高。”

(2)中立情感

对于中立情感的标注，如果文本中出现的信息为与公司经营相关，但无法判断是有利还是不利的事情标注为中立，包括一下情况：

①一些与公司经营相关的事实性的陈述，包括（但不限于）：公司人事变更、子公司或者下属经营企业的设立与关闭、公司财务或投资操作等等。

②既有有利事实也有不利事实（句中不存在转折词例如尽管、然而、虽然、但是等表达情感偏向的副词）。

[0]http://news.21so.com/chanye/
[1]http://companies.caixin.com/news/
[2]http://www.nbd.com.cn/columns/346
[3]http://news.toocle.com/list/c-3511—1.html
[4]http://industry.people.com.cn/GB/413887/index.html

③一些与公司经营相关的中性人为表述与评价。

例如：“电商是未来发展的方向，所有的企业都在发力，**华为**也不例外，但目前来看，这一动作的成效需要检验。”

(3)消极情感

对于消极情感的标注，若文本的信息不利于公司经营，标注为消极。包括一些不利于公司经营的事实，以及一些人为的消极评价。

例如：“由于游戏收入下滑，热门游戏进入周期末尾，近期市场对**腾讯**的评估本来就不太乐观。”

为了构建金融领域细粒度情感分析数据集，从爬取的22681篇新闻文本中选取了5206篇进行标注。首先由4名标注人员进行预标注2000条，在标注过程中分别对各自的标注结果进行比对收集差异与有歧义的地方，制定对各类实体以及针对模糊和有冲突的语境制定相应的标注准则。在标注过程中，每一段新闻文本由至少两名标注者独立标注，即标注过程中标注者之间彼此没有交流，完全依赖先前制定好的标注准则。独立标注完成后，对于有差异或有错误的标注结果，一名额外的标注者会参与讨论，直到所有的标注者意见统一后，对以标注数据进行人为修改，最终完成标注。

最终整理出3325个金融实体，每个金融实体对应一个或多个语句，共计有9240条对应的情感语句，共108.7万字。在9240个情感语句中，金融实体情感是积极的有4189条，中性的有3202条，消极的有1627条。具体的统计结果如表1所示。

极性	积极	中性	消极
数量	4179	3202	1627
百分比	46.39%	35.55%	18.06%

表 1. 金融实体情感数据统计

通过分析金融文本数据，根据经验判断出了哪些词汇会影响对与实体情感极性的判断，从而构建了一个金融领域情感词典，其中包含了2079个积极词，1070个中立词和1898个消极词。金融领域情感词典的具体统计信息如表2所示。

极性	积极	中性	消极
数量	2079	1070	1898
百分比	41.19%	21.20%	37.61%

表 2. 金融领域情感词统计

为了计算带标注的语料库与标注者之间的一致性，计算了Cohen’s Kapp(Cohen, 1960)值与Fleiss’ Kappa(Fleiss, 1971)值。Fleiss’ Kappa值为0.6686，实验标注结果数据具有较好一致性。Cohen’s Kapp值达到0.7210，这说明标注者可以在给定文本的情况下可靠地识别目标实体的情感。

4 结合金融领域情感词典和注意力的细粒度情感分析模型

本文提出的一种结合金融领域情感词典和注意力的情感分析模型框架图如图1所示。为提取细粒度的语义信息，使用LSTM提取词级别的语义信息（见模型右半部分）；为了让模型关注到不同类型词语的特殊性，并获取更宏观的文本信息作为对词级别信息的补充，使用另一个LSTM提取词类级别的语义信息（见模型左半部分）。其中词类级别的表示是指将文章词分成5个类别：Pos，Neg，Neu，Entity，Other，即“积极”，“消极”，“中立”，“金融实体”，“其他”五个类别。然后使用Word2Vec模型对文本进行训练，从而获取每一类词语词向量的平均值来表示该类词向量。为了更关注到与预测情感极性相关度高的词语，使用金融实体与金融文本进行词级别的注意力。为了让模型更加关注金融领域情感词，模型还使用了金融领域情感词典去指导注意力机制，从而使模型更加关注金融情感词所在的位置，提升情感分析的准确度。

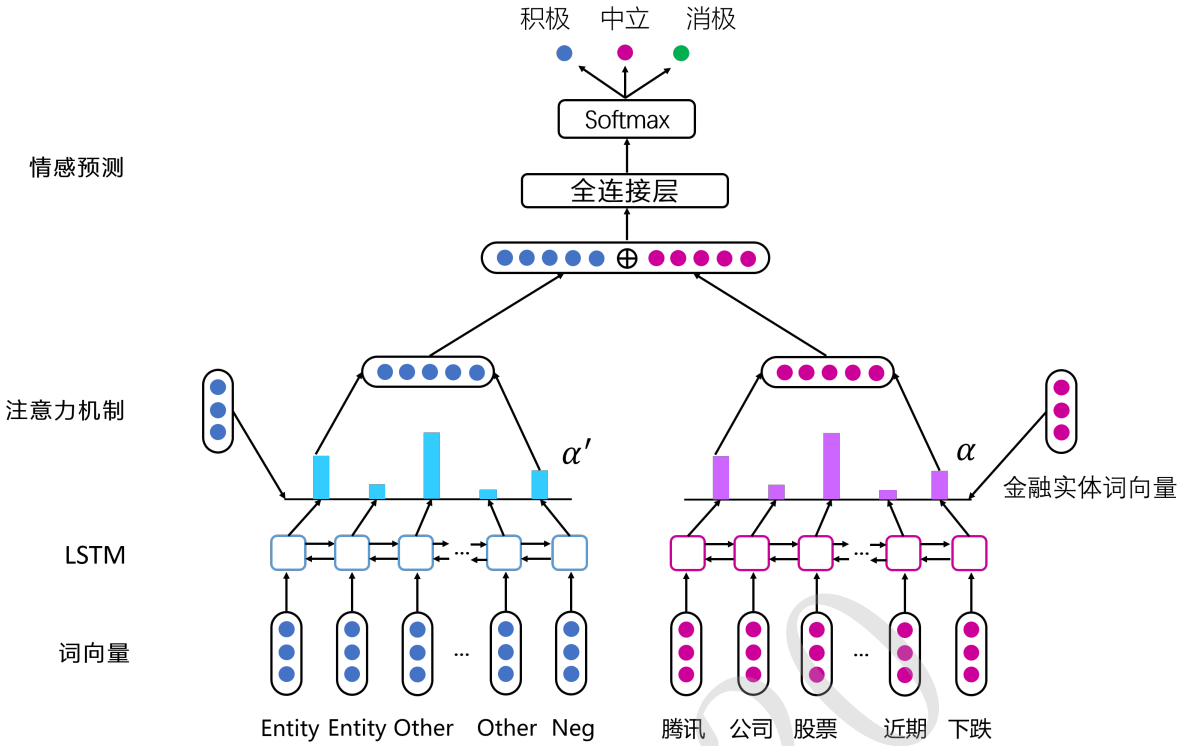


图 1. 结合金融领域情感词典和注意力的情感分析模型

4.1 基于LSTM的词级别编码器

为了提取文本的语义，使得其特征表示更加符合当前语境信息，我们采用了长短记忆网络（Long-Short Term Memory，LSTM）提取词级别的语义特征，将整条金融文本先经过分词和词嵌入之后输入LSTM，如公式 (1) 所示。

$$\overrightarrow{h_t^l} = \overrightarrow{LSTM_w}\left(x_{it}\right), t \in [1, k] \tag{1}$$

其中，一共有i个金融文本，每个金融文本d中包含k个词语：$\{w_{i1}, w_{i2}, \ldots, w_{ik}\}$。$x_{it}$为$t$时刻。通过LSTM就可以获得每个词语的向量表示为$[h_1^i, h_2^i, ..., h_c^i]$。

4.2 基于LSTM的词类级别编码器

上一节中提取的文本的是词级别较为细粒度的信息，但不能注意到不同类别的词汇。在本节中使用词类级别的表示方法，本文基于领域情感词典和已经标注好的金融实体，将文章词分成5个类别：Pos，Neg，Neu，Entity，Other，即“积极”，“消极”，“中立”，“金融实体”，“其他”五个类别。其中“积极”，“消极”，“中立”来自标注的情感词典，“金融实体”为标注的金融实体，“其他”为其他词汇或是未登录词。然后使用Word2Vec模型对文本进行训练，从而获取每一类词语词向量的平均值来表示该类词向量。通过词类级别的表示不仅可以给模型提供不同词类级别的信息，让模型关注到不同类型词语的特殊性，从而更好的理解上下文的语义信息，而且还能获取更宏观的文本信息，作为对词级别较为细粒度的信息的补充。

为了具体说明如何进行词类级别的表示，在此举一个简单的例子：经过分词后“腾讯 公司 股票 近期 下跌”这句话中有5个词语，其中“腾讯”和“公司”两个词语为金融实体，“股票”和“近期”属于其他词汇，“下跌”为消极类金融领域情感词。则这句话的词类级别表示为$[Entity\ Entity\ Other\ Other\ Neg]$，每一类对应着相同的词向量。

LSTM的隐状态输出序列$[h_1, h_2, \ldots, h_t]$可以作为当前文本的特征表示，其中的h_t对应于文本序列中第t个词的特征。

为了提取词类级别的语义特征，本文使用另一个LSTM网络作为文本的特征提取器，将之前介绍的经过词嵌入表示的词类级别文本输入$LSTM'$，如公式(2)。

$$\overrightarrow{h_t^{l'}} = \overrightarrow{LSTM_w'}\left(x_{it}\right), t \in [1,k] \tag{2}$$

4.3 词级别的注意力机制

通过两个LSTM建模得到的文本表示，会给每一个词分配相同的权重，因而无法准确把握语义的重点。注意力机制的思想是不同的情境下不用文本的重要程度不同，在计算过程中将文本的语义根据分配的权重进行加权求和，获得与任务更相关的文本的表示。为了进一步提升情感分析的准确度，借助注意力机制，建模实体情感与各个词语之间的关系，为子句的词序列语义特征分配不同的权重，使得更重要的词语得到更多的关注。由公式(3)与公式(4)为词语的注意力权重计算方式:

$$\alpha_{it} = \frac{\exp\left(\gamma\left(h_c^i, e^E\right)\right)}{\sum_{j'}\exp\left(\gamma\left(h_c^j, e^E\right)\right)} \tag{3}$$

$$\gamma\left(h_c^j, e^E\right) = \tanh\left(h_c^i \cdot w_m^T \cdot e^{E^T} + b_a\right) \tag{4}$$

其中经过LSTM后每个词语的向量表示为$[h_1^i, h_2^i, ..., h_c^i]$,$w_m$是权值矩阵，$b_a$为偏移量，$e^E$为金融实体的词向量（如果金融实体被切分为多个词则$e$为这些词向量取均值后的结果）；$\alpha_{it}$为词$w_{it}$相对于金融实体$e^E$的注意力权重。

LSTM提取的带有注意力加权的子句文本特征表示如公式(5)所示:

$$o_i = \sum_t \alpha_{it} h_{it} \tag{5}$$

将词类级别的送入$LSTM'$后，同样使用注意力机制确定与实体相关的上下文语义信息。相似的，得到注意力得分为α_{it}'，得到的特征向量为o_i'。

4.4 基于金融领域情感词典的注意力指导

注意力机制能够更好的关注到重要的词汇从而提高模型识别的准确率，但不一定能够准确的识别哪些词语是对结果有较大影响的金融领域情感词。为了解决这个问题，我们使用构建的金融领域情感词典去指导注意力机制，使得金融情感词的获得更大的关注。

为了使用构建的金融领域情感词典，对于一个输入的分词后的句子，构建了一个与分词后的句子长度相同的情感词向量，称为Vec_{Lex}，并初始化为0。遍历输入金融文本中的词语若其出现在金融领域情感词典中，则在情感词向量中将对应位置设为1。 为了更方便的理解金融领域

图 2. 金融领域情感词向量示意图

情感词向量的概念，举一个简单的例子例如图 2 所示，假设输入的金融文本为“腾讯 公司 股票 近期 下跌”，首先初始化一个情感词向量$[0,0,0,0,0]$，遍历输入的句子发现“下跌”这个词出现在金融领域情感词典中，属于消极词，便把“下跌”这个词在情感词向量对应的位置设置为1，则该句话的情感词向量为$[0,0,0,0,1]$。

为了使得注意力机制更加的关注金融领域情感词，我们修改了损失函数，在交叉熵损失后又加入了一项$\lambda\left(\alpha - Vec_{Lex}\right)^2$。其中$\lambda$是确定情感词典损失重要性的超参数，$\alpha$为注意力机制的得分，$Vec_{Lex}$为情感词典向量。从而使得注意力机制得分$\alpha$去拟合金融情感词向量，从而使模型更加关注输入金融文本金融情感词。

之后将词类级别的注意力表示和词级别的注意力表示相结合，将两个带有注意力加权句子文本体征表示向量拼接起来，最后经由$softmax$层得到模型的概率输出，如公式(6)所示:

$$p_i' = softmax\left(o_i \oplus o_i'\right) \tag{6}$$

其中$\oplus$为向量拼接操作，o_i经过LSTM的注意力机制的词级别表示，o_i'是经过$LSTM'$的注意力机制的词类级别表示。

模型的最终的损失函数为公式(7)：

$$L = -\sum_{i \in D} y_i \log p_i + \lambda\left(\alpha_{norm} - Vec_{Lex}\right) \tag{7}$$

其中，D为样本集合，y_i为子句真实标签，p_i为模型的预测结果，λ是确定情感词典损失重要性的超参数，α_{norm}为LSTM词级别注意力得分α和经过$LSTM'$的词类级别注意力得分α'的平均值。

5 实验

5.1 数据集

实验数据集采用构建的金融领域实体级细粒度情感分析语料库，将数据集分成测试集，验证集与训练集，具体的划分如表3所示。

划分	积极	中性	消极	合计
训练集	2771	2157	1072	6000(64.94%)
验证集	564	461	215	1240(13.42%)
测试集	907	725	368	2000(21.64%)

表 3. 金融领域情感文本训练集测试集数据统计

5.2 评价指标

本文使用准确率(Accuracy)和Macro-F1值作为评价标准。

5.3 实验设计

(1)词向量

实验的词向量采用腾讯AI Lab公开的中文词向量数据(Song et al., 2018)，该数据集涵盖面广，囊括了800余万个中文词语，数据集的维度为200维。该词向量的训练使用了腾讯研制的Directional Skip-Gram (DSG)算法，相比于广泛采用的词向量训练算法Skip-Gram (SG)，DSG算法额外考虑了词对的位置信息，以从而能更准确的表示词汇的语义。具有词语覆盖率全，新鲜度高，词向量准确率高的特点。在训练的过程中词向量不冻结，参数随训练一起更新。

(2)超参设置

数优化采用 Adam(Kingma and Ba, 2014)优化算法, 学习率设置为 0.0001。对词向量矩阵以及不同LSTM 层之间的连接采用Dropout(Srivastava et al.,)，对LSTM 层内部与隐状态相关的权重矩阵采用 DropConnect(Cho et al., 2014)。Batch-size设置为128，Dropout为0.2，DropConnect为0.1，LSTM的隐藏层为200维，LSTM Attention的输出为200维，$LSTM'$ Attention的输出为50维，情感词典损失重要性的超参数$\lambda = 0.035$。

(3)对比模型介绍

对比模型包括基础的Bi-LSTM模型和近些年在方面级的情感分析（Aspect Based Sentiment Analysis）领域的深度学习模型进行对比，参与对比的模型有以下几种：

- Bi-LSTM：Bi-LSTM是Bi-directional Long Short-Term Memory的缩写，是由前向LSTM与后向LSTM组合而成，使用Bi-LSTM模型提取文本的语义信息之后，直接送入softmax层进行分类。

- TD-LSTM(Tang et al., 2016a)：基于目标的长短期记忆网络，根据特定目标单词的所在位置，将训练语句拆分左右两部分，通过LSTM获取左右部分两个隐层的输出，输入分类器，获取分类结果。

- IAN(Ma et al., 2017)：该模型改进了传统的分类模型中将两者分开独立建模或者只针对内容建模的方法，该模型先让内容和目标分别通过不同的LSTM后，利用注意力机制实现两者的信息交互，从而提升模型的准确度。

- AOA(Huang et al., 2018)：该模型建模了目标和文本的交互关系，分别将文本和目标经过双向的LSTM，并使用隐藏层的输出接着计算两者的交互矩阵，将该矩阵得到的信息送入softmax实现对情感的分类。

- MemNet(Tang et al., 2016b)：该模型利用了注意力机制的QA系统中的深度记忆网络，将方面词的上下文信息作为存储器中存储的内容，实现了一个针对方面级的情感分析模型。

- ATAE-LSTM(Wang et al., 2016)：该模型利用了注意力机制来获取上文下信息与目标词信息之间的关系，结合了LSTM神经网络与注意力机制提取句子语义，从而提升情感分类的准确度。

5.4 实验结果与分析

(1)总体性能

所有实验均采用 NVIDIA GeForce GTX 2080Ti 显卡进行计算加速，并在单张显卡下完成。在自标注的数据集进行了实验，总体性能的实验结果如表4。

模型	Precision	Macro-F1
Bi-LSTM	0.7040	0.6860
TD-LSTM	0.7132	0.6847
IAN	0.7115	0.6853
AOA	0.7285	0.6971
MemNet	0.7225	0.6954
ATAE-LSTM	0.7165	0.6913
FinLexNet(Ours)	**0.7425**	**0.7147**

表 4. 总体实验性能结果图

从实验结果可以看出，我们提出的模型FinLexNet取得了74.25%的准确度和0.7147的Macro-F1值，均达到了对比模型的最佳效果。基础模型Bi-LSTM的效果最不理性，是因为只能获取总体的文本信息，并不能对实体进行建模。TD-LSTM提取实体前后语句语义的综合，性能有所提升。IAN和AOA实现了实体与模型之间的交互，更好的理解到了实体在文中的语义信息，同Bi-LSTM相比也有不小提升。ATAE-LSTM使用注意力机制对实体和文本进行建模，但是我们认为注意力机制没有准确把握关键词导致性能没有明显提升。我们认为提出的FinLexNet模型性能较好的原因是结合了金融领域词性信息并用金融情感词指导注意力机制，能让模型获得的信息更加丰富并使得注意力更好的把握关键词。

(2)消融实验

为了考察模型框架中各组件的贡献程度，本文设置了模型中不同结构的消融实验。

- LSTM-ATT：使用LSTM去提取文本信息，并使用注意力机制。

- LSTM-ATT-Lex：使用LSTM提取文本信息，并使用标注的情感词典指导注意力机制。

- Double-LSTM-ATT：使用两个LSTM分别提取文本和词类表示的文本信息，并使用注意力机制后输入到$softmax$层，不使用标注的情感词典。

- Double-LSTM-ATT-Lex：使用两个LSTM分别提取文本和词类表示的文本信息并结合注意力机制，并使用标注的情感词典指导注意力机制。

实验结果如表5所示，从实验结果来看，使用金融领域情感词典指导注意力机制对实验结果具有较大的提升，说明金融领域情感词典中的词对判断实体的情感极性有较大的帮助，而通过

模型	Precision	Macro-F1
LSTM-ATT	0.7255	0.6914
LSTM-ATT-Lex	0.7295	0.6903
Double-LSTM-ATT	0.7303	0.6964
Double-LSTM-ATT-Lex	0.7425	0.7147

表 5. 消融实验结果

修改损失函数可以有效的指导注意力机制着重关注金融领域情感词，从而达到提升实验效果的目的。

(3)注意力可视化

为了探究模型注意力机制关注的内容，对三个输入样例的注意力权重进行了可视化，颜色表示一个词在给定句子中的重要性，颜色越深越重要。如图3所示。

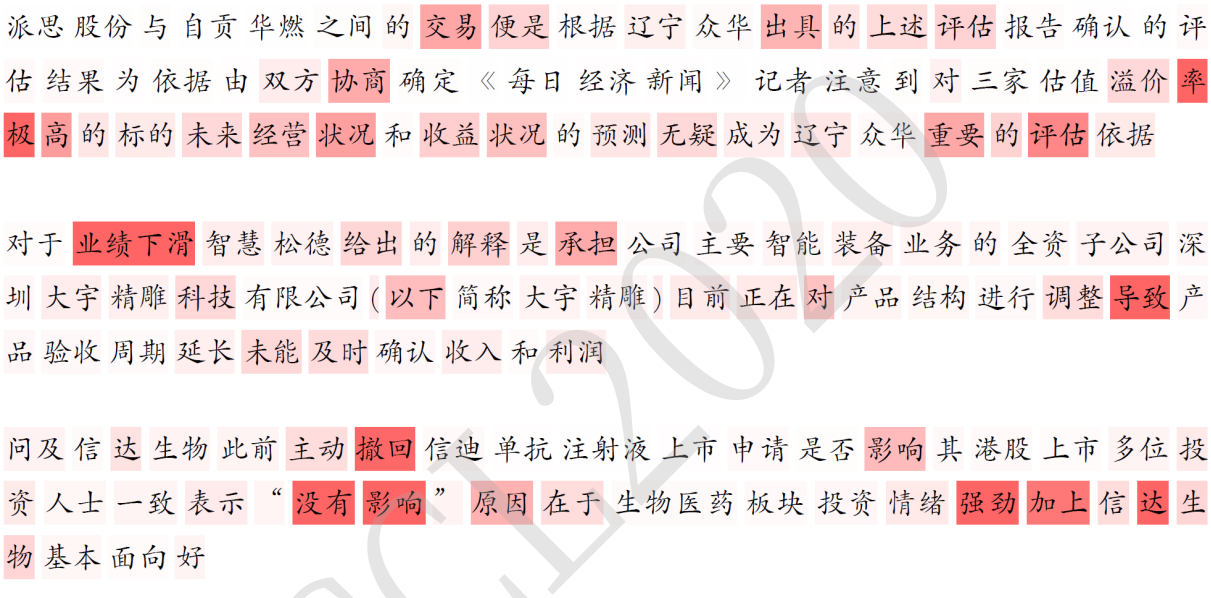


图 3. 注意力机制可视化图

如在第一段话中，金融实体为“派思股份”，情感极性为积极。从注意力可视化看出“溢价极率高”的颜色最深，对照了“派思股份”积极的情感极性。

在第二个金融文本中，金融实体为“智慧松德”，在文本中该公司对业绩下滑进行解释，情感极性为消极。从注意力可视化可以看出“业绩下滑”最能体现出情感极性，颜色最深。而连词“导致”往往用于不好的结果，也被模型准确的识别出来。

第三个金融文本中，金融实体为“信达生物”，文本说了撤回上市申请对公司的业务没有影响，所以情感极性为中立。在可视化结果中着重强调了“主动撤回”和“没有影响”，较为准确的找到了判断情感极性的关键词。

通过以上可视化的结果可以说明注意力机制较好的注意到了关键词和和金融领域情感词典，有助于模型判断金融实体的情感极性。

(4)错误分析

为了更好的改进模型，选取了一些错误案例进行分析。为了更方便的进行分析，将金融文本中的实体进行了加粗表示，消极的语句加上了下划线，积极的语句用波浪线标识。

例如：“市场认为，从财务数据来看，宣亚的收购是划算的。宣亚2016年的营业收入为4.67亿元，净利润为5871.01万元。8月15日，宣亚国际发布2017年中报，报告期内，公司实现营业收入2.10亿元，同比下降6.74；净利润为2722.00万元，同比增长4.22。而**映客直播**的同期营收达到了43.47亿元，归母净利润更是高达4.8亿元，远远高于上市公司。”

这个例子中，“映客直播”的情感极性为正面，而模型判断为负面。分析原因是模型可能没

有找准映客直播对应的语句，同“宣亚国际”的营收下降产生了混淆。

例如：“中信银行向**佳兆业**伸出援手始于佳兆业陷入债务危机之时。彼时中信银行深圳分行对危机中的佳兆业施以援手，提供大约300亿元资金助其解困，100亿元用于置换佳兆业位于上海、杭州等地的8个优质资产项目债务；另有100亿元将作为佳兆业的后续开发贷款。此后平安银行也与佳兆业达成全方位的战略合作，签约金额为500亿元，用于支持佳兆业的未来发展。”

该例子中，“佳兆业”的情感极性为正面，而模型判断为中立。分析原因是模型注意到了消极观点“陷入债务危机”，同时也注意到中信银行伸出援手，提供了300亿资金纾困，“支持佳兆业的未来发展”。从而认为是中立。而如果进一步的推理可以知道，这是一个利好的消息，所以情感极性为积极。这说明虽然模型学习到了很多情感词，但是由于缺乏对这种褒贬都存在的情况的进一步推理，导致分类错误。

6 总结与展望

本文构建了一个金融领域实体级细粒度情感分析语料库，并提出了一种结合金融领域情感词典和注意力的细粒度情感分析模型。为了利用金融领域词性信息并结合粗细粒度的文章信息，本文使用两个LSTM网络分别提取词类级别和词语级别的语义。为了让模型有针对性地关注对情感结果影响较大的词语，本文使用金融领域情感词典对注意力机制进行修正。最后，在本文标注的金融领域细粒度情感分析语料库上进行实验，实验结果表明本文提出的结合金融领域情感词典和注意力模型能有效提升细粒度情感分析的准确性。未来的工作可以针对文本中有正负两面评价金融实体的情感进行研究，并考虑如何充分利用文本中的金融数字信息。

致谢

国家自然科学基金 61876053, 61632011，深圳市技术攻关项目JSGG20170817140856618, 深圳市基础研究学科布局项目JCYJ20180507183527919, JCYJ20180507183608379, 广东省新冠肺炎疫情防控科研专项项目 2020KZDZX1224

参考文献

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46. Publisher: Sage Publications Sage CA: Thousand Oaks, CA.

Keith Cortis, André Freitas, Tobias Daudert, Manuela Huerlimann, Manel Zarrouk, Siegfried Handschuh, and Brian Davis. 2017. SemEval-2017 Task 5: Fine-Grained Sentiment Analysis on Financial Microblogs and News. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 519–535, Vancouver, Canada, August. Association for Computational Linguistics.

Hai Ha Do, PWC Prasad, Angelika Maag, and Abeer Alsadoon. 2019. Deep Learning for Aspect-Based Sentiment Analysis: A Comparative Review. *Expert Systems with Applications*, 118:272–299, March.

Christiane Fellbaum. 2012. WordNet. *The encyclopedia of applied linguistics*. Publisher: Wiley Online Library.

Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378. Publisher: American Psychological Association.

Binxuan Huang, Yanglan Ou, and Kathleen M. Carley. 2018. Aspect level sentiment classification with attention-over-attention neural networks. In *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*, pages 197–206. Springer.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. 2014. NRC-Canada-2014: Detecting Aspects and Sentiment in Customer Reviews. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 437–442, Dublin, Ireland, August. Association for Computational Linguistics.

Janna Lipenkova. 2015. A system for fine-grained aspect-based sentiment analysis of Chinese. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 55–60, Beijing, China, July. Association for Computational Linguistics and The Asian Federation of Natural Language Processing.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. *arXiv preprint arXiv:1709.00893*.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. WWW'18 Open Challenge: Financial Opinion Mining and Question Answering. In *Companion of the The Web Conference 2018 on The Web Conference 2018 - WWW '18*, pages 1941–1942, Lyon, France. ACM Press.

Xinfan Meng, Furu Wei, Ge Xu, Longkai Zhang, Xiaohua Liu, Ming Zhou, and Houfeng Wang. 2012. Lost in Translations? Building Sentiment Lexicons using Context Based Machine Translation. In *Proceedings of COLING 2012: Posters*, pages 829–838, Mumbai, India, December. The COLING 2012 Organizing Committee.

Bo Pang and Lillian Lee. 2008. Opinion Mining and Sentiment Analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135, July. Publisher: Now Publishers, Inc.

Arti Ramesh, Shachi H. Kumar, James Foulds, and Lise Getoor. 2015. Weakly Supervised Models of Aspect-Sentiment for Online Course Discussion Forums. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 74–83, Beijing, China, July. Association for Computational Linguistics.

Ashish Salunkhe and Shubham Mhaske. 2019. Aspect Based Sentiment Analysis on Financial Data using Transferred Learning Approach using Pre-Trained BERT and Regressor Model. 06(12):5.

Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. 2018. Directional Skip-Gram: Explicitly Distinguishing Left and Right Context for Word Embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 175–180, New Orleans, Louisiana, June. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. page 30.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective LSTMs for Target-Dependent Sentiment Classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3298–3307, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Duyu Tang, Bing Qin, and Ting Liu. 2016b. Aspect Level Sentiment Classification with Deep Memory Network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 214–224, Austin, Texas, November. Association for Computational Linguistics.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for Aspect-level Sentiment Classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615, Austin, Texas, November. Association for Computational Linguistics.

Chung-Hsien Wu, Ze-Jing Chuang, and Yu-Chung Lin. 2006. Emotion recognition from text using semantic labels and separable mixture models. *ACM Transactions on Asian Language Information Processing*, 5(2):165–183, June.

Steve Yang, Jason Rosenfeld, and Jacques Makutonin. 2018. Financial Aspect-Based Sentiment Analysis using Deep Representations. *arXiv:1808.07931 [cs]*, August. arXiv: 1808.07931.

吴小华, 陈莉, 魏甜甜, and 范婷婷. 2019. 基于Self-Attention和Bi-LSTM的中文短文本情感分析. 中文信息学报, 33(6):100–107.

曾锋, 曾碧卿, 韩旭丽, 张敏, and 商齐. 2019. 基于双层注意力循环神经网络的方面级情感分析. 中文信息学报, 33(6):108–115.

梅莉莉, 黄河燕, 周新宇, and 毛先领. 2016. 情感词典构建综述. 中文信息学报, 30(5):19–27.

赵冬梅, 李雅, 陶建华, and 顾明亮. 2018. 基于协同过滤Attention机制的情感分析模型. 中文信息学报, 32(8):128–134.

赵妍妍, 秦兵, and 刘挺. 2010. 文本情感分析. 软件学报, 21(8):1834–1848.

郝志峰, 杜慎芝, 蔡瑞初, and 温雯. 2015. 基于全局变量CRFs模型的微博情感对象识别方法. 中文信息学报, 29(4):50–58.

基于层次注意力机制和门机制的属性级别情感分析

冯超
云南广电网络集团有限公司
471573289@qq.com

黎海辉
深圳职业技术学院
lihaihuiscnu@m.scnu.edu.cn

赵洪雅(✉)
深圳职业技术学院
hy.zhao@szpt.edu.cn

薛云
华南师范大学
xueyun@scnu.edu.cn

唐婧尧
华南师范大学
manderous@foxmail.com

摘要

近年来，作为细粒度的属性级别情感分析在商业界和学术界受到越来越多的关注，其目的在于识别一个句子中多个属性词所对应的情感极性。目前，在解决属性级别情感分析问题的绝大多数工作都集中在注意力机制的设计上，以此突出上下文和属性词中不同词对于属性级别情感分析的贡献，同时使上下文和属性词之间相互关联。本文提出使用层次注意力机制和门机制处理属性级别情感分析任务，在得到属性词的隐藏状态之后，通过注意力机制得到属性词新的表示，然后利用属性词新的表示和注意力机制进一步得到上下文新的表示，层次注意力机制的设计使得上下文和属性词的表达更加准确；同时通过门机制选择对属性词而言上下文中有用的信息，以此丰富上下文的表达，在SemEval 2014 Task4和Twitter数据集上的实验结果表明本文提出模型的有效性。

关键词： 属性级别 ； 情感分析 ； 注意力机制 ； 门机制

Aspect-level Sentiment Analysis Based on Hierarchical Attention and Gate Networks

FENG Chao
Yunnan Radio Broadcasting &Television Networking Group CO., Ltd.
471573289@qq.com

LI Haihui
Shenzhen Polytechnic
lihaihuiscnu@m.scnu.edu.cn

ZHAO Hongya(✉)
Shenzhen Polytechnic
hy.zhao@szpt.edu.cn

XUE Yun
South China Normal University
xueyun@scnu.edu.cn

TANG Jingyao
South China Normal University
manderous@foxmail.com

Abstract

As a fine-grained task, aspect-level sentiment analysis whose purpose is to identify the sentiment polarity corresponding to the specific aspect in a sentence has received more and more attention in the business and academic. The most of the related works focus on the design of attention network to highlight the different contributions of words in context and make context and aspect interactived. In this paper, we put forward the hierarchical attention and gate networks to process aspect-level sentiment analysis task. we obtain the new representation of the context through the attention network between the context and the new representation of the aspect which weight by the context. At the same time, the useful information in the context is selected through the gate networks to enrich the representation of the context. The design of hierarchical attention gate networks make the representation of the context and the aspect more accurate.The experimental results on the SemEval 2014 Task4 and Twitter show the validity of the model.

Keywords: Aspect-level , sentiment analysis , attention networks , gate networks

1 引言

近年来，随着互联网的发展，大量的消费平台和社交网络平台逐渐走进人们的生活 (Li et al., 2019b)。在消费平台上，消费者消费之后往往会留下一段关于产品优劣的评论；在社交网络平台上，网民也会留下关于某个事件的看法。这些含有情感信息的评论无论是对企业或者是政府而言都是极具价值的，企业可以通过分析消费者的评论，了解产品的不足，以达到改善产品性能的目的；政府也可以通过总结网民对事件的看法以引导事件的发展方向。因此，如何从富含有情感信息的评论中挖掘出有用的信息成为自然语言处理的热点研究方向之一。

情感分析是自然语言处理中重要的子领域，可分为句子级别情感分析、篇章级别情感分析和属性级别情感分析 (Shaikh and Deshpande, 2016)。句子级别情感分析为对整个句子的单一情感极性（积极、中性和消极）进行判断。篇章一般由多个句子构成，所以篇章级别情感分析在于根据其构成的多个句子判断篇章的整体情感极性 (欧阳志凡, 2018)。由此可知，句子级别和篇章级别情感分析均为对评论文本的单一情感极性进行判断，而在产品以及产品属性日益多样化的今天，消费者一般会留下一段关于产品多个属性评论的评论文本，如：*"The food was extremely tasty, but the service was dreadful."*，这是一段来自酒店的评论，通过分析可以看出属性词*"food"*和"service"的情感极性分别是积极和消极，而得到整体情感极性的句子和篇章级别情感分析明显不适合这种含有多个属性词且对每个属性词表达情感不一致的情况，所以，属性级别情感分析逐渐受到学术界和商业界的关注。

属性级别情感分析需要找出特定属性词在上下文（本文将评论中除属性词部分称作为属性词的上下文）中所关联部分以达到判断特定属性词情感倾向的目的 (Li et al., 2019a)。因此，属性级别情感分析需要充分利用属性词和上下文的关系，即不同属性词在上下文中关注的词也有所不同 (Ma et al., 2017)，如：当识别属性词*"picture quality"*情感极性的时候，*"clear-cut"*自然能与属性词*"picture quality"*产生联系。除此之外，在属性级别情感分析中，不同词对于不同属性词情感极性判断的作用也是不一致的，如：情感词。因此，在近些年属性级别情感分析工作中，如何通过上下文和属性词的交互式学习，从而达到识别出上下文中对属性词重要词的目的已经成为了研究重点。常见的交互方式有如下几种：第一，将属性词向量与上下文向量进行拼接；第二，通过设计上下文和属性词之间的交互注意力机制。但是简单的拼接并不能达到识别出上下文中对属性词重要词的效果，因此，通过设计上下文和属性词之间的注意力机制成为了学术界研究热点方向之一。但是单层的注意力机制难以达到将上下文和属性词相互关联的目的。而且，在大多数属性级别情感分析工作中将上下文与属性词之间的注意力机制结果直接作为上下文和属性词的表达，本文认为该做法在一定程度上忽略了原上下文和属性词表示的作用。因此本文提出使用层次注意力机制和门机制（Aspect-level Sentiment Analysis Based on Hierarchical Attention and Gate Networks,HAG）来处理属性级别情感分析任务，贡献如下：

首先，本文认为单层交互注意力机制并不能很好的将上下文和属性词相互关联，而且难以识别出在上下文中对于属性词比较重要的词。如：在对*"The food is average and dreadful serving speed"*中的*"The food"*进行情感分析的时候，单层交互注意力机制会同时加大*"average"*和*"dreadful"*的权重，而*"dreadful"*的情感强度通常更大，*"dreadful"*的权重一般会大于*"average"*，而多层交互注意力机制将进一步对词语进行加权，调节*"average"*和*"dreadful"*的权重。本文使用层次注意力机制先获取上下文对属性词的影响矩阵，从而更新属性词的表示，进一步的得到上下文的表示。此方式利用上下文与属性词之间的注意力机制使属性词的表达更加准确，再将利用新的属性词表示与上下文的注意力机制，使的上下文的表达更加准确，通过层次注意力机制的设计将上下文和属性词之间进行了充分的关联，同时对不同词进行了加权以达到判断属性级别情感分类的目的。

其次，在以往属性级别情感分析的研究工作中，通常将属性词对上下文的影响矩阵当作是上下文的最终表示，将上下文对属性词的影响矩阵之间当作是属性词的表示,该做法忽略了上下文和属性词原表示的作用。本文认为影响矩阵只能作为对原表示调整的依据而不能直接作为上下文和属性词的直接表示，因此，本文将二者相加，一定程度上保留了原始信息。

最后，本文认为门机制的选择作用能够选择出对于不同属性词上下文中重要的词,因此本文通过门机制改变上下文的表达，丰富上下文的表示。实验结果表明门机制的选择作用在一定程度上提高了模型的准确率。

2 相关工作

从Minqing Hu (Hu and Liu, 2004)提出需要关注产品不同属性的情感表达起，便出现了大量属性级别情感分析工作，这些工作大致可以分为基于传统机器学习的属性级别情感分析方法、基于深度学习的情感分析方法、深度学习与注意力相结合的方法和其他混合的方法。本文提出的基于层次注意力机制和门机制的属性级别情感分析属于近几年成为研究热点的深度学习和注意力机制相结合的方法，本章将从注意力机制和门机制两部分具体介绍属性级别情感分析的相关工作。

2.1 注意力机制

近些年来，神经网络的发展加快了属性级别情感分析的研究，CNN、RNN、LSTM、GRU等经典的神经网络更是在提取文本隐藏特征有着显著的作用；另一方面，在自然语言处理中，注意力机制能够选择出有助于特定自然语言处理任务的词，在减小计算量同时，可以大大减少噪声的引入。因此，在近些年的自然语言处理任务中，将神经网络与注意力机制相结合成为了主流方法。在属性级别情感分析上，注意力机制的使用一定程度上可以选择出有助不同属性情感分类的词，从而提高了实验准确率，因此，深度学习与注意力机制相结合的方法也成为了属性级别情感分析的主要研究方法。Wang (Wang et al., 2016)在处理属性识别情感分析任务时提出了两个模型。第一个模型通过平均池化将属性词矩阵表示转化成属性词词向量，然后通过属性词词向量与上下文隐藏表示之间的注意力机制对上下文表示进行加权。相对于第一个模型，第二个模型在上下文进入LSTM之前先将属性词向量与上下文矩阵表示进行拼接。由于使用了注意力机制，两个模型都取得了较高的实验结果，后者通过拼接进一步加大了上下文和属性词之间的关联性，因此实验效果也有一定的提升。自此，在处理属性级别情感分析任务的大多数工作中都引入了注意力机制。Ma (Ma et al., 2017)将上下文和属性词之间的关联定义为交互并认为不仅属性词对上下文具有选择作用，而且上下文对属性词也有选择作用，因此，提出使用二者之间的交互注意力机制来处理属性级别情感分析任务。Huang (Huang et al., 2018)提出使用Attentio-over-Attention来增强上下文和属性词之间的交互程度。Zhang (Zhang et al., 2019)提出运用协同注意力机制处理属性级别情感分析任务，具体为：将评论句子分成属性词上文、属性词和属性词下文，然后将三部分两两进行协同注意力机制。Huang (Huang and Carley, 2019)通过语法感知和图注意力机制得到属性级别情感极性分布。以上工作都是通过单层注意力机制加深上下文和属性词之间的交互。Li (Li et al., 2019a)先通过协同注意机制在上下文和属性词的表示中分别融入彼此之间的影响，然后通过自注意机制进一步对自身表示进行加权。Li (Li et al., 2018)将上下文向量与位置向量进行拼接，经过GRU之后再与通过注意力机制得到的属性词向量进行拼接，最后再通过注意力机制得到含有属性词情感信息的最终表示。该工作通过多层注意机制使得上下文的表示能够更加准确的表达出属性词对应的情感信息。Song (Song et al., 2019)分别将上下文词嵌入与属性词词嵌入输入到多头注意力机制，分别得到上下文和属性词的表示，然后通过二者之间的注意力机制得到二者之间的交互向量，通过分类器进一步得到属性词对应情感极性分布。Zheng (Zheng and Xia, 2018)先将属性词隐藏表示与属性词上下文和属性词下文进行注意力机制得到属性词上文向量和属性词下文向量，然后将属性词上文向量和属性词下文向量分别与属性词隐藏表示进行注意力机制。该工作先通过注意力机制更新了属性词上文和属性词下文的表示，使得属性词上文和属性词下文的表示更加准确，然后进一步的获取属性词的表示，层次注意力机制的交互和互相选择作用使实验效果得到进一步的提升。

2.2 门机制

属性级别情感分析在于从上下文中选择出针对当前属性词情感分析有用的词从而改变上下文表示，以达到判断属性词情感极性分布的效果，而门机制在选择和提取信息有显著的效果。因此，也存在大量使用门机制处理属性级别情感分析任务。Xue (Xue and Li, 2018)先使用CNN提取属性词和上下文的信息，然后通过门机制（gating mechanisms）选择上下文和属性词中有用的信息。Liang (Liang et al., 2019)处理属性级别情感分析任务时设计了两种不同的门机制，使上下文的表示融入了属性词信息，同时提取和选择出了有用的上下文的信息，该方法在两个数据集上也取得了较好的实验效果。因此，本文使用了门机制选择出上下文中有用的信息，以此丰富了上下文的表达，实验结果证明门机制的设计在一定程度上丰富了上下文的表

达，对实验准确率的提高有一定的作用，进一步提高了模型的有效性。

3 模型

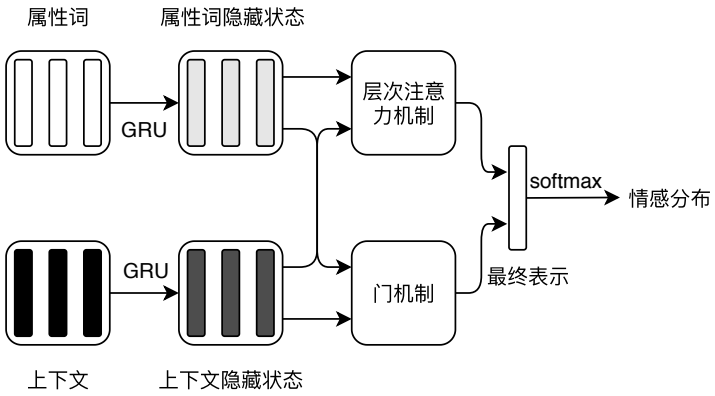


图 1: 总体模型结构

在属性级别情感分析中，常常给定一段对属性进行评论的评论文本，因此需要依次对评论文本中的属性进行情感极性的判断。本文将需要判断情感极性的属性称作为属性词，而评论中除属性词之外的词称作为上下文。给定一段长度为$n+m$的评论文本，其中上下文$[w_c^1, w_c^2, \cdots, w_c^n]$、属性词$[w_t^1, w_t^2, \cdots, w_t^m]$，根据GloVe得到对应上下文和属性词的词嵌入矩阵分别是$C = [v_c^1, v_c^2, \cdots, v_c^n] \in \mathbf{R}^{n \times d_w}$和$T = [v_t^1, v_t^2, \cdots, v_t^m] \in \mathbf{R}^{m \times d_w}$，其中$d_w$是词向量维度的大小。为了进一步提取出文本的隐藏特征，本文使用了GRU提取出上下文和属性词隐藏信息。GRU不仅可以缓解传统RNN在反向传播时出现的梯度消失和梯度爆炸的问题，而且在参数量上较少，因此GRU更适合短文本的属性级别情感分析任务。在通过GRU提取出上下文隐藏状态$H_C = [h_c^1, h_c^2, \cdots, h_c^n] \in \mathbf{R}^{n \times d_h}$和属性词隐藏状态$H_T = [h_c^1, h_c^2, \cdots, h_c^m] \in \mathbf{R}^{m \times d_h}$之后，为进一步在将上下文和属性词进行关联得到上下文和属性词更精准的表达，本文使用层次注意力机制和门机制在上下文和属性词的表示中融入了彼此影响之间的关系，拼接之后再通过分类器得到属性词在上下文中的情感极性分布，具体模型结构如图1所示。

3.1 层次注意力机制

层次注意力机制为本文的两个核心之一，其目的在于在通过层次注意力机制的加权，使上下文和属性词相关关联，表达更加准确。本文先通过上下文对属性词进行注意力机制，得到上下文对属性词的影响矩阵，从而更新属性词的表示；然后通过属性词新的表示，进一步对上下文进行注意力机制，得到属性词对上下文的影响矩阵，进一步更新上下文的表示。注意力机制的层次设计，先获取了属性词的表示，再获取上下文的表示，使属性词和上下文获取了更加准确的表示，同时加深了上下文和属性词之间的关联性；通过对原表示与影响矩阵进行相加然后再加权的结构设计，在一定程度保留原有信息，丰富了新表示的信息，通过多次加权结构，进一步的体现了上下文和属性词中不同词对于属性级别情感分析不同作用性的思想，具体结构如图 2所示。

在通过GRU得到属性词隐藏状态H_T和上下文隐藏状态H_C之后，为了在属性词与上下文之间进行注意力机制，首先需要得到属性词与上下文之间的注意力权重矩阵，如公式1所示：

$$H = relu(H_T \cdot W \cdot H_C^T) = \begin{bmatrix} h_{1,1} & \cdots & h_{1,n} \\ \vdots & \ddots & \vdots \\ h_{m,1} & \cdots & h_{m,n} \end{bmatrix} \tag{1}$$

其中，relu为线型激活函数，$W \in \mathbf{R}^{d_h \times d_h}$为参数矩阵，$H_C^T$为$H_C$的转置矩阵，$H \in \mathbf{R}^{m \times n}$为注意力机制权重矩阵，为属性词中每个词与上下文中每个词之间的关联系数。

首先利用上下文对属性词进行加权，进而得到新的属性词表示如公式2所示：

$$H_T^a = relu[(H_T + HH_C)W_T + b_T] \tag{2}$$

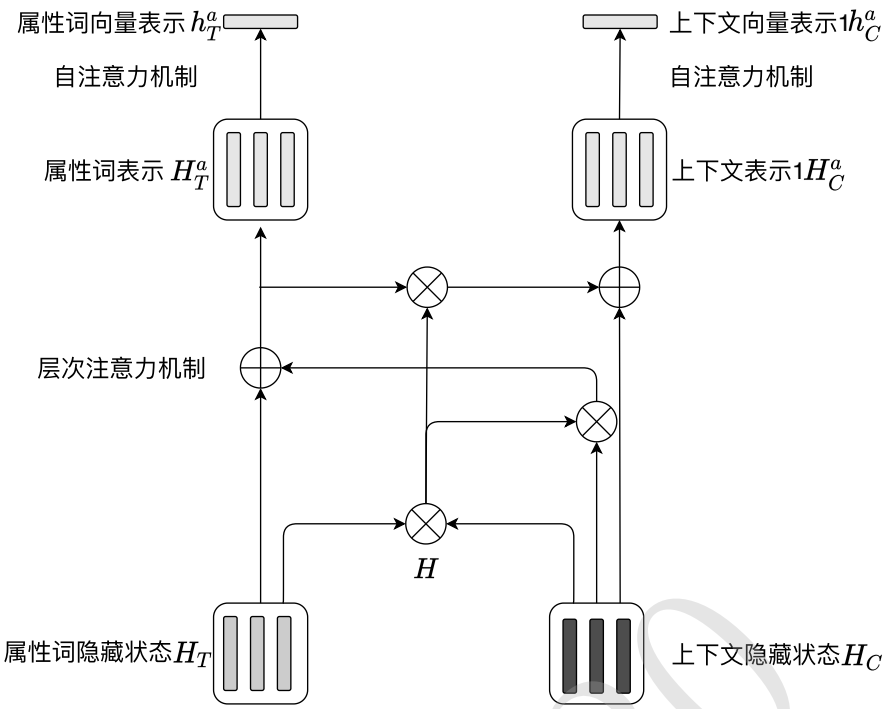


图 2: 层次注意力机制结构图

其中，HH_C表示利用注意力机制权重与上下文相乘，得到上下文对属性词的影响矩阵，然后进一步的与属性词的隐藏状态进行相加，再通过参数矩阵$W_T \in \mathbf{R}^{d_h \times d_h}$进行调整，$b_T \in \mathbf{R}^{m \times d_h}$为偏置矩阵，得到基于层次注意力属性词表示$H_T^a \in \mathbf{R}^{m \times d_h}$。进一步的根据属性词表示$H_T^a \in \mathbf{R}^{m \times d_h}$，得到基于层次注意力上下文表示$H_C^a \in \mathbf{R}^{n \times d_h}$,如公式3所示:

$$H_C^a = relu[(H_C + H^T H_T^a)W_C + b_C] \tag{3}$$

其中，$H^T \in \mathbf{R}^{n \times m}$为$H$的转置，表示上下文中每个词与属性词中每个词之间的关联系数。$H^T H_T^a$表示利用注意力机制权重与属性词新的表示相乘，得到属性词新表示对上下文的影响矩阵，然后进一步的与上下文的隐藏状态进行相加，再通过参数矩阵$W_C \in \mathbf{R}^{d_h \times d_h}$进行调整，$b_C \in \mathbf{R}^{n \times d_h}$为偏置矩阵，得到基于层次注意力属性词表示$H_C^a$。为进一步的突出属性词和上下文中不同词的重要性，本文通过自注意力机制得到属性词向量表示和上下文向量表示1如公式4、5、6、7所示:

$$\alpha = softmax(H_T^a w_T) \tag{4}$$

$$\beta = softmax(H_C^a w_C) \tag{5}$$

$$h_T^a = \sum_{i=1}^{m} \alpha^i H_T^{a,i} \tag{6}$$

$$h_C^a = \sum_{j=1}^{n} \beta^j H_C^{a,j} \tag{7}$$

其中$w_C \in \mathbf{R}^{d_h}$、$w_T \in \mathbf{R}^{d_h}$为参数向量，α、β分别是属性词和上下文自注意力机制系数, $h_T^a \in \mathbf{R}^{d_h}$、$h_C \in \mathbf{R}^{d_h}$分别是属性词向量表示和上下文向量表示，$\alpha^i$、$\beta^j$分别为$\alpha$、$\beta$中的第$i$、$j$个数，$H_T^{a,i}$、$H_C^{a,j}$分别表示$H_T^a$和$H_C^a$中第$i$、$j$个向量。

3.2 门机制

为进一步丰富上下文的表示，本节通过使用门机制选择出针对属性词而言，选择出上下文中重要词信息，并把这一部分当作是属性词对上下文表示的影响矩阵。通过影响矩阵与上下文原表示相加，并在参数矩阵调节下得到上下文表示2H_C^g，具体结构如图3所示。

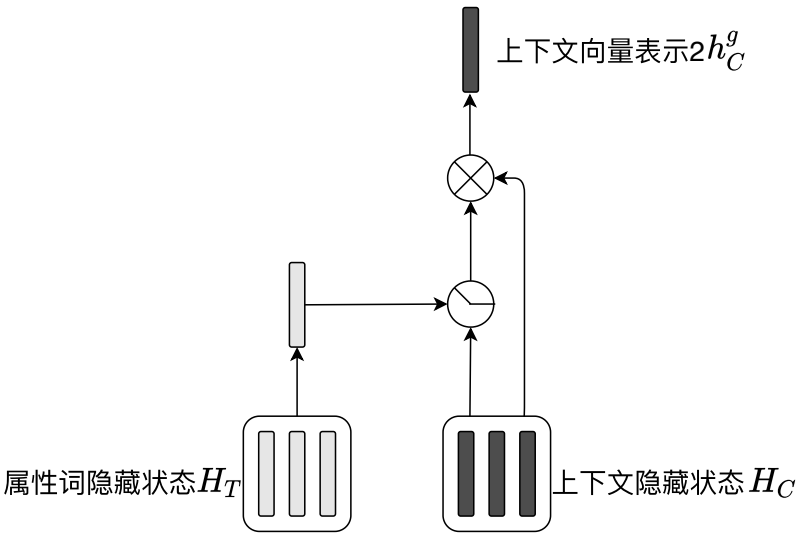


图 3: 门机制结构图

首先，通过将属性词隐藏状态H_T进行平均池化如公式8所示，得到属性词的平均向量表示$h_T^g \in \mathbf{R}^{d_h}$。

$$h_T^g = \sum_{i=1}^{m} H_T/m \tag{8}$$

然后通过门机制对上下文的表示进行调整，并把调整后的上下文表示当作是属性词对上下文表示的影响矩阵，具体门机制计算如公式9、10、11所示。

$$A^i = relu(H_C^i w_A + h_T^g w_g) \tag{9}$$

$$S^i = tanh(H_C^i w_s) \tag{10}$$

$$P^i = A^i \times S^i \tag{11}$$

其中，$w_A \in \mathbf{R}^{d_h}$、$w_g \in \mathbf{R}^{d_h}$、$w_s \in \mathbf{R}^{d_h}$为参数向量, relu、tanh为激活函数，$P^i \in \mathbf{R}^{d_h}$为属性词向量通过门机制对上下文中第i个向量进行选择后的表示，因此可以得到属性词向量对上下文中每个向量进行选择后的表示$P \in \mathbf{R}^{n \times d_h}$。$P$为经过属性词经过门机制对上下文进行选择后表示，本文这部分看作是属性词对上下文的影响矩阵。因此将P与上下文原表示进行相加，并通过参数矩阵进行调整如公式12所示：

$$H_C^g = relu[(H_C + P)W_C] \tag{12}$$

其中，$W_P \in \mathbf{R}^{d_h \times d_h}$为参数矩阵，并通过自注意力机制得到上下文向量表示2$h_C^g \in \mathbf{R}^{d_h}$，如公式13、14所示：

$$\gamma = softmax(H_C^g w_C) \tag{13}$$

$$h_C^g = \sum_{i=1}^{n} \gamma^i H_C^{g,i} \tag{14}$$

其中，γ为上下文注意力机制系数2，γ^i为上下文注意力机制系数2中的第i个数，$H_C^{g,i}$为H_C^g中的第i个向量，通过自注意力机制后的h_C^g为考虑了属性词对上下文这部分影响后的上下文向量表示2。

将上下文向量表示1、属性词向量表示和上下文向量表示2进行拼接之后得到$r \in \mathbf{R}^{3d_h}$,并通过分类器得到属性词在上下文对应的情感极性分布，具体如以下公式所示：

$$J = -\sum_{i=1}^{C} g_i log y_i + \lambda_r (\sum_{\theta \in \Theta} \Theta^2) \tag{15}$$

$$\Theta = \Theta - \lambda_l \frac{\partial J(\Theta)}{\partial \Theta} \tag{16}$$

其中C为情感分类的总数，在本文中$C = 3$，即积极、中性和消极，g_i为属性词在评论中真实的情感分布，y_i为通过模型对属性词情感的预测，Θ为所有参数的集合，λ_r为L_2正则化的参数，λ_l为更新参数的学习率。

4 实验

为验证基于层次注意力机制和门机制的模型的性能，本章将主要对实验设置、实验结果与分析和案例分析进行详细的介绍。

4.1 实验设置

数据集：本文使用的数据集为公开的英文属性级别情感分析数据集SemEval 2014 Task 4 (Pontiki et al., 2014)和Twitter (Dong et al., 2014)数据集，其中SemEval 2014 Task 4由两部分组成，分别是笔记本领域和餐厅领域。数据集的具体分布情况如表1所示。

表 1: 数据集统计

数据集	积极	中立	消极
Laptop-Training	994	464	870
Laptop-Testing	341	169	128
Restaurant-Training	2164	637	807
Restaurant-Testing	728	196	196
Twitter-Training	1561	1560	3127
Twitter-Testing	173	173	346

评价指标：本文使用平均准确率来衡量基于层次注意力机制和门机制的属性级别情感分析模型的性能，具体平均准确的公式如下所示。

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{17}$$

其中TP表示为真实类别为正例，预测类别也为正例；TN表示为真实类别为反例，预测类别也为反例；FP表示为真实类别为反例，预测类别也为正例；FN表示为真实类别为正例，预测类别也为反例。

超参数:本文通过使用GloVe词向量对词进行初始化，词向量维度为200，同时GRU隐藏层的维度也设置为200；对于不在GloVe词典中的词均在[-0.1,0.1]之间随机取值，所有参数矩阵和向量的初始值均在[-0.1,0.1]之间随机选取，偏置的初始值均设置为0；为对参数进行调整，本文使用的优化器是Adam，学习率设置为0.01，同时为了防止过拟合，dropout设置为0.5。

4.2 实验结果与分析

为了衡量本文提出的基于层次注意力机制和门机制模型（HAG）的性能，本选取了7个基准模型与之进行比较，具体如下：

SVM：通过简单的特征工程提取特征之后，再使用SVM分类器进行分类。

TD-LSTM：将评论分成属性词上文与属性词、属性词与属性词下文两部分，然后将两部分分别经过两个LSTM之后的最后一个隐藏向量进行拼接，再通过softmax分类器分类 (Tang et al., 2016a)。

ATAE-LSTM：将平均池化后的属性词向量与上下文中每个词向量进行拼接得到新的上下文矩阵，然后将新的上下文矩阵通过LSTM得到上下文隐藏状态再与属性词向量拼接，接下来通过注意力机制选择出对于属性词而言上下文中较为重要的词，最后通过softmax分类器分类 (Wang et al., 2016)。

MemNet：通过上下文和属性词之间的多层注意力机制加大有助于属性级别情感分析词的权重 (Tang et al., 2016b)。

IAN：通过上下文和属性词之间的交互注意力机制加大对于属性词而言上下文中重要词的权重和加大对于上下文而言属性词中重要词的权重 (Ma et al., 2017)。

RAM:将评论句子输入双向LSTM之后，利用多层注意机制综合句子中的重要特征 (Chen et al., 2017)。

Co-attention：首先将评论分为属性词上文、属性词和属性词下文，然后通过属性词上文与属性词、属性词与属性词下文和属性词上文与属性词下文之间的协同注意力机制得到6个向量表示，将这6个向量进行拼接之后输入到softmax分类器得到属性词的情感分布 (Zhang et al., 2019)。

7个基准模型与HAG的对比实验结果如下表所示。

表 2: 实验结果

	模型	Restaurant(%)	Laptop(%)	Twitter(%)
基准模型	SVM	73.20	66.50	66.50
	TD-LSTM (Tang et al., 2016a)	75.60	68.10	70.80
	AEAT-LSTM (Wang et al., 2016)	77.20	68.70	68.60*
	MemNet (Tang et al., 2016b)	78.16	70.33	68.50
	IAN (Ma et al., 2017)	78.60	72.10	70.40*
	RAM (Chen et al., 2017)	80.23	**74.49**	69.36
	Co-attention (Zhang et al., 2019)	80.40	73.20	71.70
本文模型	Self-attention	79.03	71.63	70.08
	Hierarchical attention + Self-attention	80.26	72.10	71.39
	HAG	**80.50**	72.50	**72.70**

在表2中“*”表示作者根据原论文描述复现后的实验结果，根据实验结果可以看出，传统机器学习的SVM在三个数据集上的效果较差。相对于SVM，TD-LSTM由于使用了LSTM提取特征同时考虑了属性词信息，因此在三个数据集上的实验效果有所提高。ATAE-LSTM通过注意力机制进一步选择出针对不同属性词而言，上下文中不同词的重要性，因此在一定程度上提高了实验的准确率。MemNet通过上下文和属性词之间的多层注意力机制进一步调整了上下文的表示。IAN认为上下文和属性词是相互选择，因此利用交互注意力机，通过上下文对属性词的表示进行加权、属性词对上下文进行加权。与MemNet相比，IAN更加充分的利用了属性词信息，并通过上下文对属性词的表示进行调整，使得实验结果更加准确。Co-attention通过属性词上文与属性词、属性词与属性词下文和属性词上文与属性词下文之间的协同注意力机制，使得属性词上文、属性词和属性词下文三者之间相互选择。本文使用层次注意力机制，在属性词的表示中加入了上下文的影响，进一步得到属性词的表示，使得属性词的表示更加准确；在更新属性词表示之后再更新上下文的表示，层次性注意力机制结果设计使得上下文的表示更加准确，相对于使用自注意力机制，层次注意力机制的增加在一定程度上提高了准确率。其次本文利用了门机制进一步的丰富了上下文的表示。实验结果表明HAG在Restaurant和Twitter数据集上取得了较好的实验室结果，在Laptop数据集低于RAM，这是由于Laptop数据集中含有如否定转移、转折等结构较为复杂的评论，RAM运用多重注意机制综合难句结构中的重要特征，提高了实验的准确率。

4.3 案例分析

本节从餐厅领域选取了一条评论作为案例进行分析，通过注意力机制系数的可视化以验证模型的有效性。如图4所示，其中图的左侧为属性词，右侧为属性词对应的上下文，图中颜色的深浅分别代表了注意力机制系数α和$\beta+\gamma$的大小。通过分析可知，当属性词是*“places”*在上下文

中所关注的词为*“cool”*，而在案例中*“so cool and”*三个词的系数较大，当属性词变为*“service”*的时候，*“is prompt and curious”*几个词的权重加大，*“cool”*的权重降低，由此可见模型的合理性。

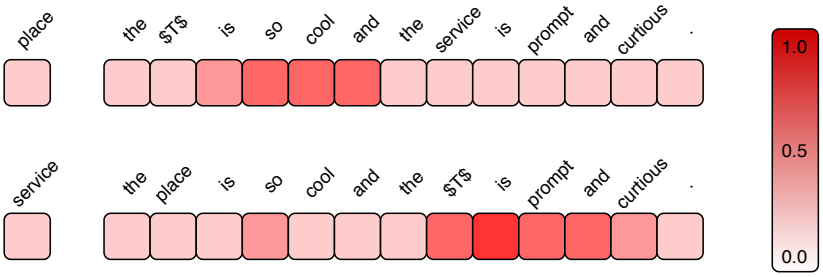


图 4: 案例分析

5 结论

本文提出了一种基于层次注意力机制和门机制的属性级别情感分析方法，通过层次注意力机制先利用上下文对属性词进行选择作用，并将选择后的属性词表示认为是上下文对属性词的影响矩阵从而更新属性词的表示，在得到属性词新的表示之后，再更新上下文的表示，以此在属性词的表示中融入上下文，在上下文的表示中融入属性词信息，层次性的注意力机制结构设计使得上下文和属性词的表示更加准确。其次，本文通过门机制选择出针对于属性词而言上下文中有用的信息，进一步的丰富了上下文的表示，实验结果证明本文提出的HAG模型的有效性。在未来的工作中将会考虑利用更多的信息以丰富词的嵌入表示，如加入一些词性信息、情感信息和位置信息等，在词向量的使用上将会考虑信息更加丰富的BERT词向量,同时会对复杂句式的属性级别情感作进一步研究.

致谢

基金项目:深圳市科创委知识创新基础研究项目(JCYJ20160527172144272)；广东省教育厅特色创新项目(2018GKTSCX069)

参考文献

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 452–461. Association for Computational Linguistics.

Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive recursive neural network for target-dependent twitter sentiment classification. volume 2, pages 49–54, 06.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

Binxuan Huang and Kathleen M Carley. 2019. Syntax-aware aspect level sentiment classification with graph attention networks. *arXiv preprint arXiv:1909.02606*.

Binxuan Huang, Yanglan Ou, and Kathleen M Carley. 2018. Aspect level sentiment classification with attention-over-attention neural networks. pages 197–206.

Lishuang Li, Yang Liu, and AnQiao Zhou. 2018. Hierarchical attention based position-aware network for aspect-level sentiment analysis. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 181–189.

Haihui Li, Yun Xue, Hongya Zhao, Xiaohui Hu, and Sancheng Peng. 2019a. Co-attention networks for aspect-level sentiment analysis. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 200–209. Springer.

Haihui Li, Ting Yuan, Haiming Wu, and Yun Xue. 2019b. Granular computing-based multi-level interactive attention networks for targeted sentiment analysis. *Granular Computing*, 04.

Yunlong Liang, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2019. A novel aspect-guided deep transition model for aspect based sentiment analysis. pages 5568–5579.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. *arXiv preprint arXiv:1709.00893*.

Maria Pontiki, Dimitrios Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 27–35, 01.

Tahura Shaikh and Deepa Deshpande. 2016. A review on opinion mining and sentiment analysis. *International Journal of Computer Applications*, 975:8887.

Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Attentional encoder network for targeted sentiment classification. *arXiv preprint arXiv:1902.09314*.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective LSTMs for target-dependent sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3298–3307, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Duyu Tang, Bing Qin, and Ting Liu. 2016b. Aspect level sentiment classification with deep memory network. pages 214–224.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. *arXiv preprint arXiv:1805.07043*.

Peiran Zhang, Hongbo Zhu, Tao Xiong, and Yihui Yang. 2019. Co-attention network and low-rank bilinear pooling for aspect based sentiment analysis. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6725–6729. IEEE.

Shiliang Zheng and Rui Xia. 2018. Left-center-right separated neural network for aspect-based sentiment analysis with rotatory attention. *arXiv preprint arXiv:1802.00892*.

欧阳志凡. 2018. 基于依存树和注意力的属性级别情感分类研究. Master's thesis, 华南理工大学.

基于循环交互注意力网络的问答立场分析

骆旺达[1]，刘宇瀚[1]，梁斌[1]，徐睿峰[1,†]
1. 哈尔滨工业大学(深圳) 计算机科学与技术学院，广东 深圳 518055
†. 通讯作者，Email：xuruifeng@hit.edu.cn

摘要

针对问答立场任务中，现有方法难以提取问答文本间的依赖关系问题，本文提出一种基于循环交互注意力（Recurrent Interactive Attention, RIA）网络的问答立场分析方法。该方法通过模仿人类阅读理解时的思维方式，基于交互注意力机制和循环迭代方法，有效地从问题和答案的相互联系中挖掘问答文本的立场信息。此外，该方法将问题进行陈述化表示，有效地解决疑问句表述下问题文本无法明确表达自身立场的问题。实验结果表明，本文方法取得了比现有模型方法更好的效果，同时证明该方法能有效拟合问答立场分析任务中的问答对依赖关系。

关键词： 问答立场分析；循环交互注意力；类人学习；注意力机制

A Recurrent Interactive Attention Network for Answer Stance Analysis

Wangda Luo[1], Yuhan Liu[1], Bin Liang[1], Ruifeng Xu[1,†]
1. Department of Computer Science, Harbin Institute of Technology (Shenzhen), Shenzhen, Guangdong 518055, China;
†. Corresponding author, Email: xuruifeng@hit.edu.cn

Abstract

For answer stance analysis task, most existing methods are difficult to extract the significant dependency between questions and answers. To this end, this paper proposes a novel method for answer stance analysis based on a recurrent interactive attention (RIA) network. By imitating the human-like learning method, the proposed model exploits the interactive attention mechanism and the recurrent training iteration for answer stance analysis, which can effectively extracts the dependency between the question and then derives the representation of the stance according to the contextual information of the answer. In addition, to address the problem that the problem text cannot clearly express the corresponding stance, the proposed method presents a novel way of enhancing the representations of the question sentences via switching the question expressions into statements. Finally, the experimental results on the Chinese social media question-answer dataset show that the proposed method achieves the state-of-the-art performance. It also verifies effectiveness of our method in extracting the dependency between questions and answers for answer stance analysis task.

Keywords: Answer stance analysis , Recurrent interactive attention , Human-like learning , Attention mechanism

1 引言

问答立场分析的目的是识别给定的（问题，答案）对中答案相对于问题所表达的立场，如支持、中立和反对。问答数据广泛存在于社交媒体平台中，对问答数据进行合理化分析，能反应社交舆论中人们对各类事物的态度立场，具有重大的学术和商业价值(Küçük and Can, 2020)。问答立场分析任务由Yuan等(2019)提出，其数据样例如表1所示。相比Semeval-2016 task6(Mohammad et al., 2016) 等先前传统的给定分析目标的文本立场检测任务，问答立场分析仅提供了问答文本，并没有提供目标词，致使传统立场模型很难去构建答案相对于问题的立场表示。

问题	雅培奶粉好不好?	世界上有没有美人鱼?
答案	我们都是吃这个，还是不错，但是是从美国买的。	没有。但有一种情况下是有的，那就是你正在作梦。哈哈。
立场	支持	反对

Table 1: 问答立场数据实例

现有的问答立场分析方法主要包括统计机器学习、深度学习和注意力机制等方法。统计机器学习方法主要以诸如词袋模型、支持向量机和逻辑回归等模型为主，它们的目的是拟合问答文本数据中的文本特征，构建概率模型。深度学习方法主要以CNN和RNN为主，将拼接的问答文本作为特征，深度学习模型通过训练特征与立场标签的关系，来构建高效的问答立场分类器。注意力机制方法主要以BiCond(Augenstein et al., 2016)、TAN(Du et al., 2017)和RAM(Chen et al., 2017)等模型为主，它们的目的是通过语义加权的方式，进一步表示问答文本间依赖关系。然而，由于答案相对于问题的立场与问题相对于答案的立场是一致的，上述模型方法都是单方向地构建基于问题的答案依赖表示，忽略了基于答案的问题依赖表示信息。此外，由于问句是一种疑问表述形式，它包含了大量的“能不能”，“有没有”等疑问词，这会导致问题无法有效地传递出一个确切的立场信息。

受阅读理解任务的启发，本文以人类解决问答类问题时的两种方式（即带着问题看答案、从答案中找出问题涉及的关键部分）为基础，并引入多次阅读加深理解的策略，提出了基于循环交互注意力网络的问答立场分析方法来改善此类问题。同时，由于中文社交文本问答数据集的立场标签分布不平衡，容易致使模型无法有效地学习样本类别特征，本文采用focal loss(Lin et al., 2017)的训练手段来改善此类问题。此外，为了改善问题文本无法有效传递其所蕴含的立场信息，本文引入陈述化问句的思想进行优化。

本文的主要贡献如下：1）本文基于模型更容易学习陈述句的思想，将数据集中的疑问句转化成陈述句表示，从而改善了数据集中文本在疑问表述下，问句无法有效表达自身立场的问题；2）本文提出了一种循环交互注意力（Recurrent Interactive Attention, RIA）网络模型，它通过结合基于问题的答案表示和基于答案的问题表示，能有效挖掘问答文本间的依赖关系；3）在中文社交问答数据集上的实验结果表明，本文提出的循环交互注意力网络模型取得了相比现有模型更好的实验效果。

2 相关工作

2.1 问答文本立场分析方法

传统问答立场分析方法可分为基于特征的机器学习方法和深度学习方法两大类。基于特征的机器学习方法主要以句法、语法、情感词、词性和词频等手工特征为基础，通过模型特征训练和筛选等方式构建分类器对问答文本进行立场的分析和检测(Küçük and Can, 2020)。例如，Menini等(2016)利用支持向量机模型，基于立场文本中蕴涵表示、情感词和文本相似度等特征，有效解决政治领域的立场分类问题；同样，Addawood等(2017)以文本间的论辩关系、词汇和语法等作为特征，促进立场目标和文本的交互，提升模型分类效果。深度学习方法主要以RNN和CNN等端到端的模型为主，该类方法主要以拼接的立场文本对作为特征，并通过深度网络对其进行训练，获得分类模型。Lozano等(2017)通过对CNN模型进行改造，引入自动规则生成模块和人工规则，提升模型对特定数据的关注。Dey等(2018)在LSTM(Hochreiter and Schmidhuber, 1997)模型基础上引入注意力机制，通过强化文本中关键词的语义表示，挖掘文

本的立场表示，从而提升模型立场分类性能。总的来说，上述方法在一定程度上改善了立场分析任务的性能，但由于手工特征的构建需要消耗巨大的人力资源，并且忽略了问答文本数据间的交互依赖关系，所以如何从给定文本中挖掘问题和答案间的依赖关系是立场分析任务中亟需解决的问题。

2.2 基于目标依赖的文本分类方法

基于目标依赖的文本分类方法通常是通过对特定目标建模来挖掘目标与上下文词语之间的依赖关系。Xu等(2018)提出了基于Self-Attention的CrossNet模型，其基本思想是从源领域学习一系列特征并将其应用于目标领域。Augenstein等(2016)提出BiCond模型，其以Condition LSTM为基础对立场文本进行编码，构建目标条件依赖下的文本表示。此外，为了加深目标与文本之间的依赖表示，RAM模型(Chen et al., 2017)被提出，其采用循环迭代的思想来强化目标与文本间的关系。在此基础上，Yuan等(2019)提出RCA模型，该模型是问答文本立场任务的强模型，其继承了RAM和AoA(Cui et al., 2016)模型的思想，利用循环依赖的方式提升问答文本中问题与答案间的依赖关系。然而，在问答立场任务中，答案相对于问题的立场与问题相对于答案的立场是一致的，上述模型方法都是单方向地构建基于问题的答案依赖表示，这忽略了基于答案的问题依赖表示信息。

3 模型框架

3.1 问题描述

问答立场分析需要针对给定的问题和答案对来预测答案所表达的立场。具体的，给定问题和答案文本，其中问题文本Q由k个词组成$\{w_1, w_2, ..., w_k\}$，答案文本A由m个词组成$\{w_1, w_2, ..., w_m\}$。问答立场分析任务的目标是对问题和答案文本进行形式化表示，并拟合问题文本Q和答案文本A的交互关系，尽可能准确地分类答案相对于问题的立场（支持、中立和反对）。

3.2 总体框架

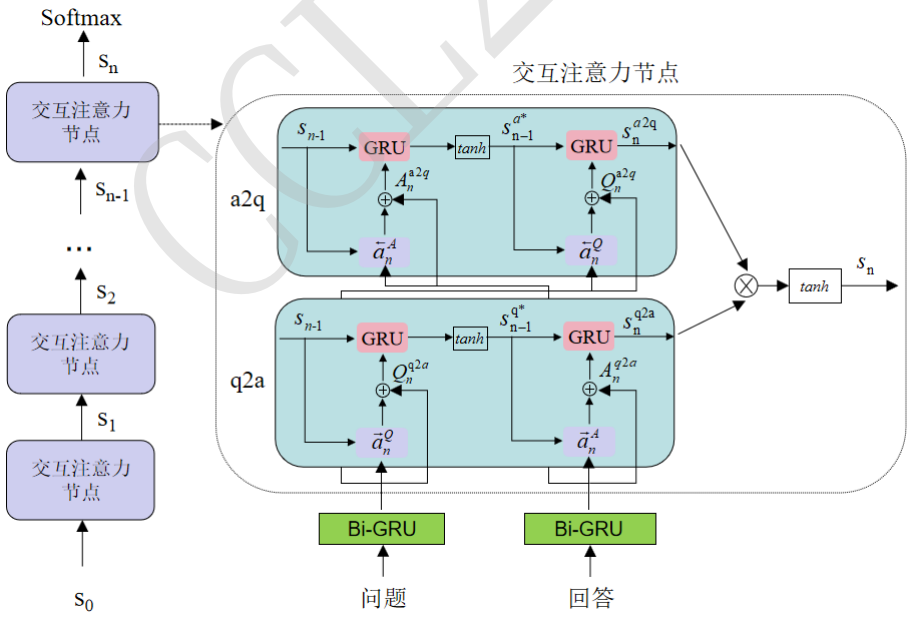


Figure 1: 循环交互注意力网络框架

通常在解决问答类问题时，有一部分人喜欢带着问题看答案，另一部分人喜欢基于答案看问题。此外，有些人还会反复对照问题和答案进行理解并推敲获得最终的结果。从这种类似人类解决问答类问题的思想启发，如图1所示，本文提出了一种基于循环交互注意力的网络模型，来解决问答立场分析任务。该模型由三部分组成，分别为循环网络、基于问题的答案注意力模型和基于答案的问题注意力模型。相比于传统的问答立场分析模型，本文结合问题的答案表示

和答案的问题表示，以获得更强的问答文本间的语义依赖关系。此外，本文还引入了循环网络，目的是反复迭代每次交互注意力节点输出的立场表示，来获得一个结合多次问答理解的立场表示。

如图1所示，该模型以s_n作为经历n次问答理解后的问答立场表示，其中s_0表示中立立场。将立场表示s_{n-1}作为交互注意力节点的输入，结合基于问题的答案注意力表示和基于答案的问题表示，获得新的问答立场表示结果s_n，其计算如公式（1-3）所示。其中，Q和A分别代表将输入文本经过Bi-GRU模型获得的包含上下文语义的问题和答案表示，q2a为基于问题的答案注意力模型，a2q为基于答案的问题注意力模型。将基于问题的答案注意力表示s_n^{q2a}和基于答案的问题表示s_n^{a2q}进行点乘，目的是更好地获得问答文本对中包含真实语义的实词。将结果投入到激活函数tanh中，目的是为模型引入非线性元素，使得模型能更好地拟合问答立场表示结果。

$$s_n^{q2a} = q2a(s_{n-1}, Q, A) \tag{1}$$

$$s_n^{a2q} = a2q(s_{n-1}, Q, A) \tag{2}$$

$$s_n = tanh(s_n^{q2a} \cdot s_n^{a2q}) \tag{3}$$

基于上述方法，模型反复对问答文本进行理解，不断拟合问答立场表示。当循环网络达到预设定的n次阅读理解时，模型停止问答立场表示迭代过程，并将经过n轮问答文本拟合后的立场表示结果s_n作为softmax函数的输入，获得最终的问答立场分类结果。

3.3 基于问题的答案注意力

如图1中的q2a结构所示，基于问题的答案注意力模型目的是根据立场表示信息s_{n-1}、问题表示Q和答案表示A的关系，构建一个基于问题依赖的答案表示。q2a结构由两个注意力机制组成，分别为问题注意力和答案注意力。

针对问题注意力，其思想是通过将历史立场表示信息s_{n-1}和问题表示Q进行注意力交互，来弱化虚词在文本中的作用，以此获得一个加权的问题语义表示，从而更好地抓住问题的重点。其计算如公式（4-6）所示。

$$\vec{u}_{ni}^{Q} = s_{n-1}^{T} \cdot h_i^Q \tag{4}$$

$$\vec{a}_{ni}^{Q} = \frac{\exp\left(\vec{u}_{ni}^{Q}\right)}{\sum_{l=1}^{k} \exp\left(\vec{u}_{nl}^{Q}\right)} \tag{5}$$

$$Q_n^{q2a} = \sum_{i=1}^{k} \vec{a}_{ni}^{Q} \cdot h_i^Q \tag{6}$$

其中，h_i^Q为问题表示Q中每一个词的隐状态，$\vec{u}_{ni}^{Q}$为第n轮问题理解中问题表示第i个词的权值，$\vec{a}_{ni}^{Q}$为第n轮问题理解中问题为第i个词的归一化权值，Q_n^{q2a}为q2a结构中带权的问题表示。

基于获得的带权问题表示Q_n^{q2a}，本文将其与历史立场表示信息s_{n-1}结合，经过单层GRU(Cho et al., 2014)，目的是整合问题和立场信息。此外，为了引入非线性因素改善模型，本文还将整合后的立场信息作为激活函数tanh的输入，获得一个基于问题的立场表示$s_{n-1}^{q^*}$。其计算如公式（7）所示。

$$s_{n-1}^{q^*} = \tanh\left(GRU\left(s_{n-1}, Q_n^{q2a}\right)\right) \tag{7}$$

针对答案注意力，其目的是基于问题立场的前置条件，来更好地抓住答案的重点。该方法利用注意力机制，根据基于问题的立场表示$s_{n-1}^{q^*}$和答案表示A，获得基于问题的加权答案语义表示。其计算如公式（8-11）所示。

$$\vec{u}_{ni}^{A} = \left(s_{n-1}^{q^*}\right)^{T} \cdot h_i^A \tag{8}$$

$$\vec{a}_{ni}^{A} = \frac{\exp\left(\vec{u}_{ni}^{A}\right)}{\sum_{l=1}^{m} \exp\left(\vec{u}_{nl}^{A}\right)} \tag{9}$$

$$A_n^{q2a} = \sum_{i=1}^{m} \vec{a}_{ni}^{A} \cdot h_i^A \tag{10}$$

$$s_n^{q2a} = GRU\left(s_{n-1}^{q^*}, A_n^{q2a}\right) \tag{11}$$

其中，h_i^A为答案表示A中每一个词的隐状态，$\vec{u}_{ni}^{A}$为第n轮问题理解中基于问题的答案表示中第i个词的权值，$\vec{a}_{ni}^{A}$为第n轮问题理解中基于问题的答案表示中第i个词的归一化权值，A_n^{q2a}为q2a结构中带权的答案表示。

3.4 基于答案的问题注意力

基于答案的问题注意力模型目的是构建一个答案依赖的问题表示。同3.3节中基于问题的答案注意力原理，该方法结构与基于问题的答案注意力结构正好相反。该方法首先基于GRU结合历史立场表示信息s_{n-1}和带权的答案表示A_n^{a2q}，并将其通过激活函数tanh，获得基于答案的立场表示$s_{n-1}^{a^*}$。然后，基于GRU将答案的立场表示$s_{n-1}^{a^*}$与带权的问题表示Q_n^{a2q}进行结合，获得基于答案的加权问题语义表示s_n^{a2q}。

3.5 模型训练

给定问题表示Q、答案表示A和循环迭代阈值n，基于循环交互注意力网络模型，可以获得经过n轮问答理解后的问答立场表示s_n。然后，基于softmax函数，将问答立场表示s_n归一化并映射为各立场对应的概率表示p。由于问答立场类别（支持、中立和反对）数据不均衡等问题，为了更好地训练循环交互注意力模型，本文在最小化交叉熵损失函数的基础上，引入focal loss思想进行改善，损失函数计算如公式（12）所示。

$$FL(p) = -a(1-p)^{\gamma}\log(p) \tag{12}$$

其中，a为类别共享权值，用以平衡各类别对总损失的影响;γ为调制系数，用于平衡难分和易分样本的权重。

4 实验

4.1 数据集

数据	支持	中立	反对
训练集	4050	1060	5088
测试集	856	1018	1119

Table 2: 中文社交问答数据集立场标签分布

本文的实验在公开的中文社交问答数据集(Yuan et al., 2019)上进行，数据主要来源于百度知道、搜狗问答等流行互联网社区平台，涉及包括日常生活和医疗疾病等领域内容。中文社交问答数据集中每个数据为一个三元组的表示<问题,答案,立场>，且该数据集的立场类别分布如表2所示。

由于数据集中问题文本是一个疑问表述形式，例如“雅培奶粉好不好？”和“世界上真的有美人鱼吗？”。这种表述形式包含着众多“好不好”，“能不能”和“是不是”等疑问词，使得问题文本不能有效地传递出问题自身是何种立场信息。基于上述问题，本文采用问句陈述化的手段，即基于规则的方式将数据集中的问题文本由疑问表述转换成陈诉表示，例如“雅培奶粉好”和“世界上真的有美人鱼”。陈述化问句的方法，目的是使模型能更高效的明确问题文本的立场信息，促进循环交互注意力模型对问答文本的拟合。

4.2 实验参数设置和评价指标

在循环交互注意力模型中，本文设置问题和答案文本长度分别为25和45（文本过长则进行截断，否则进行0补全），采用100维的Glove.6B词向量(Pennington et al., 2014)作为文本的初始输入。此外，模型中Bi-GRU隐层维度为100，dropout为0.5，模型最小批大小Batch_size为16，模型阅读最大次数n为3，模型训练周期epoch为10轮。在模型训练方面，

本文采用Adam(Kingma and Ba, 2014)作为优化器，其学习率为6×10-4。同时，本文采用focal loss损失函数，其类别共享权值a为0.25，γ调制系数为1.5。

在评估方面，本文在实验上将使用准确率（Accuracy）、支持类标的F1值（F1-支持）、反对类标的F1值（F1-反对）、宏平均（Macro-average F1）和微平均(Micro-average F1)作为问答立场任务的评价指标。其中，针对宏平均和微平均指标，本文与(Yuan et al., 2019)采用相同的评估策略，即不考虑中性立场结果。

4.3 对比方法

为了验证本文提出方法在问答立场分析任务中的有效性，本文将提出的方法和目前取得重要成果的基于统计机器学习方法、深度学习方法和基于注意力机制的深度网络模型在中文社交问答数据集上进行对比实验。各实验介绍如下:

针对基于统计的机器学习方法模型，本文采用逻辑回归（LR）、支持向量机（SVM）、决策树（DT）和词袋模型（BOW）作为基线模型。此类模型主要是基于统计分析的思想，通过统筹文本数据特征及其概率分布，获得一个相对概率最优的模型。

针对深度学习模型，本文采用CNN(Wei et al., 2016)、LSTM和Bi-LSTM(Mrowca et al., 2017)作为基线模型。此类模型是一种端到端的模型，其主要以拼接的问答文本作为模型特征，通过深度神经网络进行学习和拟合。

针对注意力机制模型，本文采用TAN(Du et al., 2017)、IAN(Ma et al., 2017)、BiCond(Augenstein et al., 2016)、AoA(Cui et al., 2016)、RAM(Chen et al., 2017)和RCA(Yuan et al., 2019)作为基线模型。此类模型以问题作为源，以答案作为目标，以注意力机制为基础，基于语义权值构建目标相对于源的依赖关系。其中，RCA模型是目前本任务的最优模型。

4.4 实验结果分析

模型	准确率(ACC)	F1-反对	F1-支持	Macro-F1	Micro-F1
LR	0.5302	0.6034	0.6452	0.6243	0.6282
SVM	0.5289	0.6025	0.6388	0.6206	0.6234
DT	0.5252	0.5728	0.6350	0.6039	0.6076
BOW	0.5132	0.5779	0.6519	0.6157	0.6181
CNN	0.5359	0.6373	0.6422	0.6408	0.6402
LSTM	0.5316	0.6148	0.6541	0.6346	0.6367
Bi-LSTM	0.5747	0.6336	0.6854	0.6599	0.6623
TAN	0.5780	0.6410	0.6917	0.6667	0.6692
IAN	0.5636	0.6418	0.6966	0.6706	0.6713
BiCond	0.5887	0.6623	0.6885	0.6754	0.6771
AoA	0.5864	0.6586	0.6963	0.6775	0.6796
RAM	0.5874	0.6742	0.6885	0.6815	0.6824
RCA	0.6204	0.7066	0.7043	0.7043	0.7037
RIA	**0.6278**	**0.7132**	**0.7107**	**0.7128**	**0.7113**

Table 3: 问答立场分析实验结果

首先，本文在中文社交问答数据集上，评估本文提出的循环交互注意力网络模型（RIA）和各基线系统模型在问答立场任务上的性能。如表3所示，为本文方法与各基线系统模型在问答分析实验上的结果。从表3中可以观察到，基于统计的机器学习方法在各个评价指标上表现最差。普通的深度学习模型效果较基于统计的机器学习方法有所提升。而引入了注意力机制的神经网络模型则有进一步提升。其中，本文提出的RIA 模型取得了最好的问答立场分析性能，在精确率和F1 值方面优于各基线模型，且在Macro-F1和Micro-F1指标上比RCA 模型提升了0.8%。此外，应用交互思想的IAN模型和多次问答理解思想的RAM模型相比于其他基于注意力的模型，均有不同程度的提升。其中，应用循环思想的RAM模型在除同使用该思想的RCA模型外达到最优性能。这证明交互和循环思想的引入对模型性能提升是有作用的，这

也侧面应证了本文提出的RIA模型基于类人类思想进行问答交互依赖和反复理解的策略是有效的。

为了进一步探索多次问答理解对RIA模型的影响，本文以RCA作对比模型，基于问答理解次数参数n进行实验。如图2所示，为RIA和RCA模型在不同问答理解次数下的正确率和Macro-F1成绩指标变化情况。从中，可以看出使用多次问答理解的方法，有助于模型更好地表示问答文本间的依赖关系。在n为[1,4]的范围内，可以看到RIA模型在正确率指标上明显优于RCA模型。在n为[1,7]的范围内，RIA模型在宏平均Macro-F1和微平均Micro-F1指标上优于RCA模型，说明RIA的交互注意力方法在浅层多次阅读理解中能更有效地构建问答文本间的依赖表示并获得问答立场信息。但倘若模型阅读次数过多（$n \geq 7$），可以看到RIA的性能有所下降，可能的原因是过多的问答理解导致了模型的过拟合。

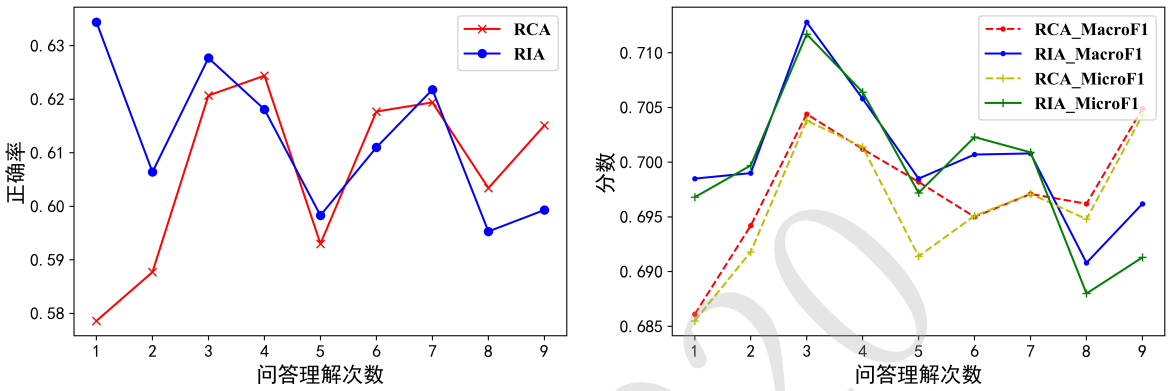


Figure 2: 多次问题理解对实验结果的影响

4.5 消融实验

模型	准确率(ACC)	F1-反对	F1-支持	Macro-F1	Micro-F1
RIA	0.6278	0.7132	0.7107	0.7128	0.7113
RIA w/o q2a	0.5773	0.6795	0.6868	0.6847	0.6838
RIA w/o a2q	0.6030	0.6995	0.6943	0.6978	0.6964
RIA w/o focal loss	0.6181	0.6836	0.7169	0.7005	0.7021
RIA w/o 问题陈诉表示	0.6117	0.6965	0.7077	0.7026	0.7030

Table 4: 消融实验结果

为了验证本文提出的RIA模型中各个部分对实验结果的影响，本文对RIA模型进行了消融实验。RIA模型由四个部分构成，分别为基于问题的答案注意力、基于答案的问题注意力、focal loss和问句陈述化表示。如表4所示，为RIA模型消融实验的结果，其中去除focal loss损失函数部分，本文将采用交叉熵损失函数替代；去除问题陈述化表示，本文将采用原始疑问表示进行替代。从表4中，可以看到去除q2a或a2q结构会导致模型在准确率、Macro-F1、Micro-F1三个指标上下降明显，分别平均下降了3.76%、2.15%和2.12%，说明RIA模型中q2a和a2q的交互对模型性能具有提升作用，证明了交互注意力在问答立场分析任务是有效的。同时，从表4中，对比于交叉熵函数，可以了解到引入focal loss损失函数有助于提升模型性能，证明focal loss损失函数在中文社交问答数据集上对问答立场类别数据不均衡有改善作用。此外，表4证明了问题陈述化表示，能有效提升模型性能，改善疑问表述下模型无法有效识别问题立场的问题。

为了进一步探讨问题陈述化表示对模型性能的影响，本文对问题陈述化表示处理前后的中文社交问答数据结果进行消融分析，其结果如表5所示。从全体测试集上看，采用问题陈述化表示方法后模型准确率指标提升1.61%；从对齐的陈述化表示数据结果上看，采用该方法后模型准确率指标提升1.54%。基于上述结果，可以得出问题陈述化表示方法能有效提升模型性能，更明确地表达出问答文本中问题自身的立场信息。

模型	ACC	F1-反对	F1-支持	Macro-F1	Micro-F1	数据集大小
RIA	0.6278	0.7132	0.7107	0.7128	0.7113	2993
RIA w/o 陈述化表示 I	0.6117	0.6965	0.7077	0.7026	0.7030	2993
RIA	0.6400	0.7307	0.7223	0.7265	0.7258	2874
RIA w/o 陈述化表示II	0.6246	0.7129	0.7191	0.7160	0.7165	2874

Table 5: 基于问题陈述化表示的消融实验（其中，去除问题陈述化表示I为全体疑问表述的测试集；去除问题陈诉表示II为对齐陈述表示的疑问表示测试集）

5 结论

本文以人类解决问答类问题时的两种问答解决方式为基础，同时结合多次阅读加深理解的策略，提出一种基于循环交互注意力（Recurrent Interactive Attention, RIA）网络的问答立场分析方法。该模型首先结合基于问题信息的答案表示和基于答案信息的问题表示来挖掘问题和答案之间的依赖关系。然后，为了获取更精确的问答立场表示，本文使用多次阅读理解的策略迭代更新立场信息来加强问题和答案中能突出立场信息的关键部分。此外，针对疑问句文本在表达立场时信息模糊的问题，本文将文本中的疑问句表示成陈述句来增强文本的立场信息表达。最后，在中文社交问答数据集上的实验结果表明，本文提出的模型取得了相比现有模型更好的效果，从而证明了本文提出方法在问答立场分析任务中的有效性。

致谢

国家自然科学基金61876053, 61632011，深圳市技术攻关项目JSGG20170817140856618, 深圳市基础研究学科布局项目JCYJ20180507183527919, JCYJ20180507183608379, 广东省新冠肺炎疫情防控科研专项项目2020KZDZX1224

参考文献

Dilek Küçük and Fazli Can. 2020. Stance detection: A survey. *ACM Computing Surveys (CSUR)*, 53(1):1–37.

Jianhua Yuan, Yanyan Zhao, Jingfang Xu, and Bing Qin. 2019. Exploring answer stance detection with recurrent conditional attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7426–7433.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41.

Stefano Menini and Sara Tonelli. 2016. Agreement and disagreement: Comparison of points of view in the political domain. In *Proceedings of COLING 2016, the 26th international conference on computational linguistics: Technical papers*, pages 2461–2470.

Aseel Addawood, Jodi Schneider, and Masooda Bashir. 2017. Stance classification of twitter debates: The encryption debate as a use case. In *Proceedings of the 8th International Conference on Social Media & Society*, pages 1–10.

Marianela García Lozano, Hanna Lilja, Edward Tjörnhammar, and Maja Karasalo. 2017. Mama edha at semeval-2017 task 8: Stance classification with cnn and rules. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 481–485.

Kuntal Dey, Ritvik Shrivastava, and Saroj Kaushik. 2018. Topical stance detection for twitter: A two-phase lstm model using attention. In *European Conference on Information Retrieval*, pages 529–536. Springer.

Chang Xu, Cécile Paris, Surya Nepal, and Ross Sparks. 2018. Cross-target stance classification with self-attention networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 778–783, Melbourne, Australia, July. Association for Computational Linguistics.

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. Stance detection with bidirectional conditional encoding. *arXiv preprint arXiv:1606.05464*.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. *arXiv preprint arXiv:1709.00893*.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 452–461.

Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. 2016. Attention-over-attention neural networks for reading comprehension. *arXiv preprint arXiv:1607.04423*.

Jiachen Du, Ruifeng Xu, Yulan He, and Lin Gui. 2017. Stance classification with target-specific neural attention networks. International Joint Conferences on Artificial Intelligence.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988.

Pennington, Jeffrey and Socher, Richard and Manning, Christopher D. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Wan Wei, Xiao Zhang, Xuqin Liu, Wei Chen, and Tengjiao Wang. 2016. pkudblab at semeval-2016 task 6: A specific convolutional neural network system for effective stance detection. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 384–388.

Damian Mrowca, Elias Wang, and Atli Kosson. 2017. Stance detection for fake news identification. *Stanford University, California, US, rep.*

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *Computer Science*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

新型冠状病毒肺炎相关的推特主题与情感研究

梁帅龙
新加坡科技设计大学
新加坡

黄辉
澳门大学计算机与资讯科学系
澳门

张岳
西湖大学工学院
杭州

shuailong_liang@mymail.sutd.edu.sg
derekfw@um.edu.mo
zhangyue@westlake.edu.cn

摘要

我们基于从2020年1月22日至2020年4月30日在推特社交平台上抓取的不同国家和地区发布的50万条推文，研究了有关2019新型冠状病毒肺炎相关的主题和人们的观点，发现了不同国家之间推特用户的普遍关切和看法之间存在着异同，并且对不同议题的情感态度也有所不同。我们发现大部分推文中包含了强烈的情感，其中表达爱与支持的推文比较普遍。总体来看，人们的情感随着时间的推移逐渐正向增强。

关键词： 主题模型；社交媒体分析；新冠肺炎；

Exploring COVID-19-related Twitter Topic Dynamics across Countries

Shuailong Liang[1], Derek F. Wong[2], Yue Zhang[3]
[1]Singapore University of Technology and Design
[2]Department of Computer and Information Science, University of Macau
[3]School of Engineering, Westlake University
shuailong_liang@mymail.sutd.edu.sg
derekfw@um.edu.mo
zhangyue@westlake.edu.cn

Abstract

We investigate the topics and sentiment concerning COVID-19 from half-a-million tweets across different countries between January 22 and April 30, 2020, finding similarities and differences between the general concerns and feelings between countries, and varying attitudes towards different issues. Strong sentiments are found in the tweets, yet love and support also gain much popularity. In general, a trend of increasingly positive sentiment was observed.

Keywords: Topic Model , Social Media Analysis , COVID-19

1 引言

2019年新型冠状病毒肺炎（新冠肺炎）是由2019年新发现的冠状病毒所引起的传染病。由于高死亡率和高传染性，它在全世界已经造成了近千万例感染和近五十万例死亡，213个国家受到影响（截至2020年6月27日）。许多国家都采取了严格的措施来防止这种疾病的传播，包括封锁、居家隔离、社交隔离和旅行禁令等。新冠病毒对世界的经济和政治产生了前所未有的影响。了解人们如何看待和响应政府的政策，他们的关注点、态度和观点以及他们的健康信息需

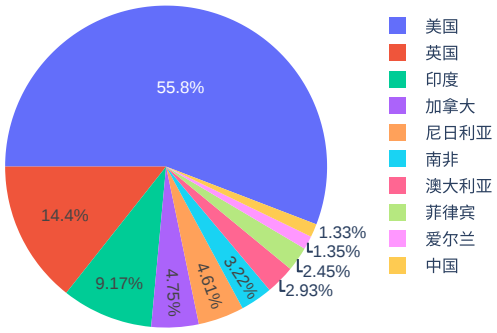


Figure 1: 饼图显示了推文数前十国家推文的比例

国家	推文数
美国	235122
英国	60434
印度	38591
加拿大	20008
尼日利亚	19424
南非	13574
澳大利亚	12348
菲律宾	10310
爱尔兰	5668
中国	5589

Table 1: 推文总数前十的国家的推文数

求和健康寻求行为是至关重要的。从社交媒体中获取的信息甚至可以帮助预测疾病的传播和增长(Turiel and Aste, 2020; Liu et al., 2020; Skiera et al., 2020)。

推特是一个流行的社交平台，有3.3亿月活跃用户，每天有5亿条推文被发出，并且有92.23%的联合国会员国的公民拥有推特账号（截至2019年第一季度）(Aslam, 2020)。与新闻相比，推特是一种更加动态和民主的信息来源，已被广泛用于社交媒体和自然语言处理研究。例如，Fung et al. (2014)和Lancet (2014)使用推特数据研究了埃博拉疫情。与官方新闻文章相比，推特在了解危机的真实叙述方面更具基层性和动态性。我们使用隐含狄利克雷分布（LDA）(Blei et al., 2003)来分析Chen et al. (2020)使用推特应用程序接口收集的从2020-01-22至2020-04-30期间与新冠肺炎相关的推文的主题。我们针对推文数量最多的前十个国家进行分析，分析了几个具有代表性的主题以及人们对这些主题的情感及其动态演变过程。

我们发现在情绪和情感的话题方面，包含了情绪的两个极端：仇恨言论和爱与支持言论。总的来看，人们的情绪是稍微偏积极的，并且随着时间的变化，情绪的积极程度也逐渐提高。不同国家和地区的推文所反映的情绪也各不相同：中国，爱尔兰和印度的情绪相对积极，而美国，澳大利亚和南非的情绪相对消极。人们对于特定主题的情绪，例如**隔离生活**和**病毒起源**，也随着时间而改变。我们还发现新冠肺炎对美国的选举和政治生活有着较为负面的影响。

2 数据集与实验设置

我们使用**COVID-19-TweetIDs**数据集(Chen et al., 2020)获取与COVID-19相关的所有推文ID。截至2020年6月5日，推文总数为1.44亿条，涵盖多种语言，其中英语占60%以上。我们使用Twarc [0]工具从推特应用程序接口获取推文全文（此过程称为hydration）。我们成功收集了从1月22日到4月30日（共100天）总计115,010,623条推文，平均每日超过100万条。

获取全文以后，我们通过将推文对象的*lang*字段限制为*en*来提取英语推文，并仅保留带有*place*属性的推文以包含位置信息。一共获得了498,852条推文。 [1]要研究各个国家的推文主题，我们选择数据集中推文数量最多的前十个国家。表1和图1中显示了按国家和地区分类的推文总数。图2中显示了前十个国家的每日推文计数。在所有国家中，与新冠肺炎相关的具有地理位置信息的推文中，美国的推文数量最多（每日超过2000条），其次是英国和印度。

预处理 过滤出包含地理位置信息的推文后，我们通过删除所有链接和用户提及（@）来对推文进行预处理，然后将主题标记符号#删除。我们使用*emoji*库将表情符号转换为标准文本（例如，将大拇指表情符号转换为*thumbs_up*）。最后，我们将所有文本转换为小写。图3中显示了完整的数据处理流程图。对于每个经过预处理的推文，我们使用NLTK工具包(Bird and Loper, 2004)将其分词，并删除数字，停用词和太短的词（少于三个字符）。此外，我们还删除了直接描述新冠肺炎关键词，例如*COVID19*，*cronavirus*，*COVID*，因为这些词无法提供有关新冠肺炎本身的更多信息。之后，我们去掉了文档频率少于5的词和出现在50%以上的推文中的词。

[0]https://github.com/DocNow/twarc

[1]还有其他可用的COVID19 Tweet数据集，例如 Qazi et al. (2020)，它们可以在基于地名词典的方法中推断地理信息。我们将其留给以后的工作。

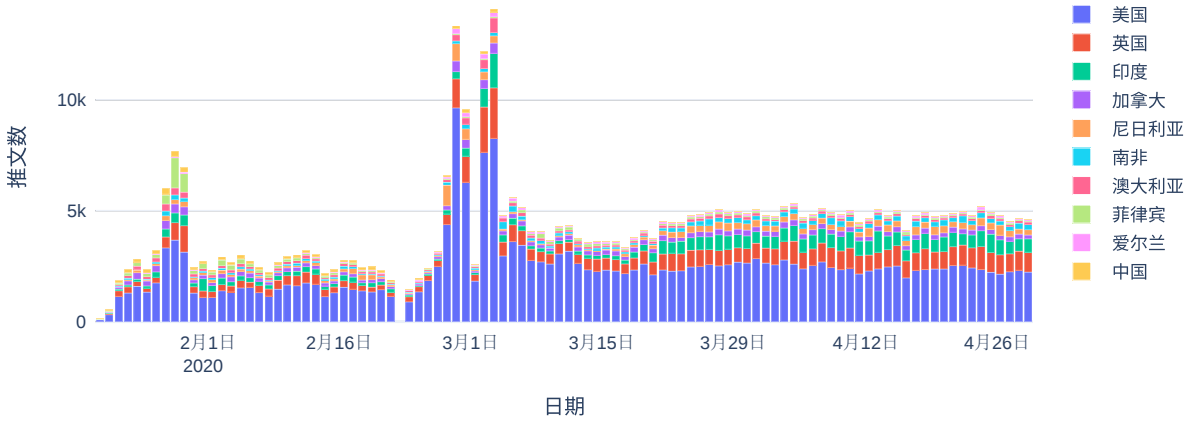


Figure 2: 推文总数前十的国家的每日推文计数。推文最多的国家位于最下方，而推文最少的国家位于最上方。

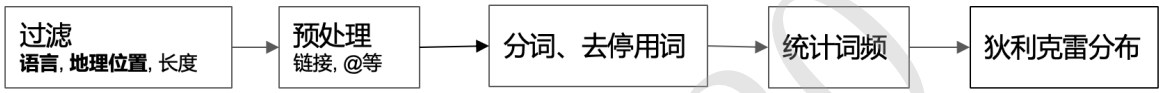


Figure 3: 推文处理流程图

主题模型 我们使用隐含狄利克雷(Blei et al., 2003)模型为每条推文计算话题分布。隐含狄利克雷模型将文档视为词袋（单词顺序无关紧要），它的概率模型的生成过程如下：对于每个文档，首先生成一组主题，然后每个主题选择自己的一组单词。主题的概率分布提供了文档的明确表示。参数估计通常基于变分方法和Gibbs采样。我们使用Gensim(Řehůřek and Sojka, 2010)的MALLET LDA框架(McCallum, 2002)[2]在所有推文上训练主题模型。MALLET框架使用了Gibbs采样的快速且高度可扩展的实现，并提供了非常有效的方法来进行文档主题超参数优化，并提供了工具来根据经过训练的模型推断主题。

为了找到最佳主题数目，我们将主题数目分别设置为$5, 10, 15, 20, 25, 30, 50, 100, 150, 200$，并计算每个主题模型的连贯性得分$C_V$(Röder et al., 2015)。迭代次数设置为2000。我们用每个主题的前20个关键词计算模型的主题连贯性得分，如表2所示。从表2中我们发现主题数为50的模型具有最高的主题连贯性得分。此后增加主题数将逐渐降低连贯性评分。当主题数为20时，相关度得分0.514略低于最高分数0.547，因此在分析中，为了方便研究，我们选择主题数20，没有牺牲太多模型质量。

情感分析 为了调查人们对特定主题的情绪及其随着时间的发展，我们使用了TextBlob [3]提供的基于词典的情感分析工具。对于每条推文，TextBlob返回的情感极性得分在-1和1之间，其中-1代表最负面的情绪，而1代表最正面的情绪。我们计算每日特定主题的推文的情感得分均值。

3 结果与分析

在本节中，我们首先介绍主要主题建模结果，包括提取的主题及其形成关键词。然后，我们讨论随着时间的推移总体和特定国家/地区主题的发展。最后，我们针对特定主题进行情感分析，并比较各国之间推特用户对某个主题的情感。

3.1 话题关键词

表3中显示了为这20个主题提取的关键词。对于每个主题，我们对主题关键词进行汇总，然后发现每个关键词簇都反映了特定主题。**主题名称**列是我们的归纳总结出来的。为了弄清楚每

[2]http://mallet.cs.umass.edu/topics.php
[3]https://textblob.readthedocs.io/en/dev/

话题数量	5	10	15	20	25	30	50	100	150	200
主题连贯性(C_V)	0.350	0.427	0.468	*0.514*	0.522	0.534	**0.547**	0.543	0.528	0.505

Table 2: 具有不同主题数的隐含狄利克雷主题模型的主题连贯性得分。

#	主题名称	主题关键词
0	情绪情感	good thing feel thought bad lot doesn change make sad hear idea hope head guy wow
1	隔离生活	stayhome quarantine eye socialdistancing day fire morning staysafe quarantinelife
2	仇恨言论	shit fuck man gonna lol guy fucking game damn real play joke catch yeah beer ain
3	投票与政治	house woman stupid party man white ppl won hate wrong left racist black power vote
4	特朗普与美国反应	trump president american cdc pandemic response leader lie hoax democrat called
5	封城日子	lockdown day week today month end state hour ago april start coming nigeria lock
6	爱与支持	friend live video lockdown watch night love family today tonight share watching movie
7	保健与医疗	health hospital care test patient public positive doctor medical worker testing tested
8	必需品供给	mask food open face free panic order buy place essential run local store street wear
9	工作	time pandemic work life great working long job hard year good making find start lost
10	爆发与旅行禁令	china country travel flight italy outbreak south japan korea australia iran ship canada
11	信息与媒体	news read medium fear fact question true information story real tweet check article
12	印度反应	india government lockdown fight govt sir situation nation action minister support
13	病毒来源	china wuhan outbreak city control war government problem usa animal bat lab threat
14	流行病与疫苗	people flu year epidemic disease die human vaccine kill million died cure sick outbreak
15	经济危机	business money global market economy pay company crisis big deal stock impact cut
16	祈祷	home stay safe god folded_hands love family hope happy save pray healthy microbe
17	预防与保护措施	social distancing hand spread stop wash measure place avoid protect prevent water
18	学校与学生	school community student team plan kid support child online event group class join
19	情况报告	case death number confirmed update person report rate state infection infected total

Table 3: 每个主题的关键词。由于空间有限，我们仅显示部分关键词。

个主题的主要程度，我们找到每个推文的主要主题。总主题分布显示在图4中。在这里，我们按贡献的多少的顺序讨论这些主题。

仇恨言论 如表3中所示，该主题由关键词*shit*，*fuck*，*damn*，*joke*，*play*等定义，并且包含强烈的个人负面情绪。此主题在大多数推文中表现显著。推特用户使用大写字母表达自己的强烈感情。例如：

> *I'm just convinced that we all gonna die anyways SINCE EVERYBODY AND THEY MOMMAS DONT KNOW HOW TO STAY THE FUCK HOME AND STOP HANGING OUT WITH THEIR FRIENDS. (*我只是相信，我们所有人都会死去，因为每个人和他们的妈妈都不知道如何待在他妈的家中并停止与他们的朋友们闲逛。*)*

上面的示例显示了推文作者对那些不执行居家隔离不停止社交聚会的人们的强烈消极情绪。另一个例子是

> *FUCK COVID-19 ITS PISSING ME OFF FIRST NIGGAS DONT WANNA VOTE CUZ "boo hoo I don't wanna get sick" NOW THEY'RE PLAYING WITH MY DAMN MONEY FUCK 2020 SO FAR MAN THIS SHIT IS RIDICULOUS* （新冠肺炎真是气死我了，黑人不想投票因为"我可不想得病"，现在他们在挥霍我的钱，到现在为止*2020*真是太荒谬了！）

显示了新冠肺炎对人们投票行为的影响。

由于隔离政策和城市关闭，许多人的生活变得不稳定或难以预测。因此，人们表达了强烈的消极情绪。这种现象与之前关于推特仇恨言论研究的一些文献是一致的(Kshirsagar et al., 2018; Hillard, 2018; Drum, 2017).

情绪情感 这个主题与仇恨言论类似。这是一个通用主题，包括诸如*feel*，*think*，*hear*和*hope*之类的关键词。该主题下的推文是关于人们分享其情感，思想和观点的。例如：

> *The best way to not feel hopeless is to get up and do something. Don't wait for good things to happen to you. If you go out and make some good things happen, you will fill the world with hope, you will fill yourself with hope.* （不感到绝望的最好办法就是起来做一些事情。不要等待美好的事情发生在你身上。如果您出去做一些美好的事情，您将使世界充满希望，您也将充满希望。）

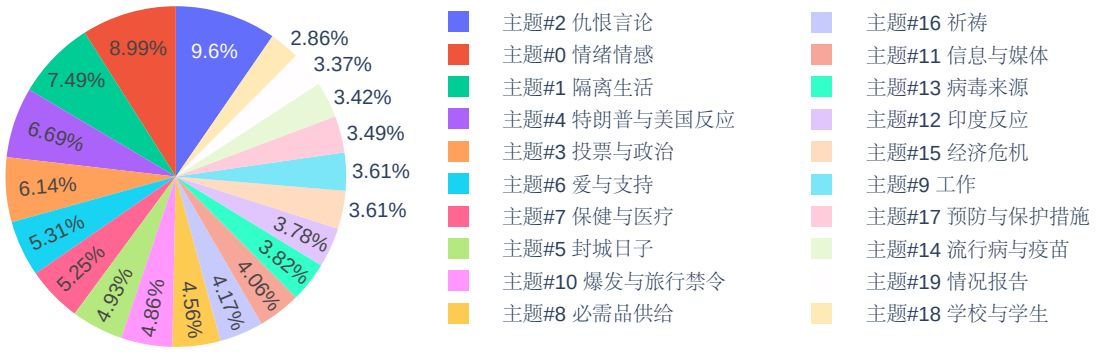


Figure 4: 所有新冠肺炎相关推文的主题分布，按每个主题所对应的推文数量进行排序，并按逆时针顺序显示。

隔离生活/封城日子 人们谈论了居家和社交距离，以及如何度过沉闷的居家时光。我们可以通过*stayhome*，*staysafe*，*quarantinelife*和*socialdistancing*等关键词来发现这个主题。人们焦急地等待封锁的结束。

> *From Thursday, 23 January 2020 to Tuesday, 14 April 2020, there are 82 days That's 2 months and 22 days*（从*2020*年*1*月*23*日星期四至*2020*年*4*月*14*日星期二，一共共*82*天，也是*2*个月*22*天）
>
> *Today was supposed to be the end of this bloody lockdown*（今天本应该是这场该死的封锁结束的日子）

特朗普与美国反应 有关美国总统唐纳德·特朗普和美国对大流行病的回应的讨论占总推文的6.7%，使其成为第四大热门话题。该主题由关键词*leader*，*lie*，*hoax*，*blame*和*truth*标识。我们随机选择几个示例，如下所示。

> *#DonaldTrump Dangerously Suggest Injecting Disinfectant As Treatment For #COVID19...*（特朗普危险地建议将注射消毒剂作为新冠肺炎的治疗方法...）
>
> *Trump failed America by dismantling the CDC pandemic response group in 2018, ignoring warnings, and claiming it was a hoax. The deranged imbecile Trump wasted precious time and lives.*（特朗普在*2018*年撤掉了疾病预防控制中心流行病应对小组，并无视警告，声称这是一个骗局。他辜负了美国。愚蠢的特朗普浪费了宝贵的时间和生命。）

不可避免地，美国民主党和共和党之间进行了大量的政治辩论和指责。我们将在第 3.4部分对美国的情况进行详细分析。

投票与政治 该主题的主要关键词包括*house*，*party*，*black*，*white*，*left*，*power*和*won*。这种流行病显然对2020年美国总统大选产生了根本影响。例如：

> *1. Pandemic emerges that disproportionately kills elderly 2. Masses of elderly congregate at polls during pandemic outbreak to vote for Joe Biden 3. Joe wins primary 4. Elderly voters are culled by pandemic 5. Trump wins in historic landslide*（*1.*新冠大流行导致更多的老年人丧生*2.*在大流行爆发期间的在民意调查显示大量的年长者倾向于投票支持乔·拜登（*Joe Biden*）*3.*乔赢得初选*4.*娇年长的选民在新冠大流行中丧生*5.*特朗普在历史性滑坡中获胜）
>
> *It was "unacceptable to hold an election ... in which people were forced to choose between their safety and voting," wrote new Democratic justice-elect.*（新当选的民主党大法官写道：*"*在选举中人们被迫在安全和投票之间作出选择，这样的选举是不可接受的。*"*）

爱与支持 人们向遭受大流行的国家表示支持，呼吁人类团结在一起度过难关，体现了国际主义的精神。

> *We stand by Italy during these trying times. Share your Support for our Italian friends, They are our colleagues, friends and family. Cari amici, siamo con voi.*（在这些艰难的时刻，我们支持意大利。分享你对我们的意大利朋友们的支持，他们是我们的同事，朋友和家人。亲爱的朋友，我们与您同在。）
>
> *I'm in Sanya, cheering Wuhan on. Hainan rice noodles is also cheering on Wuhan hot-dry noodles!!*（我在三亚，为武汉加油。海南米粉也为武汉热干粉加油！）

保健与医疗 在这场突发公共卫生事件中，人们倾向于讨论自己国家的卫生保健系统，批评其弊端，或者担心医院床位有限，无法适应迅速增加的新冠肺炎感染人数，并关注医务人员的必要保护设备。这个话题的关键词包括*test*，*positive*，*medical*，*worker* 等。

Doctors who have not been provided with Personal Protective Equipments have said that treating patients without the protective gear and masks is akin to a suicide mission.（没有个人防护设备的医生表示，没有防护装备和口罩为患者治疗无异于自杀。）

Uganda Virus Research Institute has the necessary equipment &reagents to test &confirm any suspected COVID-19 sample in country. So far samples from 10 persons who presented with signs & symptoms similar to that of COVID-19 have been tested. All tested negative.（乌干达病毒研究所拥有必要的设备和试剂来测试和确认该国任何可疑的新冠病例。到目前为止，已经测试了*10*名出现类似新冠症状的人的样品。所有检测均为阴性。）

爆发与旅行禁令 此主题的关键词列表中有多个国家/地区名称：中国、意大利、韩国、日本、澳大利亚、伊朗和加拿大。这些国家是最早爆发的国家之一，并在3月初被列为高风险国家。这些国家的公民被拒绝进入许多国家。

British Airways to cancel some Italy, Singapore, South Korea flights: 56 roundtrip flights from Heathrow and Gatwick airports to several destinations in Italy, including Milan, Bologna, Venice and Turin, between 14 - 28 March. (BBC) 英国航空公司取消了意大利，新加坡和韩国的部分航班：*3*月*14*日至*28*日之间，从希思罗机场和盖特威克机场到意大利的多个目的地（包括米兰，博洛尼亚，威尼斯和都灵）的*56*次往返航班。（英国广播公司）

基本服务 对于佩戴口罩是否可以降低新冠感染的风险这个话题引起了大量讨论。由于经济和供应链受到严重影响，食品和卫生产品等必需品供应也受到关注。

Can we spend a moment on stopping the panic buying/hoarding over the coronavirus? Warehouse clubs and stores are being emptied of masks, cleaning products, toilet paper, bread, and eggs in massive amounts for no reason at all. You don't need six months of canned goods either.（我们可以花点时间停止针对冠状病毒的恐慌购买和屯积吗？仓库和商店里大量的口罩，清洁用品，卫生纸，面包和鸡蛋被无故抢购一空。您也不需要六个月的罐头食品。）

FACE MASKS ARE NOT EFFECTIVE PROTECTION AGAINST THE CORONA VIRUS Face masks are effective for preventing spread when worn by those infected, and not by healthy people. Even that, the specialised face masks, N95 respirator, is what's recommended, not a regular surgical mask（口罩不是针对冠状病毒的有效防护。口罩可有效防止被感染者而非健康人群佩戴时扩散。即便如此，还是建议使用专用面罩*N95*呼吸器，而不是常规的手术口罩）

其他主题 由于篇幅所限，我们在此处简要讨论其余主题。主题16 祈祷与主题6 爱与支持较为接近。主题11与新闻、故事和事实的讨论有关，而主题19情况报告是关于确诊病例和死亡人数的每日更新。人们还谈论传染病带来的经济危机（主题15）:大流行导致许多小企业倒闭，许多国家的国内生产总值预期为负。如主题9所示，许多人失业，一些人不得不远程工作。学校和大学关闭，在主题18中得到了讨论。此外，还讨论了新冠肺炎的起源（主题13）和流行病学，包括死亡率、疫苗（主题17）以及预防和保护措施（主题14）。最后，除了美国以外，还有其他特定国家/地区的主题，例如主题12印度反应。

总之，从表3中，我们可以观察到推特上讨论的主要话题，这些主题反映了以下事实：新冠肺炎导致世界上许多国家封锁并要求人们呆在家里以隔绝病毒。结果，商业和经济受到严重影响，人们在家工作，许多人失业。人们将不可避免地表达他们对病毒以及对政府政策的情感和态度，包括责骂和仇恨言论。另一方面，他们也向朋友、家人和医务人员表示爱与支持。人们通过转发新闻、故事和事实（包括假新闻）来获取信息。关于病毒起源的阴谋论也大规模传播。人们也对寻找疫苗、保护自己和控制病毒的措施的有效性感兴趣。在所有国家中，中国作为疫情的首次爆发源地受到了广泛关注。美国拥有最多的推文用户和推文，并且确诊人数也迅速增加。因此，有关唐纳德·特朗普及其政府的讨论也有很多。随着印度确诊病例的增加，印度推特用户还讨论了很多有关印度局势及其政府政策。

3.2 总体话题流

图5中显示了20个主题的推文数量。第一个高峰出现在1月26日（图5点**A**，4千条推文）和1月30日和31日（图5点**B**，9千条推文）。最高峰大约在3月初（图5点**C**，1万6千条推文）。1月26日，中国国务院总理李克强领导一个预防和控制大流行的领导小组，然后决定延长中国春节假期，以遏制大流行(Xinhua, 2020)。最主要的主题是主题13病毒起源，主题2仇恨言论和主题19 情况报告。由于这是最初的爆发点，人们开始谈论病毒的起源，并密切关注病毒的发展。

1月30日，世界卫生组织宣布冠状病毒爆发为国际关注的突发公共卫生事件（PHEIC），并敦促所有国家为遏制疫情做好准备(WHO, 2020)。美国也宣布了国家进入“公共卫生紧急状

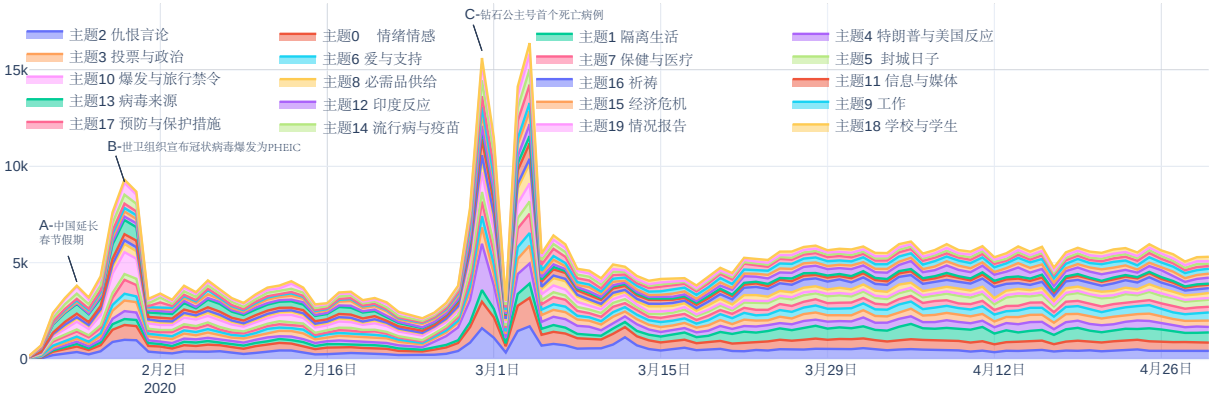


Figure 5: 20个主题所对应的推文的数量随时间的变化趋势。A，B和C是三个峰值。顺序（从下到上）与图4相同。

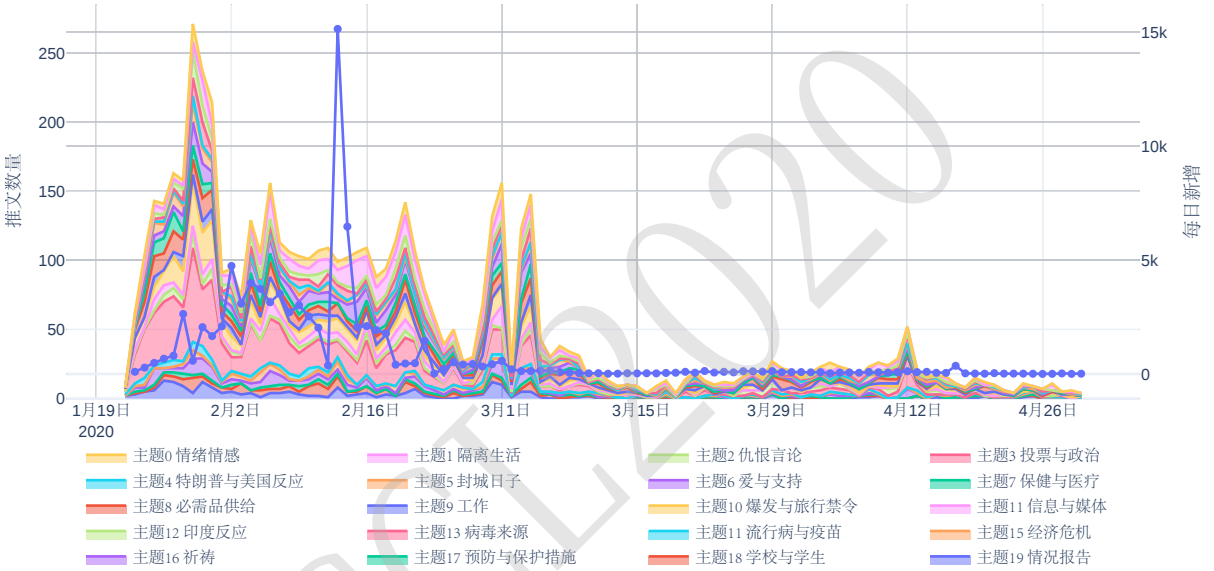


Figure 6: 中国推文话题流与每日新增确诊病例。

态”，两家美国航空公司宣布，在大流行期间取消往返中国的所有航班。1月30日最主要的话题是主题10爆发与旅行禁令，主题2 仇恨言论和主题0情绪情感感。在这个时间点，已经发生了几次爆发，许多国家禁止主要爆发国家的旅行者。人民对此表现出消极情绪。

从2月29日到3月4日，与新冠肺炎相关的推文总数达到了一万六千条，是我们整个研究时期中最高的。钻石公主号发生第一例死亡病例(Martin and Henriques-Gomes, 2020)，美国发生第一次死亡病例(Baker and Crowley, 2020)。美国扩大旅行限制到伊朗、意大利和韩国。最显著的主题是主题4特朗普与美国回应和主题2仇恨言论，这些主题和上述新闻事件紧密相关。

3月中旬之后，与新冠肺炎相关的总推文稳定地增加到将近六千，并且一直保持到研究阶段结束。在此期间，大流行逐渐蔓延到了欧美，确诊的美国病例开始超过中国(Dong et al., 2020)。

3.3 中国话题流与每日新增确诊病例

图6显示了中国各个话题的推文数量的动态变化和每日新增确诊病例数量的联系。由于中国国家防火墙的原因，以及本次研究只考虑英文推特，包含中国位置信息的英文推文数量总体较少。但是从趋势上看，推文数量的变化表现出和新增确诊病例的较强关联，尤其表现在3月1日以后，中国每日新增确诊病例归零之后，相关的推文也逐渐减少。值得注意的是，从1月下旬开始，话题13病毒起源一直占据讨论的话题中心，直到3月初。这也与中国是病毒的最初爆发国有

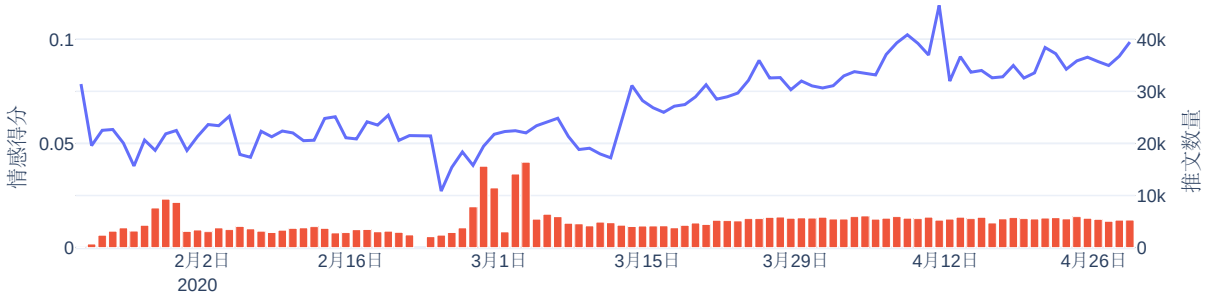


Figure 7: 整体情绪评分。折线显示情感评分，条形图显示推文数量。

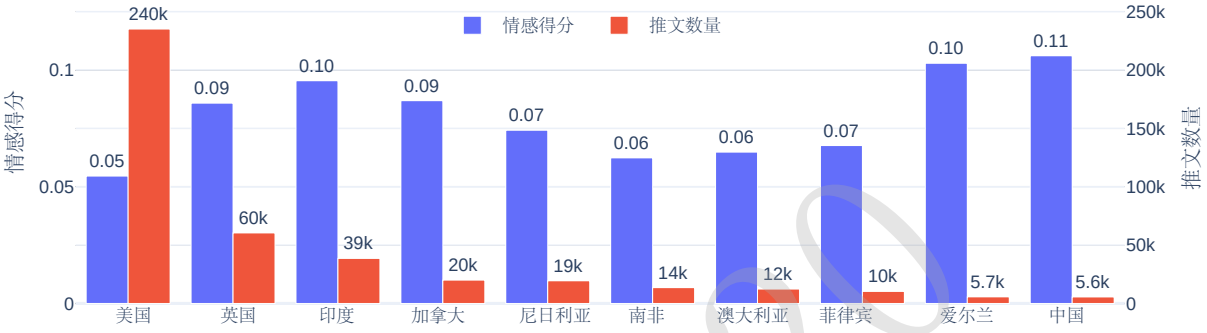


Figure 8: 推文总数前十的国家的综合情感评分和推文数量。左侧的条形图（蓝色）是该国家所有推文的平均情感得分，右侧的条形图（红色）是该国家的推文数量。

关。

3.4 特定国家的主题情绪变化

我们首先讨论与新冠肺炎相关的所有推文的情绪变化，不仅限于特定的国家或主题，然后针对某个主题的情感对特定国家进行分析。

总体情绪发展 图7显示了整体情绪的发展。我们可以看到，总体情绪略微乐观，并在2月底左右达到最低值，此时推文的数量达到了最大值。在最低点之后，情绪值逐渐增加。总的来看，人们对新冠肺炎的讨论持稍微积极的态度，并且随着时间的流逝，这些积极的态度也在增强。

图8中显示了推文数量前十的国家的综合情感评分。美国的推文数量最多，情感评分最低。爱尔兰和中国的推文绝对数量虽然不多，但是情感评分最高。印度，加拿大和英国的情感得分也比较高。

特定主题和国家的情感发展 我们选择了美国，英国，印度和加拿大作为研究对象，选择了主题0情绪情感、主题1隔离生活和主题13病毒起源进行了详细分析，如图9所示。在整个研究时期，这四个国家的推文数量最多。从图9中我们可以看出，这四个国家的情绪总体上是积极的，印度和加拿大的波动要比美国和英国的波动大。对于主题1隔离生活，在3月之前，情绪波动很大，随后逐渐稳定并转为积极状态。在上半年，印度和加拿大的波动最大。有趣的是主题13病毒起源的情绪发展具有相反的趋势：在下半年，情绪波动变得更加强烈，这一趋势在美国和加拿大尤为明显。

对美国特定主题的情感分析 我们选择了主题0情绪情感、主题1隔离生活、主题3投票与政治、主题4特朗普与美国反应、主题6爱与支持和主题13病毒来源，并在图10中说明了人民对各个主题的情感发展动态变化过程。对于主题0情绪情感，情感极性略微正面。对于主题1隔离生活，情绪最低值发生在3月初，然后逐渐上升，表明人民对城市封锁和隔离生活的逐步接受和适应。主题3投票与政治的情绪大多是负面的，表明大流行对美国大选的巨大影响，以及政治领域的指责与博弈。在关于主题13的讨论中，推文的数量在3月初达到顶峰，当时世卫组织宣布新冠肺炎为国际关注的突发卫生事件。人们的情绪在整个时期内都在波动。毫不奇怪，主题6爱与支持的情绪非常积极，并且呈上升趋势。最后，关于主题13病毒起源的推文情绪在4月份出现了较

Figure 9: 特定主题下推文数量前四的国家的推文数量和针对某个主题的情感波动。

大分歧，这可能表明人们对此问题的看法不同。

4 相关工作

最近有很多关于新冠肺炎的社交媒体分析的文献。我们将它们分为三类：数据集获取、社会和心理研究以及以自然语言处理方法为主的研究。

新冠肺炎相关的推特数据集 Yu (2020)创建了一个专用于机构和新闻媒体帐户的推特数据集。Chen et al. (2020)是一项持续的推特收集工作，其追踪可追溯至2020年1月22日的推文。他们利用推特的搜索和流应用程序接口来跟踪特定帐户，并实时收集涉及特定关键词的推文，例如**Coronavirus**（冠状病毒），**covid19**（新冠肺炎）和**social distancing**（社交距离）。GeoCov19(Qazi et al., 2020)是另一个推特数据集，包含超过五亿条多语言推文，可回溯至2020年2月1日。他们使用基于地名词典的方法来推断推文的地理位置。MegaCov(Abdul-Mageed et al., 2020)是一个十亿规模的多语言推特数据集，涵盖234个国家和地区、65种语言，具有超过3200万条带有地理位置标签的推文。此外，还有针对特定语言构建的数据集，例如，Alqurashi et al. (2020)和Haouari et al. (2020)是两个阿拉伯语推特数据集。

社会与心理学研究 Liang et al. (2019)从三个主流媒体的新闻报道中研究了在2008-2010年美国金融危机期间谁指责了谁。 Li et al. (2020)通过词频、情绪指标得分和情绪分析研究新冠肺炎对微博用户的心理后果。 Thelwall and Thelwall (2020)专注于转发次数最多的87条推文（这87条推文产生了1400万条转发），发现推特内容的主要主题包括隔离的生活、安全防护措施、人们对社交限制的态度、政治，以及对与新冠肺炎相关的工作者的关注。他们的工作以定性研究为主，我们的工作则以定量分析为主。 Rajput et al. (2020)在推特数据上使用了词频和情感分析。

关于推特的主题建模 主题建模已被研究人员广泛用于社交媒体分析和信息检索(Sun et al., 2017; Xu et al., 2017)。Wang et al. (2016)使用隐含狄利克雷分布来推断美国总统唐纳德·特朗普的追随者在推特上的话题题偏好，发现关于攻击奥巴马和希拉里·克林顿等民主党人的推特获得了最多的赞。最近，在社交媒体上有很多关于新冠肺炎分析的工作。 Boberg et al. (2020)分析了2020年1月至2020年3月下半月冠状病毒初期脸书上德国地区的帖子。 Wicke and Bolognesi (2020)根据在2020年3月和4月期间以关键词提取的20万条推文的语料库，分析了围绕新冠肺炎的论述。我们的工作和他们的区别在于：1）我们涵盖了更长的时间（2020年1月22日至2020年4月30日）2）我们侧重研究不同国家和地区（包括中国）的主题分布3）我们研究了人们对于相关主题的情感态度随着时间的动态变化。

Figure 10: 美国特定主题的情感得分。折线代表情感得分，条形图代表特定主题的推文数量

5 结论

我们分析了从1月22日（武汉封城前一天）到2020年4月30日的50万条包含了地理位置信息的推文，发现推特上最主要的话题是对新冠肺炎的看法和感受，包括仇恨言论和爱与支持两个极端。总体而言，讨论的气氛较为积极，并且随着时间的演变积极的程度逐渐增强。另外，中国、爱尔兰和印度比美国、澳大利亚和南非更积极。人们对特定主题的态度也随着时间而改变，例如，对隔离生活情感逐渐变得积极，但是不同国家的波动程度有所不同。我们希望我们的研究可以促进对社交媒体平台上的舆论的进一步理解。

致谢

感谢闵庆凯为本文提供许多帮助和建议。崔乐阳和何奇也参与了本文讨论。梁帅龙由新加坡科技设计大学校长奖学金资助。黄辉由澳门特别行政区科学技术发展基金资助（档案编号：0101/2019/A2）。张岳在工作中受到西湖大学融汇金信（www.rxhui.com）联合研究项目资助。张岳为通讯作者。

参考文献

Muhammad Abdul-Mageed, AbdelRahim Elmadany, Dinesh Pabbi, Kunal Verma, and Rannie Lin. 2020. Mega-cov: A billion-scale dataset of 65 languages for covid-19. *arXiv preprint arXiv:2005.06012*.

Sarah Alqurashi, Ahmad Alhindi, and Eisa Alanazi. 2020. Large arabic twitter dataset on covid-19. *arXiv preprint arXiv:2004.04315*.

Salman Aslam. 2020. Twitter by the numbers: Stats, demographics & fun facts.

Mike Baker and Michael Crowley. 2020. Trump calls for calm on virus and expands travel restrictions. The New York Times.

Steven Bird and Edward Loper. 2004. NLTK: The natural language toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 214–217, Barcelona, Spain, July. Association for Computational Linguistics.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Svenja Boberg, Thorsten Quandt, Tim Schatto-Eckrodt, and Lena Frischlich. 2020. Pandemic populism: Facebook pages of alternative news media and the corona crisis–a computational content analysis. *arXiv preprint arXiv:2004.02566*.

Emily Chen, Kristina Lerman, and Emilio Ferrara. 2020. Tracking social media discourse about the covid-19 pandemic: Development of a public coronavirus twitter data set. *JMIR Public Health and Surveillance*, 6(2):e19273.

Ensheng Dong, Hongru Du, and Lauren Gardner. 2020. An interactive web-based dashboard to track covid-19 in real time. *The Lancet infectious diseases*, 20(5):533–534.

Kevin Drum. 2017. Twitter is a cesspool, but it's our cesspool. Mother Jones.

Isaac Chun-Hai Fung, Zion Tsz Ho Tse, Chi-Ngai Cheung, Adriana S Miu, and King-Wa Fu. 2014. Ebola and the social media. *The Lancet, 384(9961), 2207*.

Fatima Haouari, Maram Hasanain, Reem Suwaileh, and Tamer Elsayed. 2020. {ArCOV-19}: The first arabic covid-19 twitter dataset with propagation networks. *arXiv preprint arXiv:2004.05861*.

Graham Hillard. 2018. Stop complaining about twitter - just leave it. National Review.

Rohan Kshirsagar, Tyrus Cukuvac, Kathy McKeown, and Susan McGregor. 2018. Predictive embeddings for hate speech detection on twitter. In *Proceedings of the 2nd Workshop on Abusive Language misc (ALW2)*, pages 26–32, Brussels, Belgium, October. Association for Computational Linguistics.

The Lancet. 2014. The medium and the message of ebola.

Sijia Li, Yilin Wang, Jia Xue, Nan Zhao, and Tingshao Zhu. 2020. The impact of covid-19 epidemic declaration on psychological consequences: a study on active weibo users. *International journal of environmental research and public health*, 17(6):2032.

Shuailong Liang, Olivia Nicol, and Yue Zhang. 2019. Who blames whom in a crisis? detecting blame ties from news articles using neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 655–662.

Dianbo Liu, Leonardo Clemente, Canelle Poirier, Xiyu Ding, Matteo Chinazzi, Jessica T Davis, Alessandro Vespignani, and Mauricio Santillana. 2020. A machine learning methodology for real-time forecasting of the 2019-2020 covid-19 outbreak using internet searches, news alerts, and estimates from mechanistic models. *arXiv preprint arXiv:2004.04019*.

Sarah Martin and Luke Henriques-Gomes. 2020. Coronavirus: man evacuated from diamond princess becomes first australian to die of covid-19. The Guardian.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.

Umair Qazi, Muhammad Imran, and Ferda Ofli. 2020. Geocov19: A dataset of hundreds of millions of multilingual covid-19 tweets with location information. *ACM SIGSPATIAL Special*, 12(1).

Nikhil Kumar Rajput, Bhavya Ahuja Grover, and Vipin Kumar Rathi. 2020. Word frequency and sentiment analysis of twitter messages during coronavirus pandemic. *arXiv preprint arXiv:2004.03925*.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA. http://is.muni.cz/publication/884893/en.

Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining*, pages 399–408.

Bernd Skiera, Lukas Jürgensmeier, Kevin Stowe, and Iryna Gurevych. 2020. How to best predict the daily number of new infections of covid-19. *arXiv preprint arXiv:2004.03937*.

Shiliang Sun, Chen Luo, and Junyu Chen. 2017. A review of natural language processing techniques for opinion mining systems. *Information fusion*, 36:10–25.

Mike Thelwall and Saheeda Thelwall. 2020. Retweeting for covid-19: Consensus building, information sharing, dissent, and lockdown life. *arXiv preprint arXiv:2004.02793*.

Jeremy Turiel and Tomaso Aste. 2020. Wisdom of the crowds in forecasting covid-19 spreading severity. *arXiv preprint arXiv:2004.04125*.

Yu Wang, Jiebo Luo, Richard Niemi, Yuncheng Li, and Tianran Hu. 2016. Catching fire via" likes": Inferring topic preferences of trump followers on twitter. In *Tenth International AAAI Conference on Web and Social Media*.

WHO. 2020. Public health emergency of international concern declared. World Health Organization.

Philipp Wicke and Marianna M Bolognesi. 2020. Framing covid-19: How we conceptualize and discuss the pandemic on twitter. *arXiv preprint arXiv:2004.06986*.

Xinhua. 2020. China to extend spring festival holiday to contain coronavirus outbreak. Xinhua Net.

Zheng Xu, Yunhuai Liu, Junyu Xuan, Haiyan Chen, and Lin Mei. 2017. Crowdsourcing based social media data analysis of urban emergency events. *Multimedia Tools and Applications*, 76(9):11567–11584.

Jingyuan Yu. 2020. Open access institutional and news media tweet dataset for covid-19 social science research. *arXiv preprint arXiv:2004.01791*.

融入多尺度特征注意力的胶囊神经网络及其在文本分类中的应用

王超凡， 琚生根(✉)， 孙界平， 陈润

计算机学院 四川大学 成都 610065
wangcfscu@gmail.com jsg@scu.edu.cn

摘要

近些年来，胶囊神经网络(Capsnets)由于拥有强大的文本特征学习能力已被应用到了文本分类任务当中。目前的研究工作大部将提取到的文本多元语法特征视为同等重要，而忽略了单词所对应各个多元语法特征的重要程度应该是由具体上下文决定的这一问题，这将直接影响到模型对整个文本的语义理解。针对上述问题，本文提出了多尺度特征部分连接胶囊网络(MulPart-Capsnets)。该方法将多尺度特征注意力融入到Capsnets中，多尺度特征注意力能够自动选择不同尺度的多元语法特征，通过对其进行加权求和，就能为每个单词精确捕捉到丰富的多元语法特征。同时，为了减少子胶囊与父胶囊之间的冗余信息传递，本文也对路由算法进行了改进。本文提出的算法在文本分类任务上针对七个著名的数据集进行了有效性验证，和现有的研究工作相比，性能显著提高，说明了本文的算法能够捕获文本中更丰富的多元语法特征，具有更加强大的文本特征学习能力。

关键词： 胶囊神经网络 ； 多尺度特征注意力 ； 文本分类 ； 路由算法 ； 卷积神经网络

Incorporating Multi-scale Feature Attention into Capsule Network and its Application in Text Classification

Chaofan Wang, Shenggen Ju(✉), Jieping Sun, Run Chen

School of Computer Science, Sichuan University, Chengdu, 610065
wangcfscu@gmail.com jsg@scu.edu.cn

Abstract

In recent years, capsule neural networks (Capsnets) has been successfully applied to text classification due to its powerful ability in text feature learning. In previous researches, all the extracted text n-gram features play equal roles in text classification. It is ignored that the importance of each n-gram feature corresponding to a word should be determined by the specific context. This strategy will directly affect the semantic understanding of model to the whole input text. Based on this, this paper proposes Partially-connected Routings Capsnet with Multi-scale Feature Attention(MulPart-Capsnets), which incorporates multi-scale feature attention into Capsnets. Multi-scale feature attention can automatically select n-gram features from different scales, and capture accurately rich n-gram features for each word by weighted sum rules. In addition, in order to reduce the redundant information transferring between child and parent capsules, dynamic routing algorithm is improved too. In order to verify the

effectiveness of the proposed model, our experiments are conducted on seven well-known datasets in text classification. The experimental results demonstrates that the proposed model consistently improves the performance of classification and is able to capture more rich n-gram features of text and possess powerful ability of feature learning.

Keywords: Capsule neural networks , Multi-scale feature attention , Text classification , Routing algorithms , Convolutional neural networks

1 引言

文本分类属于文本挖掘应用中的一个重要组成部分，包括问题分类(Zhang and Lee, 2003)、情感分析(ZHAO Yan-Yan, 2010)和主题分类(Qun et al., 2017)等。现在很多主流文本分类模型一般是基于卷积神经网络(CNN)(Kim, 2014)、循环神经网络(RNN) (Bhowmik et al., 2018)和Transformer(Vaswani et al., 2017)。基于CNN的模型主要是通过利用不同尺度的卷积窗口提取到多元文本特征(比如窗口大小为3就能提取到文本的三元语法特征)，这些文本特征中包含了丰富的上下文信息，能够帮助模型对文本语义进行更好地理解，所以如何准确且全面的捕捉语法特征是模型性能提升的一个关键点。Kim(2014)首次提出通过多个卷积核来对句子进行编码以达到对文本分类的目的。随后各种基于CNN的文本特征模型开始出现在文本分类任务中，例如Zhang等人(2015)引入了一个使用字符级CNN进行文本分类的探索方法。但是在这些现有的基于CNN的研究工作中，为了精简模型参数，通常会对最后的文本特征表示进行池化操作，这将会使模型丢失大量有用的多元语法特征，并且CNN也不能对特征与特征之间的关系进行学习。于是Hinton(2017)等人对CNN进行了改进而提出了胶囊网络(Capsnets)。由于将神经网络中的神经元替换成了张量使得Capsnets而拥有了更加强大的特征学习能力。

Zhao(2018)等人第一次将Capsnets引入到了文本分类领域，研究发现Capsnets比现有的基于CNN和RNN的分类模型的分类效果都要好，也说明了CapsNets在文本分类领域的应用潜力。Zhao(2018)等人虽然通过平均池化在卷积层利用了文本多个尺度的多元语法特征，但是特征的融合方式却十分不合理，因为其忽视了文本内部单词所对应的各个尺度语法特征并不应该是同等重要，而应该是由具体的上下文决定的这一问题，并且这还无形之中将模型的参数规模扩大成了原来的3倍；而Zheng(2020) 等人提出的Capsnets模型只利用了文本的二元语法特征，直接忽视了文本内部还可能存在的其他多元语法特征。可以看出，现有的基于CapsNets的研究工作都不能很好的捕捉丰富的多元语法特征，这将会直接影响到模型对于整个文本的理解，因为只有当那些最重要的多元语法特征被精确提取到的时候，模型才能在考虑具体上下文的基础上正确地理解到单词的意思。基于此，本文提出了多尺度特征部分连接胶囊网络(MulPart-Capsnets)。具体地，本文使用多尺度特征注意力CNN(Wang et al., 2018)作为初级胶囊层的输入，它不仅能通过不同尺度的多元特征之间的注意力来精确捕获文本的多元语法特征，而且还避免了采用多个相似的完全胶囊层而导致的参数规模的增加。

除此之外，在胶囊网络的路由算法中，子胶囊将被路由到每个父胶囊，Ding等人(2019)发现这将会使一部分胶囊成为噪音胶囊。受到这个思想的启发，本文提出了一种能减少噪音路由的算法，那就是去掉一些子胶囊和父胶囊之间的弱连接(权重较小)，从而减少噪音从子胶囊到父胶囊之间的传递。

本文的贡献有以下两点：首先，本文将多尺度特征注意力融入到了Capsnets，其能精确地提取文本中的多元语法特征，使模型拥有强大的文本特征学习能力；其次，为了减少从低层胶囊到高层胶囊的冗余信息传输，本文选择去掉一些子胶囊和父胶囊之间的弱连接(权重较小)来改进了路由算法。

2 相关工作

深度学习在情感分类(Yi-Fu et al., 2019)、文本分类(Kim, 2014)等许多文本挖掘任务中都取得了巨大的成功。现有的文本分类方法主要是基于CNN(Kim, 2014)、RNN(Bhowmik et

al., 2018; Niculae et al., 2017)和Transformer(Vaswani et al., 2017)的。在现有的基于CNN的研究工作中，大部分是利用CNN进行文本多元特征的提取，以完成分类任务。同时，为了精简模型参数，通常会对最后的文本特征表示进行池化操作，这将会使模型丢失大量有用的多元语法特征，并且CNN也不能对特征与特征之间的关系进行学习。为此，Hinton(2017)等人提出Capsnets，将神经网络中神经元替换成了张量以对CNN进行改进。在图像分类领域的实验表明，胶囊网络比CNN具有更强的鲁棒性。Zhao(2018)等人第一次将Capsnets引入到文本挖掘领域，实验结果发现Capsnets比现有的基于CNN和RNN的模型具有更好的文本分类效果。这是因为Capsnets用一组张量来表示文本的特征，而张量的大小方向等又能具体地表示出特征某些方面的性质，这是普通的CNN所不具备的特性，而这个特性恰好能够帮助Capsnets完成更加复杂的特征学习。

卷积神经网络和多元语法特征 CNN由于可以用不同尺寸的卷积窗口来捕捉相邻位置的单词信息，就拥有了对文本多元语法特征进行建模的能力。Kim(2014)首次提出通过多个卷积核来对句子进行编码以达到对文本分类的目的。随后各种基于CNN的模型开始出现在文本分类任务中，例如Zhang等人(2015)引入了一个使用字符级CNN进行文本分类的探索方法；Conneau(2017)等人提出在文本分类中使用非常深层的CNN，因为浅层的CNN不能很好地编码长期依赖信息；Wang等人(2018) 提出了一种多尺度特征注意力CNN，通过在不同窗口尺寸大小的CNN之间做注意力来获取更精确的文本多元语法特征表示，使模型能更好的理解文本语义。

如前文所述，CNN虽然在文本分类领域已经达到了很好的性能，但是其依然存在着的一些根本上的问题，于是学者又对CNN进行了改进而提出了Capsnets。

胶囊网络 胶囊网络首先被应用于图像分类，它在一些分类任务中表现出很强的性能。而后，Zhao等人(2018)首次将胶囊网络用到了文本分类模型当中，并提出了两种结构: 第一种在卷积层采用单尺度特征(卷积窗口大小为3)，第二种在卷积层采用多尺度特征(卷积窗口大小为3, 4, 5)。他的实验证明多尺度特征是优于单尺度特征的，因为多尺度特征包含了更丰富的多元语法信息。然而在第二种结构中Zhao(2018)为了能够利用多尺度语法特征，在最后的分类决策之前对来自不同卷积窗口大小的胶囊文本特征作了平均池化操作。本文认为其忽视了文本内部单词所对应的各个尺度特征并不应该是同等重要的事实，并且这还将模型的参数规模扩大成了原来的3倍。而Zheng等人(2020)却使用单尺度特征的胶囊网络；除此之外，Ding(2019)认为子胶囊与父胶囊之间的全连接路由可能会产生噪音胶囊，他通过将子胶囊与父胶囊分割成包含一定数目的组，让路由在组与组之间进行而改进了动态路由算法，这本质上是一种限制胶囊之间连接数目的改进方法，并且连接数目是静态不可变的，组中的胶囊数目也需要经验的方法来确定。

为了解决以上问题，本文提出了MulPart-Capsnets算法。其将多尺度特征注意力融入到胶囊网络中，使胶囊网络捕获到了更加丰富精确的多元语法特征，还减少了模型的参数；并且本文通过去掉一些子胶囊与父胶囊之间的弱连接(权重较小)使得子胶囊与父胶囊之间的冗余信息传递变少，模型性能得到了进一步的提升。

3 模型

针对现有的Capsnets在文本分类领域存在的不能精确捕捉多元语法特征，以及低层与高层胶囊之间存在冗余信息传递这两个问题，本文提出了MulPart-Capsnets。如图1所示，每层上面的数字表示各层的特征维度，模型的输入是文本T所对应的词向量序列，经过双向循环层之后将会得到一个包含长期依赖关系的全局特征表示；接下来这个特征表示将会被输入到多尺度特征注意力层，这层能够精确捕捉到文本所存在的多元语法特征，然后这些特征将会在部分连接胶囊层进行动态路由而得到高层次的文本特征，最后将在类别胶囊层决定T所属的类别。本节的后续部分，会详细阐述各个部分。

3.1 双向循环层

模型的输入是一个由一系列单词w_1,w_2.....w_n组成的文本T所对应的的d维词向量序列。为了得到w_i的一个全局特征表示，w_i所对应的词向量将分别与其左右所有单词的词向量按次序一起被输入到RNN编码器中，如此便会得到单词w_i的一个上文特征表示$\boldsymbol{C}_l \boldsymbol{W}_{(i)}$(如公式1)和下文特征表示$\boldsymbol{C}_r \boldsymbol{W}_{(i)}$(如公式2)，再将两种特征表示连接起来(如公式3)就能得到wi的上下文的特征表

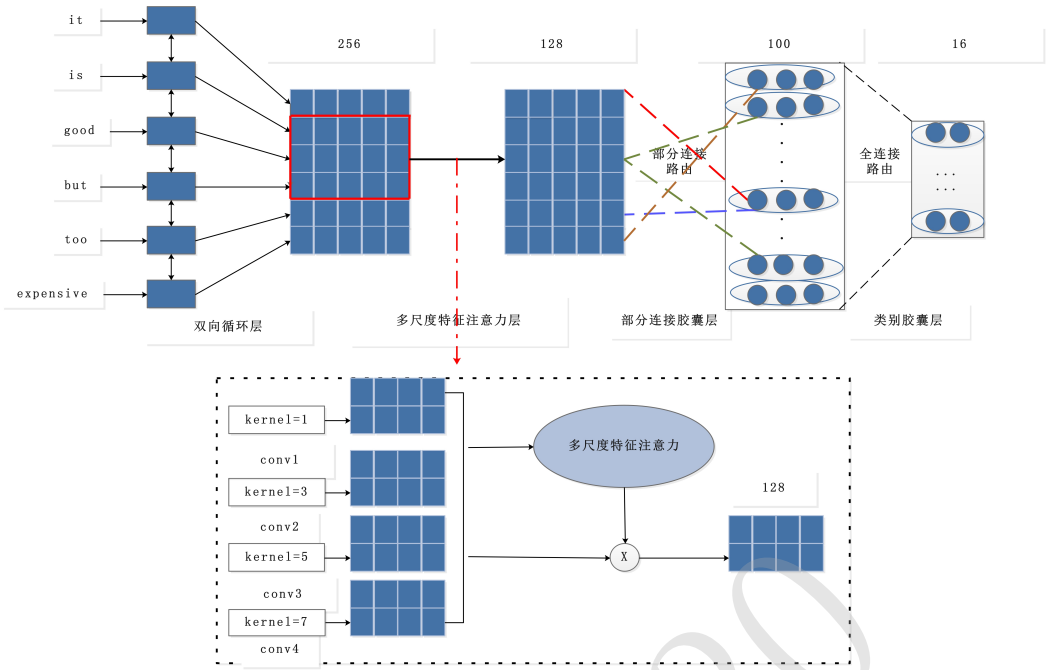


图 1. MulPart-Capsnets模型架构(不同颜色虚线表示不同子胶囊到父胶囊之间的路由)

示$\boldsymbol{x}_i$(如公式3)。

$$c_l(w_i) = RNN(w_i) \tag{1}$$

$$c_r(w_i) = RNN(w_i) \tag{2}$$

$$x_i = [c_l(w_i), c_r(w_i) \tag{3}$$

3.2 多尺度特征注意力层

经过双向循环层，文本T就转换成了一种全局特征表示$\boldsymbol{X:[x_1, x_2,\ldots\ldots, x_i,\ldots\ldots x_n]}$。如公式4，$\boldsymbol{z^i}_l \in \boldsymbol{R}_k$ (假设卷积核个数为k) 表示w_i在卷积窗口大小为l下提取到的的多元(即l元)语法特征表示，其中$\boldsymbol{Conv(.)}$表示对已获得的文本特征表示在其内部单词序列上进行卷积操作，l是指卷积窗口大小；那么，如公式5，$\mathbf{z}_l$就代表了文本T在卷积窗口大小为l下的多元(l元)元语法特征表示。

$$z_l^i = Conv(x_{i-l/2+1}, x_{i-l/2+2}..., x_{i+l/2}) \tag{4}$$

$$z_l = [z_l^1, z_l^2, ..., z_l^n] \tag{5}$$

通过m个不同大小的卷积窗口，最后文本T可被表示为$\boldsymbol{H}$:

$$H = [z_1; z_2...; z_m] \tag{6}$$

在得到了文本T的特征表示$\boldsymbol{H}$之后，我们将利用$Wang(2018)$等人提出的多尺度特征注意力来决定对于一个单词来说哪些多元语法特征是更重要的。

多尺度特征注意力多尺度特征注意力旨在使模型能够自适应地为每个单词选择多元语法特征。本文采用这种方法来精确捕捉文本中存在的多元语法特征。多尺度特征注意力包含两个步骤:卷积特征聚合和尺度特征加权。卷积特征聚合旨在用一个标量$\mathbf{s}^i{}_l$来表示w_i的l元语法特征向量$\mathbf{z}^i{}_l$；尺度特征加权使用$\mathbf{s}^i{}_l$作为输入，并输出注意力权重的$softmax$分布，以重新加权每个单词在不同尺度下的多元语法特征，比如$\mathbf{z}^i{}_1$,$\mathbf{z}^i{}_2$, $\mathbf{z}^i{}_3$,...,$\mathbf{z}^i{}_l$。

卷积特征聚合：本文对每个尺寸的卷积窗口使用k个卷积核，则卷积操作生成的文本T的l元语法特征可表示为: $\mathbf{z}_l = [\mathbf{z}^1{}_l, \mathbf{z}^2{}_l, ..., \mathbf{z}^n{}_l]_{n\times k}$。每个$\mathbf{s}^i{}_l$可由以下公式计算:

$$S_l^i = F_{ensem}(z_l^i) = \sum_{j=0}^{k} Z_l^i(j) \tag{7}$$

其中$F_{ensem}(\bullet)$表示将输入向量的各个分量求和。输出的标量可以作为多元语法特征的一种最终表示。因为$\mathbf{z}^i{}_l$是由w_i在k个卷积核下施加卷积操作产生的，那么这k分量的和在一定程度上则可以作为其特征的一种显著表示。

尺度特征加权：通过卷积特征聚合得到了标量表示$\mathbf{s}^i{}_l$，本文将使用其来生成各个尺度多元语法特征的注意力权重。可以如下定义$\mathbf{z}^i{}_{atten}$和$\boldsymbol{\alpha}^i{}_l$:

$$Z_{atten}^i = \sum_{l=1}^{L} a_l^i z_l^i (\sum_{l=1}^{L} a_l^i = 1 \forall i, 1 \le i \le n) \tag{8}$$

其中$\mathbf{z}^i{}_{atten} \in \mathbf{R}_k$，$\mathbf{s}^i{}_l$是$wi$在各个尺度多元语法特征下的加权表示，$\boldsymbol{\alpha}^i{}_l$是其对应的注意力权重。可以看出，不同大小的卷积窗口对应着不同尺度的多元语法特征。具体地，当$l = 2$，$\mathbf{z}_2$对应着文本T的二元语法特征表示；$l = 3$，$\mathbf{z}_3$对应着文本T的三元语法特征表示。注意力权重由以下公式计算:

$$s_i = [s_1^i, s_2^i ..., s_l^i] \tag{9}$$

$$a_i = soft\max(MLP(s_i)) \tag{10}$$

$$a_i = [a_1^i, a_2^i ..., a_l^i] \tag{11}$$

其中，MLP代表多层感知机。经过注意力模块之后，最后所捕捉到的文本特征被表示为:$\boldsymbol{z}_{atten} = [\boldsymbol{z}^1{}_{atten}, \boldsymbol{z}^2{}_{atten}, ..., \boldsymbol{z}^n{}_{atten}] \in \boldsymbol{R}_{n\times k}$可以认为通过加权之后，此时的$\boldsymbol{z}_{atten}$已经包含了精确且丰富的多元语法特征。然后$\boldsymbol{z}_{atten}$将被送给下一层:部分连接胶囊层。

3.3 部分连接胶囊层

与其他的胶囊网络模型相比,在生成初始胶囊的时候，为了精简模型参数，本文不施加额外的矩阵乘法和卷积操作。对于上一层的输出$\boldsymbol{z}_{atten} \in \boldsymbol{R}_{n\times k}$，本文直接将其视为n个向量长度为k的初始胶囊。在经典的Capsnets中，子胶囊中的信息将会被路由到每一个父胶囊，这种方式同时也会将子胶囊中的一些冗余信息传递到父胶囊，所以我们提出了部分连接路由算法（算法1）来解决这一问题。具体地，我们丢弃掉一些父子胶囊之间的弱连接（权重较小），仅仅使与父胶囊关系最密切的子胶囊被路由。下面将简单对部分连接路由算法进行介绍。在两个邻近的胶囊层之间，为了得到第t层子胶囊$\boldsymbol{u}_i$到第t+1层父胶囊$\boldsymbol{s}_j$的预测向量$\hat{\boldsymbol{u}}_{j|i}$，可以将t层的胶囊ui乘以一个权重矩阵$\boldsymbol{W}_{ij}$得到，如式12所示。

$$\hat{u_{j|i}} = W_{ij} u_i \tag{12}$$

接下来，通过公式13-15便可以由所有预测向量得到每个父胶囊的特征表示。

$$s_j = \sum_i c_{ij} u_{j|i} \tag{13}$$

$$c_{ij} = \frac{\exp(b_{ij})}{\sum_j \exp(b_{ij})} \tag{14}$$

$$v_j = \frac{||s_j||^2}{1 + ||s_j||} \frac{s_j}{||s_j||^2} \tag{15}$$

其中，$\mathbf{c}_{ij}$是动态路由算法决定的耦合系数，通过在原有的$\mathbf{b}_{ij}$的基础之上进行一个softmax函数操作完成，也可以将其视为由$\boldsymbol{u}_i$耦合到$\boldsymbol{s}_j$的先验概率。注意，当进行最后一次路由迭代时，如算法1所示：小于阈值的所有权重$\mathbf{c}_{ij}$都将被丢弃，其他值则将被重新加权同时保持他们的和为1。这样，高层胶囊只能从与之最相关的低层胶囊接收信息，其有助于减少子父胶囊之间的冗余信息传输。

除此之外，父胶囊$\boldsymbol{s}_j$将会通过公式15对进行缩放(这相当于是一个向量版的激活函数)而得到最后的父胶囊$\boldsymbol{v}_j$，这一点和其余的胶囊网络模型是保持一致的。最后，将该层的输出$\boldsymbol{v} \in \boldsymbol{R}_{m\times d}$（d和m分别代表父胶囊的数目和维度）输入到下一层进行最后的分类决策路由。

算法 1 部分连接路由算法

Input: 第t层到t+1层的转换矩阵$\hat{u_{j|i}}$
Output: 第t+1层的胶囊表示V_j

procedure ROUTING($\hat{u_{j|i}}$,R ,t)
for all capsule i in layer t and capsule j in layer (t+ 1): **do** b_{ij}←0.
for R iterations **do**
  for all capsule i in layer t: **do**
    $c_i \leftarrow soft\max(b_i)$ (根据公式14)
  if this is the R-th iteration: **then**
    for all index i in c_i **do**
      if c_{ij} < threshold: **then**
        c_{ij}=0
    $c_i = \frac{c_i}{\sum_j c_{ij}}$
  for all capsule j in layer (t+1) **do** $s_j = \sum_i c_{ij} u_{j|i}$(根据公式13)
  for all capsule j in layer (t+ 1) **do** $v_j = \frac{||s_j||^2}{1+||s_j||} \frac{s_j}{||s_j||^2}$(根据公式15)
  for all capsule i in layer t and capsule j in layer (t+ 1): **do** v_j ←squashing(s_j)
return V_j

3.4 类别胶囊层和损失函数

类别胶囊层作为本模型的最顶层，由C个类别胶囊组成。这一层的每一个胶囊对应一个类别。每个胶囊中向量的长度表示输入文本属于该类别的概率，并且每组向量的方向还保留了其特征的某些特性 (Sabour et al., 2017)，这些特征可以被视为输入样本的特征编码向量。为了增加类别长度之间的差异，本文的模型使用了一个分离的边际损失函数，如公式16所示:

$$L_j = G_j \max(0, m^+ - ||v_j||)^2 + \lambda(1 - G_j) \max(0, m^- - ||v_j||)^2 \quad (16)$$

其中，$\mathbf{v}_j$表示对应的类别j；$\mathbf{m}^+$，$\mathbf{m}^-$分别是上下边界；当且仅当$\boldsymbol{v}_j$被分类正确时，$\mathbf{G}_j$= 1；λ是一个超参数，在本文中取0.5。

4 实验

4.1 实验设置

数据集 本文使用如表1所示的常用7个大规模文本分类数据集(2015)。其中，AG corpus是新闻数据集；DBPedia是来自Wikipedia的本体数据集；Yelp和Amazon语料库是预测情感的用户评论，P表示只需要预测的数据评论的极性，而F表示需要预测评论的星数(1星到5星)；Yahoo.A是一个问答数据集。

	AG	DBP	Yahoo.A	Yelp. P.	Yelp. F.	Amz. F.	Amz. P.
任务	新闻	实体	问答	情感分析	情感分析	情感分析	情感分析
训练集	120k	560k	1.4M	560k	650k	3.6M	3M
测试集	7.6k	70k	60k	38k	50k	400k	650k
平均句子长度	45	55	112	153	155	93	91
类别数量	4	14	10	2	5	5	2

表 1. 数据集信息

超参设置 表2中的第一列和第二列分别列出了为各个为数据集设置的卷积核大小和词汇表大小，因为AG和DBP数据集较小并且句子长度较短，所以本文只为其设置4个尺寸的卷

数据集	卷积核大小	词汇表大小
AG	(1,3,5,7)	100k
DBP.	(1,3,5,7)	500k
Yelp.P.	(1,3,5,7,9)	200k
Yelp.F.	(1,3,5,7,9)	200k
Yahoo.A	(1,3,5,7,9)	500k
Amz.P.	(1,3,5,7,9)	500k
Amz.F.	(1,3,5,7,9)	500k

表 2. 实验设置

积窗口。在本文的实验中，词嵌入使用300D GloVe 840B(2014)进行初始化。在训练模型时，词向量会与其他参数一起进行更新。Adam(2012)被用来优化所有可训练参数；批大小设置为128，输入向量和隐藏状态的维度设置为100或128，部分连接胶囊层中胶囊数目为30，特征长度为100；类别胶囊的维度设置为16。除此之外，为了减少内存和时间开销，本文也将胶囊网络中的权值设置为共享。部分连接路由算法阈值被设置成0.05。

对比模型　本文选择11个常见的文本分类模型作为基线模型（如表3），其中包括一些线性文本分类模型（第一部分），RNN及其变种模型（第二部分），CNN及其变种模型（第三部分）以及胶囊网络模型（第四部分）。

Joulin(2016)：一种简单而又高效的文本分类模型，充分利用了h-softmax的分类功能，遍历分类树的所有叶节点，找到概率最大的标签（一个或者N个）。

Qiao(2018)：应用词袋模型进行文本分类，并为每个单词学习一个局部语境单元以更好利用上下文信息。

Yogatama(2017)：使用长短期记忆网络（LSTM）构建的生成型文本分类模型，比判别型模型更加有效。

Yang(2018)：一种用于文档分类的层次注意力机制网络，在句子级别以及文档级别提出了注意力机制，使得模型在构建文档时是能够赋予重要内容不同的权重。

Zhang(2015)：将字符级的文本当做原始信号，并且使用一维的卷积神经网络来处理文本。

Conneau(2016)：利用了深层次的CNN（29层）提升文本分类算法的精确度。

Wang(2018)：利用多尺度特征注意力CNN捕捉文本中的变长语法特征，并采用稠密连接进一步提升模型的性能。

Niu(2019)：提出两种编码方式来提取文本分类特征，第一层编码提取全局特征，第二层通过全局编码指引局部特征提取。

Xiang(2019)：使用了一种领域嵌入的方法来增强CNN的特征表示能力，考虑到了更加丰富的上下文信息。

Ren(2018)：使用了压缩编码的方法精简了Capsnets模型的参数并使用k均值方法改善了路由算法。

Zhao(2018)：第一次提出了Capsnets在文本分类上的应用，并采用3个相似的Capsnets网络学习文本特征。

评估指标　本文采用的评估指标为准确率(accuracy)。

4.2 主要实验结果

与经典模型的比较　为了验证Capsnets比经典的线性模型、RNN模型以及CNN模型拥有更加强大的特征学习能力，本文了列出了如表3一二三部分所示的实验结果。可以看出，对于Yahoo、Yelp和Amazon所对应的五个数据集，MulPart-Capsnets都达到了最好的分类效果。特别是在Yahoo和Amaz-F数据集上精确度比最好的CNN模型分别提升了0.9和0.5，这是因为这两个数据集的文本平均长度都比较长且目标类别数目较多，这样的文本中包含了大量复杂的语法特征信息，只有拥有强大特征学习能力的模型才能取得好的分类效果。而MulPart-Capsnets在这些数据集上都取得了很好的效果，证明了Capsnets在文本分类任务上的特征学习能力是远远优于CNN，RNN模型的。

模型	AG	DBP	Yahoo.A	Yelp. F.	Yelp. P.	Amz. F.	Amz. P.
(Joulin et al., 2016)	92.5	98.6	95.7	63.9	72.3	60.2	94.6
(Qiao et al., 2018)	92.8	98.9	95.3	64.9	73.7	60.1	95.3
(Yogatama et al., 2017)	92.1	98.7	92.6	59.6	73.7	-	-
(Zhao et al., 2018)	-	-	-	-	75.8	63.6	-
(Zhang et al., 2015)	91.5	98.6	95.4	40.4	71.2	57.6	94.5
(Conneau et al., 2016)	91.3	98.7	95.7	64.7	73.4	63.0	95.7
(Wang et al., 2018)	**93.6**	**99.2**	96.5	66.0	-	63.0	-
(Niu et al., 2019)	93.2	99.0	**96.7**	67.0	75.0	63.1	96.0
(Xiang et al., 2019)	93.1	99.1	96.6	65.9	74.9	62.6	95.9
(Ren and Lu, 2018)	92.4	98.7	96.5	65.9	73.9	61.0	95.0
(Zhao et al., 2018)	92.6	98.7	95.8	65.8	74.0	61.5	94.8
Part-Capsnets	92.5	98.7	96.0	65.7	73.7	61.0	94.6
Mul-Capsnets	92.7	98.9	96.5	67.0	75.6	63.4	95.9
MulPart-Capsnets	93.4	98.9	**96.7**	**67.1**	**75.9**	**63.6**	**96.2**

表 3. 实验结果

另外，在AG和DBP这两个数据集上MulPart-Capsnets并没有达到最好的结果，可能是因为这两个数据集都比较小，且句子长度较短，这样句子所包含的语法特征就会相对稀疏，这种情况对于那些使用了特定技巧的复杂模型（比如稠密连接CNN(2018)，邻域嵌入模型(2019)等）来说无疑是更加有优势的。还有一个可能的原因是对短文本提取尺度大的多元语法特征，可能使句子中某些本来就不存在的多元语法特征被错误地引入到模型当中，比如对一个长度为7的句子提取9-gram特征，显然是不合适的，这一点将会在本节后续部分继续进行讨论。

与Capsnets模型的比较 为了证明在Capsnets中引入多尺度特征注意力是有效的，本文又列出了经典Capsnets模型的实验结果，如表3第4部分所示。其中，Part-Capsnets和Mul-Capsnets分别代表不带多尺度特征注意力和部分连接路由的模型；Ren(2014)提出的模型采取了单尺度的语法特征(在卷积层只用了一种尺寸的卷积窗口)，而Zhao(2018)提出的模型，采用了3个尺寸的卷积窗口(只不过每个卷积窗口都对应了一个完整的胶囊层，这使模型的参数提升到了原来的3倍)。MulPart-Capsnets在全部7个数据数据集上都达到了最好的分类效果，并在其中5个数据集精确度都提升了至少1个百分点以上，特别是在Amaz.F.上提升达到了2.1个百分点。这说明引入了多尺度特征注意力的Capsnets拥有远超其他Capsnets模型特征学习能力，因为MulPart-Capsnets在将文本特征输入到胶囊层之前就已经通过多尺度特征注意力捕捉到了丰富且精确的多元语法特征，精确的特征输入无疑更有利于胶囊层的特征学习。

4.3 参数规模分析

模型	参数
(Zhao et al., 2018)	24M
(Zhao et al., 2018)	8M
(Ren and Lu, 2018)	2.4M
MulPart-Capsnets	**2.0M**

表 4. 模型参数规模比较

在表4中，本文列出了几种基于胶囊网络模型的参数规模。第一个是Zhao(2018)提出的capsule-B，其利用了多尺度多元语法特征，卷积窗口大小为3,4,5；第二和第三个模型为提取单尺度多元语法特征，卷积窗口大小分别为3和2。可以看出，MulPart-Capsnets利用的多元语法特征是最丰富但参数却是最少的。与经典的文本分类胶囊网络不同，MulPart-Capsnets不需要采用几个相似的完全胶囊网络层来获取全面的多元语法特征，因为其在文本特征表示输入到

胶囊网络之前就已经利用多尺度特征注意力捕获了精确的文本语法信息，用较少的参数而得到了更加丰富的多元语法特征。例如capsule-B用24M的参数才获得了文本的3,4,5元语法特征，而MulPart-Capsnets用2M的参数却获得了文本的1,3,5,7,9元语法特征；类似地虽然Ren(2018)利用压缩编码的形式精简了参数，但是其也只是利用了文本的2元语法特征，从而对文本特征的学习能力也远低于MulPart-Capsnets。另一个关键点在于，MulPart-Capsnets设置了比其他模型更少的胶囊数目，原因是当经过多尺度特征注意力层后，输入到胶囊层的文本特征已经是非常精炼且准确的了，所以理论上用较少的胶囊来提取底层低级的特征就已经足够了。

4.4 窗口尺寸对不同数据集的影响

卷积核大小	AG	DBP	Yahoo.A	Yelp. F.	Yelp. P.	Amz. F.	Amz. P.
(1,3)	93.2	99.0	96.1	66.0	75.1	62.4	95.2
(1,3,5)	**93.2**	**99.2**	96.5	66.4	75.4	63.0	95.4
(1,3,5,7)	93.0	98.9	**96.7**	66.9	**75.9**	63.3	96.0
(1,3,5,7,9)	92.6	98.6	**96.7**	**67.1**	**75.9**	**63.6**	**96.2**

表 5. 窗口尺寸对小数据集实验结果的影响

从4.2节的讨论可知，MulPart-Capsnets 在DBP与AG这两个最小的文本分类数据集上实验结果相对较差。本文认为这可能是由于小数据集的句子平均长度较短，句子中所富含的多元语法特征也相对稀缺。所以在将多尺度特征注意力引入到胶囊网络的时引入了一些噪音而影响模型效果。为了验证这个想法，本文对所有数据集限制窗口大小做了如表5的实验。对于AG和DBP这两个小数据集从表中可以看出：当窗口大小被设置为1, 3, 5的时候其分类效果是好于窗口大小为(1, 3, 5, 7) 和(1, 3, 5, 7, 9)的。特别是当增加窗口大小9的时候，精确度甚至下降了0.4%。这说明大的窗口对小数据集的实验效果存在很大的影响。应用越多越大的窗口可能会使模型在小数据集的实验效果严重下降，大的卷积窗口引入了一些文本中原本就不存在的多元语法特征(比如对文本提取9元语法特征，但是一些句子的总长度可能都达不到9)，同理，对于较大的五个数据集，其句子长度较长，包含的语法信息便更加复杂，单词会跟其较远的邻居单词存在依赖关系，所以利用更大的卷积窗口便能提取到这些复杂的语法特征，分类效果会得到明显提升。

4.5 可视化分析

本节将通过可视化的方式进一步阐明多尺度特征注意力和部分连接路由算法在胶囊网络中是如何高效工作的。

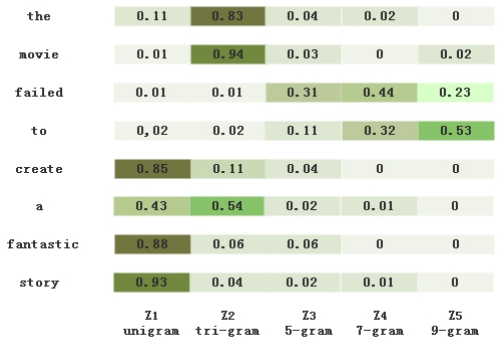


图 2. 多尺度特征注意力的可视化分析

多尺度特征注意力分析 本文对训练好的神经网络中的多尺度特征注意力进行了可视化分析。如图2所示，假设有一个文本输入"the movie failed to create a fantastic story"，图中颜色越深，表示单词所对应部分的的n元语法特征的权重越大，其中每个单词包含了

从1(unigram)到9(9-gram)的5个尺度的特征。可以看出对于”the”, ”movie”,”story”等，模型选择了相对较小尺度的特征；而对于”failed”, ”to”模型却选择了较大尺度的特征。这与人类在理解句子的时候是一致的，”the”, ”movie”, ”story” 用一元语法特征便可以表征出其意思，所以并不太需要前后的上下文来帮助确定；而对于”failed”,”to”模型倾向于使用其较大尺度的语法特征，这不仅使模型捕捉到了一些短语的固定搭配形式，而且还能帮助模型能在考虑更加丰富的上下文的基础之上理解每个单词的释义，从而得到一个更加精准的文本表示以供下一层的胶囊网络利用。所以，本文得出一个结论：将多尺度特征注意力融入到胶囊神经网络，能够使模型自适应地提取多元语法特征，而且所提取到的特征对于胶囊神经网络是十分有帮助的(在未施加平均池化的前提下，得到了文本中丰富的多元语法信息)，这将直接使模型的分类效果得到提升。

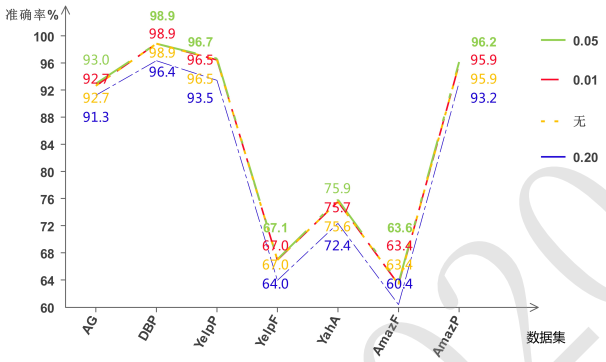


图 3. 权重阈值分析

部分连路接由算法分析 表3中Mul-Capsnets代表不使用部分连接路由算法的胶囊网络模型。如图3所示：红蓝绿黄四条线分别代表阈值设为0.01, 0.2, 0.05和非部分连接路由的MulPart-Capsnets的实验结果。可以看出，当权重阈值取0.01的时候，模型效果与Mul-Capsnets非常相近(即红黄两条线基本完全重合)，因为权重过小，模型丢弃的连接数目非常有限，此时便可以认为模型已经退化成了不带部分连接路由的模型，因此模型效果不会有太大改变。而当阈值取到0.2的时候，MulPart-Capsnets在各个数据集上精确度都严重下降，这点也很好理解，因为设置过大的阈值可能导致一些非常重要的路由信息也被丢失，子胶囊与父胶囊之间的正常路由将被打乱，关键信息不能从子胶囊路由到父胶囊。而当阈值取到0.05的时候模型的效果开始变好，并且比Mul-Capsnets这种不带部分连接路由的模型在每个数据集上大概提升0.3个百分点，说明此时模型丢弃的那些连接恰好是会使模型效果变差的连接，本文直观地将这理解为是子胶囊与父胶囊之间的冗余信息连接。

5 结论

本文提出了一种新的基于胶囊网络的模型MulPart-Capsnets，解决了目前一些文本挖掘工作中将单词所对应各个多元语法特征看作是同等重要的问题。利用多尺度特征注意力，模型能精确地捕捉到文本中的多元语法特征，并且拥有了更加强大的特征学习能力。然后本文分析了全连接路由算法中可能存在的冗余信息传递问题，提出了部分连接路由算法，以减少冗余信息从低层到高层胶囊之间的传递。与传统的胶囊网络相比，新模型参数量更小。最后，文本分类的实验也证明了MulPart-Capsnets拥有的强大特征学习能力。虽然胶囊网络已经在文本挖掘领域取得了不错的成绩，但是其本身也还是一个新兴的神经网络模型，也正处于一个不断发展完善的阶段中。下一步的研究将对其动态路由算法继续进行改进，使模型具有更加强大的特征学习能力。

致谢

本课题得到国家自然基金(61972270)、四川省新一代人工智能重大专项(2018GZDZX0039)和四川省重点研发项目(2019YFG0521)的资助。

参考文献

A Agarwal, S Negahban, and MJ Wainwright. 2012. A simple way to prevent neural networks from overfitting. *Ann. Stat.*, 40(2):1171–1197.

Showmik Bhowmik, Ram Sarkar, Mita Nasipuri, and David Doermann. 2018. Text and non-text separation in offline document images: a survey. *International Journal on Document Analysis and Recognition (IJDAR)*, 21(1-2):1–20.

Alexis Conneau, Holger Schwenk, Loïc Barrault, and Yann Lecun. 2016. Very deep convolutional networks for text classification. *arXiv preprint arXiv:1606.01781*.

Xinpeng Ding, Nannan Wang, Xinbo Gao, Jie Li, and Xiaoyu Wang. 2019. Group reconstruction and max-pooling residual capsule network. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI*, pages 10–16.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701.

Rie Johnson and Tong Zhang. 2017. Deep pyramid convolutional neural networks for text categorization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 562–570.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.

Jaeyoung Kim, Sion Jang, Eunjeong Park, and Sungchul Choi. 2020. Text classification using capsules. *Neurocomputing*, 376:214–221.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Twenty-ninth AAAI conference on artificial intelligence*.

Qiang Li, Yaqian Han, Tong Xiao, and Jingbo Zhu. 2017. Context sensitive word deletion model for statistical machine translation. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, pages 73–84. Springer.

Vlad Niculae, Joonsuk Park, and Claire Cardie. 2017. Argument mining with structured svms and rnns. *arXiv preprint arXiv:1704.06869*.

Guocheng Niu, Hengru Xu, Bolei He, Xinyan Xiao, Hua Wu, and GAO Sheng. 2019. Enhancing local feature extraction with global representation for neural text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 496–506.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Chao Qiao, Bo Huang, Guocheng Niu, Daren Li, Daxiang Dong, Wei He, Dianhai Yu, and Hua Wu. 2018. A new method of region embedding for text classification. In *ICLR*.

Nuo Qun, Xing Li, Xipeng Qiu, and Xuanjing Huang. 2017. End-to-end neural text classification for tibetan. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, pages 472–480. Springer.

Hao Ren and Hong Lu. 2018. Compositional coding capsule network with k-means routing for text classification. *arXiv preprint arXiv:1810.09177*.

Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. 2017. Dynamic routing between capsules. In *Advances in neural information processing systems*, pages 3856–3866.

Dominik Scherer, Andreas Müller, and Sven Behnke. 2010. Evaluation of pooling operations in convolutional architectures for object recognition. In *International conference on artificial neural networks*, pages 92–101. Springer.

Yangyang Shi, Kaisheng Yao, Le Tian, and Daxin Jiang. 2016. Deep lstm based feature mapping for query classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 1501–1511.

Quang Tung Thieu, Marie Luong, Jean-Marie Rocchisani, Nikolay Metodiev Sirakov, and Emmanuel Viennet. 2015. Efficient segmentation with the convex local-global fuzzy gaussian distribution active contour for medical applications. *Annals of Mathematics and Artificial Intelligence*, 75(1-2):249–266.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. *arXiv preprint arXiv:1601.04811*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Shuohang Wang and Jing Jiang. 2015. Learning natural language inference with lstm. *arXiv preprint arXiv:1512.08849*.

Yiren Wang and Fei Tian. 2016. Recurrent residual learning for sequence classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 938–943.

Shiyao Wang, Minlie Huang, and Zhidong Deng. 2018. Densely connected cnn with multi-scale feature attention for text classification. In *IJCAI*, pages 4468–4474.

Liuyu Xiang, Xiaoming Jin, Lan Yi, and Guiguang Ding. 2019. Adaptive region embedding for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7314–7321.

Jiacheng Xu, Danlu Chen, Xipeng Qiu, and Xuangjing Huang. 2016. Cached long short-term memory neural networks for document-level sentiment classification. *arXiv preprint arXiv:1610.04989*.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

ZENG Yi-Fu, LAN Tian, WUZu-Feng, and LIU Qiao. 2019. Bi-memory based attentionmodel for aspect level sentiment analysis. *Chinese Journal of Computers*, 42:1–14.

Wenpeng Yin, Katharina Kann, Mo Yu, and Hinrich Schütze. 2017. Comparative study of cnn and rnn for natural language processing. *arXiv preprint arXiv:1702.01923*.

Dani Yogatama, Chris Dyer, Wang Ling, and Phil Blunsom. 2017. Generative and discriminative text classification with recurrent neural networks. *arXiv preprint arXiv:1703.01898*.

Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2014. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

Matthew D Zeiler. 2012. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Dell Zhang and Wee Sun Lee. 2003. Question classification using support vector machines. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 26–32.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.

Wei Zhao, Jianbo Ye, Min Yang, Zeyang Lei, Suofei Zhang, and Zhou Zhao. 2018. Investigating capsule networks with dynamic routing for text classification. *arXiv preprint arXiv:1804.00538*.

LIU Ting ZHAO Yan-Yan, QIN Bing. 2010. Sentiment analysis. *Journal of Software*, 21-8:1834–1848.

结合深度学习和语言难度特征的句子可读性计算方法

唐玉玲
北京语言大学信息科学学院
blcutyling@163.com

于东*
北京语言大学信息科学学院
yudong_blcu@126.com

摘要

本文提出了可读性语料库构建的改进方法，基于该方法，构建了规模更大的汉语句子可读性语料库。该语料库在句子绝对难度评估任务上的准确率达到0.7869，相对前人工作提升了0.15以上，证明了改进方法的有效性。将深度学习方法应用于汉语可读性评估，探究了不同深度学习方法自动捕获难度特征的能力，并进一步探究了向深度学习特征中融入不同层面的语言难度特征对模型整体性能的影响。实验结果显示，不同深度学习模型的难度特征捕获能力不尽相同，语言难度特征可以不同程度地提高深度学习模型的难度表征能力。

关键词： 深度学习 ； 语言难度特征 ； 句子可读性

The method of calculating sentence readability combined with deep learning and language difficulty characteristics

Yuling Tang
College of Information Science
Beijing Language and Culture
University, Beijing 100083, China
blcutyling@163.com

Dong Yu *
College of Information Science
Beijing Language and Culture
University, Beijing 100083, China
yudong_blcu@126.com

Abstract

In this paper, an improved method for the construction of readable corpus is proposed, and a larger Chinese sentence readability corpus is constructed based on this method. The accuracy rate of the corpus in the task of evaluating the absolute difficulty of sentences can reach 0.7869, which is 0.15 higher than the previous work, which proves the effectiveness of the improvement method. Applying the deep learning method to the evaluation of the readability of Chinese language, the ability of different deep learning methods to automatically capture difficulty characteristics was explored, and the influence of incorporating different levels of language difficulty characteristics into the deep learning features on the overall performance of the model was further explored. The experimental results show that the difficulty features of different deep learning models are different, and language difficulty characteristics can improve the difficulty characterization ability of deep learning models to varying degrees.

Keywords: deep learning , language difficulty characteristics , sentence readability

*为通讯作者

1 引言

作为衡量阅读难度的标准之一，文本可读性对于阅读教学、教材编排有重要意义。可读性体现了给定文本与读者理解文本的认知负荷的关系。这种复杂的关系受到很多因素的影响，如词汇与句法复杂程度、语境和背景知识(Crossley et al, 2017)。传统的可读性研究通过量化不同层面、不同维度的语言特征，如句子长度和单词难度(Davison and Kantor, 1982)，构建多元线性回归公式来评估文本的阅读难度。这些方法因其薄弱的统计基础而受到诟病(Crossley et al, 2017)。随着计算机和自然语言处理技术的发展，越来越多的复杂模型被构建出来应用于文本可读性评估工作(Luo and Callan, 2001; Tanaka et al, 2010; Kate et al, 2010)。有监督的机器学习方法是现行自动评估文本可读性的主流方法。相关研究包括构建统计语言模型评估网页文本难度(Luo and Callan, 2001)，或者把可读性评估任务视为分类任务，构建分类模型预测文本的可读性级别(Collins and Kevyn, 2014; Sung et al, 2015)。从20世纪20年代以来，各个语言的研究者根据自身语言的特点，构建线性或者非线性的模型进行自动评估(Collins and Kevyn, 2014; Wu Siyuan et al, 2020)。这些基于特征工程的方法发现，语言特征的选择对于可读性评估起着重要的作用(Feng and Huenerfauth, 2009)。但有效特征的预测能力与语言特点有关(Feng and Huenerfauth, 2009; Karpov et al, 2014)。这些研究中预测能力强的语言特征是否适用于汉语，已在于东等(2020)的工作中得到验证。

到目前为止，深度学习方法(Goodfellow et al, 2016)在很多自然语言处理任务中都有很好的表现，尤其是与语义相关的任务(Collobert et al, 2011; Zhang,Zhao and LeCun, 2015)，但是只有很少的学者将深度学习方法用于可读性研究，Matrinc等(2019)在几大公开文本级可读性数据集如WeeBit(Vajjala and Meurers, 2012)，OneStopEnglish(Vajjala and Lucic, 2018)，Newsela(Xu, Callison-Burch, and Napoles, 2015)以及Slovenian SB(Matrinc et al, 2019)上分别用HAN(Yang et al, 2016)，BiLSTM(Zhou et al, 2016)和Bert(Devlin et al, 2018)模型进行了有监督的可读性自动评估研究。这项研究是使用现有的深度学习方法在可读性问题上的初尝试，探究了不同深度学习模型在不同数据集上的表现。深度学习模型自动学习到的特征在多大程度上表征难度没有得到验证，这种自动学习获取的特征与人工抽取的语言难度特征的差别体现在何处？现阶段还没有工作使用深度学习方法对汉语可读性问题进行研究，本研究主要是在汉语可读性问题上，结合深度学习方法与外部语言难度特征，探究深度学习方法自动学习获取的特征的表征能力以及与外部语言难度特征表征能力的是否互补的问题。

本文首先参考于东等(2020)基于五点量表和锚点对比构建可读性语料库的方法，提出改进思路，构建了新的句子可读性语料库。基于于东等(2020)构建的语料库（下称set1）和本次构建的语料库（下称set2），探究了机器学习方法在句子绝对难度评估任务上的表现，本次工作使用的语言特征为吴思远等(2020)构造的汉语可读性语言特征体系，包含汉字层面、词汇层面、句法层面。实验结果表明，通过固定标注人员进行标注构造的语料库set2能达到0.7869的准确率。同时，探究了深度学习方法在句子绝对难度评估任务上的表现。实验结果表明，深度学习方法通过自动学习提取特征，能达到比机器学习方法略胜一筹的效果，说明深度学习方法获得的特征可以很好地表征难度。本文试图通过向深度学习特征中加入外部语言难度特征来提高模型的难度表征能力。实验结果表明，外部语言难度特征能不同程度地提高深度学习特征向量的难度表征能力。

本研究的主要贡献包含以下三个方面：第一，构建了一个规模更大、噪点更低、质量更高的句子级可读性标注语料库。该语料库包含37247条汉语句子，具有五个难度等级，为汉语可读性研究提供了数据支持。第二，将深度学习方法应用于汉语句子可读性等级评估任务，验证了深度学习方法在汉语句子可读性等级评估任务上的有效性。第三，通过向深度学习特征中融入外部语言难度特征进行实验，结果表明语言难度特征能不同程度地提高模型整体性能(Tovly Deutsch et al, 2020)。

2 相关研究

自动评估可读性的方法试图发现和利用与可读性感知密切相关的因素来达到自动评估的目的。传统的可读性公式试图建立一个简单的人类可理解公式，其与人类认为的可读性程度有良好的相关性，它们考虑各种统计因素，如词长、句长等。这些公式最初是用于英语的可读性

基金项目：国家社会科学基金(17ZDA305);教育部人文社会科学研究青年基金项目(19YJCZH230);北京语言大学中青年学术骨干支持计划

研究，后来也被借鉴用于其他语言的相关研究。目前，大多数的文本可读性公式都将句子长度和词数纳入计算，如针对成人的Flesh公式(Kincaid et al, 1975)，便于个人使用的SMOG公式(Laughlin, 1969)，估计文本等级的The Gunning Fog公式(Gunning and Robert, 1952)以及用于评价书本的Dale-Chall公式(Dale and Chall, 1948)等。较新的方法是在人工标注的可读性数据集上训练机器学习模型，通过一系列的语言难度特征用来预测给定的无难度标签的文本的难度。这些方法通常依赖于广泛的特征工程，构造许多人类易于理解的特征。

现行的测量可读性的新方法是将其视为一项分类任务，并构建自动预测模型，根据多种特征属性自动预测文本的可读性得分(Schwarm and Ostendorf, 2005; Vajjala and Meurers, 2012; Petersen et al, 2009)。更复杂和适应性更强的方法通常能达到更好的效果，但是需要大量额外的数据资源作为支撑，特征的选择大多依赖专家进行判定，人工抽取特征耗时耗力，所以这些方法在不同语言，不同数据集之间的可迁移性比较差。目前几乎还没有工作涉及到跨语言，多语言，甚至多体裁，多数据库的有监督可读性自动评估。基于多层面语言特征的机器学习方法是可读性自动评估的主流方法，其核心是从词汇、句法和篇章等层面分析和筛选可以预测文本难度的有效特征(Collins and Kevyn, 2014; Pilan et al, 2016)。语言特征的选择与文本的语言属性有关，其他语言研究中的有效特征对汉语特征选择具有启发意义，但不能直接应用于汉语可读性评估(Wu Siyuan et al, 2018; Wang Lei, 2008)。

句子是语言学习中常用的语言单位。也是多项自然语言处理任务的基本处理单元。Pilan等(2016)从第二语言学习的角度探讨了影响瑞典句子难易度的语言因素。该研究将句子可读性评估抽象为多分类问题，支持向量机分类器在该任务上达到了71%的准确率。Dell'Orletta等(2011)对比了表层特征、词汇特征、形态句法特征与句法特征在意大利语文本可读性评估中的作用。他们的研究表明无论是句子级还是文档级别的可读性评估，句法特征都是预测意大利语文本可读性最重要的预测指标。Brunato等(2018)发现，在表层特征，形态句法特征和句法特征中，与句子结构相关的句法特征与英语文本的阅读难度高度相关。Schumacher等(2016)评估了一组句子在有上下文和无上下文条件下的相对阅读难度。该研究使用众包标注的方法收集了人类对句子相对难度的判断，然后使用词法和句法特征训练了逻辑回归模型预测句子对的相对难度。研究发现，词汇相关特征可以帮助预测句子对相对难度，句子在文本中的上下文信息会影响人类对句子难度的判断。句子级的可读性研究受到越来越多的关注，于东等(2020)按照任务的不同把句子级可读性评估分为单句绝对难度评估和句子对相对难度评估两项，通过抽取一系列难度特征训练机器学习模型用于句子可读性自动评估。

国内句子难易度自动评估的研究仍处于起步阶段。江少敏(2009)采用调查问卷和对比分析的方法，从汉字，词汇和句法层面收集了被试者对语言特征预测能力的主观评价，并建立了句子难易度测量公式。庞成(2016)把影响句子难度的因素分为内部结构，外部结构和意义形式三个范畴。郭望皓(2016)对汉字层面和词汇层面的特征进行了量化，并使用CRITIC加权赋值法计算了各指标在预测句子难度上的权重，构建了线性公式。于东等(2020)等通过机器学习的方法进行了语文教材句子的难易度评估工作，也对语言特征的预测作用进行了系统的考察。深度学习方法在英语文本可读性上的应用研究使可读性研究有了进一步的突破(Matrinc et al, 2019)，然而深度学习方法在汉语可读性上的研究工作甚少，本次工作希望句子级可读性问题在深度学习方法上取得突破，并探究融入语言特征的深度学习模型是否具有更好的性能。

3 数据集构建

于东等(2020)的工作中已经构建了一个包含18411条句子的开放的可读性数据集set1，难度标签为5个等级。这个数据集的优点是数据集中的数据来源于权威的语文教材，五点量表的标注方法和锚点对比标注流程的科学性也是不容置疑的。不足之处在于：第一，构建这个数据集的锚点集的数据量太少，各等级占比不均衡；第二，虽然采用众包标注可以节省成本，但是众包标注也意味着难度衡量标准的稳定性更差，噪点更多；第三，标注数据集（标3次）与锚点集（标5次）的标注次数不一致，很大程度上会影响句子的最终标签，偏差更大。针对这些问题，本文提出了相对应的数据集构建改进方法，首先扩充锚点集，然后采取固定标注人员的方式进行标注，每条数据标注5次。基于以上改进方法，我们重新构建了一个句子级可读性数据集set2，我们的句子数据集也是来源于具有权威性的北师大版、人教版和苏教版的汉语语文教材。我们在处理数据的过程中去掉了使用特殊体裁的文本和不完整的文本，如诗歌、词赋、识字文本等，经过句子去重，最后得到的句子数据集包含40192个句子，句子的平均长度为29。

3.1 基于专家标注的锚点句扩充

我们采用锚点比较法进行数据标注，在正式标注之前，首先要构建锚点数据集，我们在于东等(2020)工作的锚点数据集的基础上进行了扩充。首先在原始数据集中选取500条没有进行任何标注的句子集，邀请5名小学语文教师认真阅读句子，并根据五点量表对句子进行等级评定，1表示非常简单，5表示非常难。完成500个句子的难度评定工作大约需要一个小时。最终收集到每个句子被标注5次的数据，5位专家之间的肯德尔一致性系数为0.723（p <0.001）,说明5位专家的标注一致性较高。

对于每一个句子，我们采用多数投票原则确定锚点句的难度等级。为了保证作为锚点句的句子难易程度一致，我们计算了每个句子被标注为最终难度的概率。如句子A被标注了5次，其中三位专家标注A的难度为等级3，一位专家标注A的难度为等级1，一位专家标注A的难度为等级4，那么该句子A最终难度为等级1的概率为0%，等级2的概率为20%，等级3的概率为60%，等级4的概率为20%，等级5的概率为0%。我们选取概率大于或者等于80%的难度等级作为该句子的最终等级，并确定该句子为锚点句。

经过概率筛选后，我们确定了205个句子的最终难度等级，除去难度为5的10个句子，剩余的195个句子为最终的锚点句，其中，等级一的锚点句数量为60句，等级二的锚点句数量为48句，等级三的锚点句数量为75句，等级四的锚点句数量为12句。为了保证四组锚点句之间在难度上具有较高的差异性，对四组锚点句的难度差异进行了测量，单因素方差分析结果显示，四组句子的难度差异显著（F=580，p <0.01）。更多统计信息如表1所示。

锚点等级	于东(2020)	比例	本文锚点集	比例	相对差值
等级一	33	53%	60	31%	+23
等级二	16	26%	48	25%	+32
等级三	10	16%	75	38%	+65
等级四	3	5%	12	6%	+9

表 1: 锚点句对比详情统计

3.2 基于锚点比较的数据标注

3.2.1 标注流程

我们共招募了20名标注员对数据进行标注，标准规则为与锚点句成对比较，每个句子将在2-3次比较后被划分到最终难度等级。我们收集了标注员的年龄，性别，教育程度等个人信息，标注者年龄在19至27岁之间，学历为本科到博士，男女比例为1:5。在正式标注之前，对标注人员进行简单培训，明确标注任务和规则，然后客观负责地完成任务。我们每天定时在微信标注小程序上发布标注任务，并定期抽查，以监控标注质量。为了减少标注工作量，我们在匹配过程中使用了折半插入策略。例如，一个待标注句首先与锚点2 的某个句子进行匹配，根据标注结果，该句子与锚点1 或者锚点3 的某个句子再次进行配对。重复这个过程直至确定该句子的难度级别。每个句子由至少五位标注员进行标注，即每个句子至少被标注五次。我们的标注周期为4周，每周会对标注员的工作进行检查。

3.2.2 数据集构建

标注周期结束后，我们收集了40192条数据，每条数据都被标注了5次，删除了标注时间小于15 秒（1%）的句子。我们使用多数投票原则决定单个句子的难度级别，3名以上标注员（包含3名）意见一致则确定最终难度标签。最终我们构建了一个基于语文教材的句子难度语料库。该语料库共包含37427 个汉语句子，每个句子被标注为1 至5 的某个难度级别，级别1 表示很简单，级别5 表示很难。表2 给出了每个难度级别上的示例句子。语料库中5 个难度级别的统计信息如表3 所示。表中除了包含每个级别中句子的数量信息，还包括了每个级别上句子的平均长度（以字为单位）和句子的平均难度值。句子的难度值的计算方式来自于江少敏(2009)，值越大则难度越高。

3.3 数据集比较

我们将于东等人(2020)构造的set1与之进行对比。set1来源为汉语语文教材，基于五点量

难度等级	例句
等级一	大家都觉得很不方便。
等级二	一只塘鹅闭紧嘴巴，她急急地走在小路上。
等级三	这里“你是”含有假定语气，也带“你不是”一点讥刺的意味。
等级四	克莱谛不时用眼睛瞟我，从他的眼里表示出来的不是愤怒，而是悲哀。
等级五	根据英国南部物候的一种长期记录，拿1741到1750年10年平均的春初7种乔木油青和开花日期相比较，可以看出后者比前者早9天。

表 2: 句子难度标注语料库示例

难度等级	数量	比例	平均句长	Jiang(2009)
等级一	3158	0.08	8.08	112.84
等级二	9235	0.25	16.57	220.65
等级三	14627	0.39	28.02	353.37
等级四	7371	0.20	42.83	530.36
等级五	2856	0.08	65.40	790.42
平均值	–	–	29.29	272.128

表 3: 标注数据集详情统计

表，1表示很简单，5表示很难，通过专家标注获得锚点句，然后通过众包的方式进行大规模标注，标注的过程中采用目标句与锚点句对比的方式进行。每个句子被标注三次，通过投票原则确定句子的难度等级。最后经过数据处理得到的语料库包含18411个汉语句子。set1与set2是两个既有相同之处也存在差异的句子可读性数据集。其中的相同点包括：句子数据均来源于权威的苏教版，北师大版和人教版的汉语语文教材，在数据标注的过程中首先基于五点量表，通过专家标注获得锚点数据集，然后通过目标句与锚点句的对比来得到目标句的难度等级。标注过程中的数据处理方式为投票原则，评判标准为肯德尔系数和方差分析。

不同点之一则在于set1是通过众包的方式进行标注，而set2是通过招募固定的优秀标注员进行标注，每个人由于受教育水平和文化背景的差异，对于难度的评判标准是不一致的，那么众包标注就意味着在数据集构建过程中的评判标准差异性更大，在数据处理的时候会引入更多的噪声，从而降低数据集的质量，那么对比而言，固定的标注员会使整个数据集的评判标注趋于统一，会提高数据集的质量；不同点之二在于构建set2的锚点句的数量为195，而构建set1的锚点数据集包含62条锚点句，其中锚点一的数据量为33，锚点二的数据量为16，锚点三的数据量为10，锚点四的数据量为3。可以发现锚点句的总量相对较少，且各锚点句数量的比例相差较大，不同等级的句子在字数，句式和结构等方面都存在很大差异，若可对比的锚点句的数量过少，则可作为评判指导的依据就少，这对于整个数据集的质量会产生一定的负面影响；不同点之三在于set1中的数据内容与set2中的数据完全不重合，set1中每条数据被标注3次，set2中每条数据被标注5次，标注次数越多则产生偏差的概率越低，数据质量越高。在之后的实验中，我们分别基于这两个数据集进行实验。

4 特征及模型

4.1 特征选择

可读性特征体系的设计参考了吴思远等(2020)的特征框架，该研究把评估文本可读性的指标划分为四个层面，分别是汉字、词汇、句法和篇章结构。于东等(2020)从汉字、词汇和句法三个层面实现句子语言特征的量化，达到了较好的分类结果。

汉字是汉语的书写符号，汉字的识别难度影响句子的阅读难度。汉字层面的语言特征是从字形复杂度、汉字熟悉度和汉字多样性三个角度进行量化，共22个指标，如汉字笔画数、字频等。词是语言中最基本的造句单位，词汇复杂性在句子理解中起着关键作用。影响词汇难度的特征主要包括词长、词汇熟悉度、词汇多样性和词汇语义难度四个维度，共25个指标，如词频、词长等。句法结构层面共包括3个维度的句法特征：句子表层的复杂度、词性复杂度、句法结构复杂度，共计25个指标。

在深度学习特征向量融合外部语言难度特征的实验部分，本文采用的外部语言难度特征即为汉字、词汇和句法层面的特征以及三个层面的组合特征。深度学习特征向量的抽取则根据模型的不同而不同，对于双向循环神经网络，则抽取模型最后一层的第一个神经元和最后一个神经元输出的特征向量组合。对于卷积神经网络，则抽取最后一个卷积层经过不同卷积核卷积之后输出的特征向量的组合，对于基于transformer的神经网络模型Bert，则使用肖涵博士开发的bert-as-service默认抽取倒数第二个transformer层的输出向量。

4.2 模型介绍

正如前文中提到，近年来文本分类任务的趋势表明，采用自动特征构建的深度学习方法占主导地位。Matrinc等(2019)在首次将深度学习方法用于英文可读性研究，为了确定不同模型在可读性研究中的性能和不足，评估了大量模型的复杂性。在此之前的可读性研究依赖人工构造特征和机器学习分类器(Vajjala and Lucic, 2018; Xia Kochmar Briscoe, 2016)。在汉语可读性研究中，即使是最新的中文可读性分类方法也依赖于人工构造的特征和传统的机器学习分类器(Wu Siyuan et al, 2020; Yu Dong et al, 2020)。在这一部分中，我们将重点介绍三大特征提取器RNN，CNN和Transformer，以及语言特征融入实验的框架流程，如图1所示。

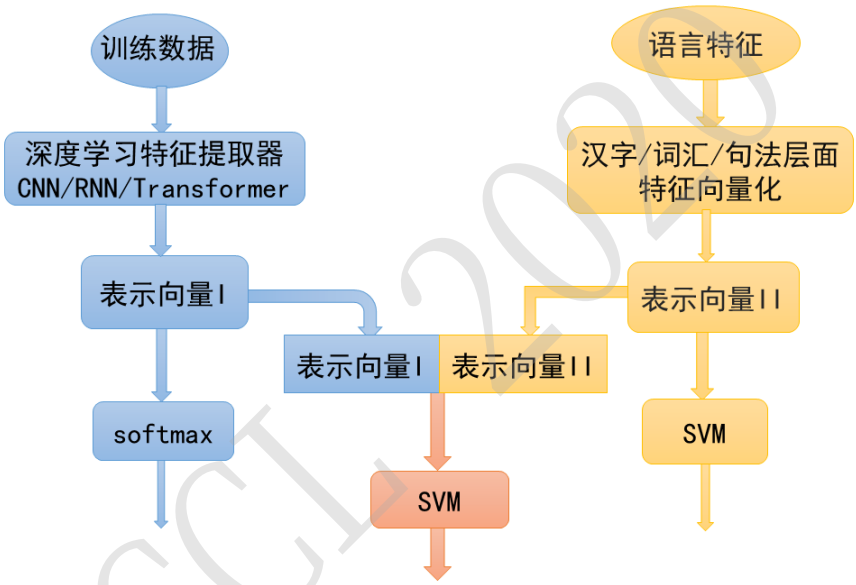


图 1: 语言特征融入流程图

- RNN：在一些自然语言处理任务中，当对序列进行处理时，我们一般会采用循环神经网络RNN，尤其是它的一些变种，如BiLSTM(Zhou et al, 2016)，GRU(Cho et al, 2014)等。循环神经网络善于捕捉更长的序列信息。LSTM因在文本分类任务上的效果非常好而受到重视。LSTM在每个时间步上的输入有两部分信息，一部分是前一个时间步的保留信息，一部分是当前时间步对应的原始信息，由此，LSTM可以在最后一个时间步获取到整个序列的信息，并且丢弃掉模型认为没有用的信息。句子作为序列，其长度是相对篇章来说较短的长度，所以LSTM模型可以很好地胜任句子的特征抽取任务。在实验中，我们采用基于双向LSTM的textRNN模型(Liu et al, 2016)，在该模型上，则抽取模型最后一层的第一个神经元和最后一个神经元输出的特征向量组合。

- CNN：卷积神经网络因其在句子分类任务上的突出表现而被选中(Kim, 2014)。本文使用的CNN模型基于Kim(2014)描述的textCNN模型。将卷积神经网络CNN应用到文本分类任务，利用多个不同尺寸的卷积核进行一维卷积来提取句子中的关键信息，每次能处理不同尺寸长度个词的完整的词向量，从上往下依次滑动卷积，这个过程中的输出就成了我们需要的特征向量，类似于多窗口大小的ngram，可以能够更好地捕捉局部相关性。CNN的并行计算能力非常强，可以快速实现特征提取。在textCNN模型上，我们抽取最后一个卷积池化层后输出的特征向量组合。

- Transformer：Transformer(Ashish Vaswani et al, 2017)是使用自注意力的Encoder-Decoder模型，其在包括可读性评估在内的众多自然语言处理任务上取得了最新的结果(Matrinc et al, 2019)。Transformer利用注意力机制，使得模型在构造输出向量时能够注意到输入的特定部分。尽管它们被表示为序列到序列模型，但是可以通过在网络的末端放置一个额外的线性层并训练该层以产生所需的输出来修改它们以完成各种NLP任务。这种方法在与预训练模型相结合时通常会获得最好的结果。在本文中，我们使用了基于Transformer的Bert中文模型(Devlin et al, 2018)，该模型是在图书语料库（800M words）(Zhu et al, 2015)和中文维基百科上预先训练的，然后在特定的可读性语料库上对模型进行微调。预训练的Bert模型来源于Huggingface的Transformer库(Thomas Wolf et al, 2019)，由12个隐藏层组成，每个隐藏层的大小为768和12个自注意力头。Transformer模型突破了RNN 模型不能并行计算的限制。相比CNN模型，计算两个位置之间的关联所需的操作次数不随距离增长。自注意力可以产生更具可解释性的模型。我们可以从模型中检查注意力分布。各个注意力头可以学会执行不同的任务。在语言特征融入的实验中，本文使用肖涵博士开发的bert-as-service[0]默认抽取倒数第二个transformer层的输出向量。

5 实验设计与结果分析

5.1 在机器学习方法上验证set2改进思路的有效性

为了验证本文提出的数据集构建改进方法的有效性，我们在set1与set2上对比了支持向量机（Support Vector Machine, SVM）和逻辑回归（Logistic Regression, LogR）两种模型的表现，我们以于东等(2020)基于tf-idf的词袋向量作为输入构建的模型作为基线模型，然后分别把汉字，词汇，句法层面的语言特征以及三个层面的组合特征作为句子的向量表示，构建特征模型。在实验过程中训练集与测试集的比例为8:2，采用五折交叉验证，评价指标为准确率。我们使用Python语言，在scikit-learn库(Pedregosa et al, 2011)中实现了模型。

Models	set1		set2		set1+set2	
	SVM	LogR	SVM	LogR	SVM	LogR
汉字	**0.6303**	**0.6308**	0.7779	**0.7740**	0.6928	**0.6901**
词汇	**0.6303**	0.6212	**0.7819**	0.7544	**0.6947**	0.6777
句法	0.6242	0.6242	0.7662	0.7461	0.6779	0.6552
all	0.6179	0.6301	0.7817	0.7597	0.6911	0.6895
tf-idf	0.4720	0.4549	0.5213	0.4740	0.4935	0.4772

表 4: 两个数据集在机器学习上的表现

我们对比分析了SVM和LogR两种不同分类模型在set1与set2上的表现，实验结果如表4所示.相对于基线模型，各个层面的语言难度特征都表现出了较好的效果，验证了语言难度特征的使用可以提升模型的效果，基于单一层面的语言特征的模型甚至比基于字词句组合特征的模型的效果更好，说明特征的使用并不是越多越好，汉字层面和词汇层面的特征的效度要比单一句法层面和三个层面的组合特征的效度更好(Yu Dong et al, 2020)。同时验证了机器学习方法在set2上的有效性。可以看出set1的整体效果在0.6179到0.6308之间，set2的整体效果在0.7461到0.7819之间，set1的最优结果是在LogR上以汉字层面特征作为特征向量的模型，准确率达到0.6308，set2的最优结果是在SVM上以词汇层面特征作为输入的模型，准确率为0.7819。set2的整体效果高出set1约0.15，说明set2数据集更加优质，噪点更低，证明了本文提出的数据集改进方法的有效性。set2在可读性评估上的效果更好，以set1作为可读性评估数据集具有更高的挑战性。

为了进一步对比两个数据集，以set1作为训练集，set2作为测试集进行实验，混淆矩阵的结果显示set2会更多地被分到比原等级更低的等级，如图2 所示。以set2作为训练集，set1作为测试集，混淆矩阵的结果显示set1会更多地被分到比原等级更高的等级，如图3所示。由此可以得出，set1与set2两个数据集的难度中心点不一致，set1的难度中心点更高，set2的难度中心点较set1偏低，set1的整体难度比set2高。这大约是因为固定的优秀标注员都有着相对较高

[0]https://github.com/hanxiao/bert-as-service

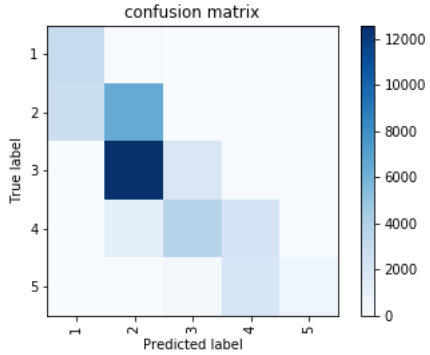


图 2: train set1,test set2.

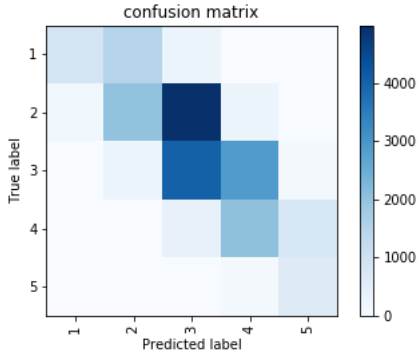


图 3: train set2,test set1.

的语言水平，对难度的感知相对较低，得到的句子难度等级更低，而众包标注过程中，参与标注的人语言水平门槛较低，使得参与标注的所有人的语言水平差异较大，低语言水平的标注员对难度的感知会更高，得到的句子难度等级更高，所以会使得set1的整体难度会比set2的高。合并set1与set2，随机混合数据，以8:2的比例切分训练集与测试集，以各个层面的特征作为输入。结果如表4所示，最优结果为0.6947。对比以上set1最高准确率0.6308，set2最高准确率0.7819，合并数据集得到的是一个折中的实验效果。通常来说，合并数据集之后，数据量更大，会得到更高的结果，但是这里合并数据集之后，得到的结果是以set1与set2分别进行实验的结果的折中效果，并且两个数据集的难度中心点不一致，所以本文认为，set1与set2是两个独立的数据集，合并set1与set2会得到一个更具挑战性的可读性数据集。

5.2 深度学习模型在汉语可读性评估上的有效性验证

深度学习方法在汉语可读性研究中的应用是可读性研究的更进一步，通过深度学习模型征提取器来自动学习获取特征,可以有效避免大规模的人工特征抽取。对于基于卷积神经网络和基于循环神经网络的模型，利用预训练的百度百科静态词向量sgns.baidubaike.bigram-char(Shen Li et al, 2018)，词嵌入维度为128，对于基于transformer的模型则在Bert上进行分类任务的微调。训练集，验证集，测试集的占比为8:1:1，以两个数据集在机器学习模型上的最优结果作为基线，评价指标为准确率和F1值。整个模型基于pytorch深度学习框架(Paszke et al, 2019)。

Models—Datasets	set1		set2	
	ACC	F1	ACC	F1
SVM	0.6303	0.6315	0.7819	0.7784
LogR	0.6308	0.6298	0.7740	0.7756
TextCNN	0.6020	0.6018	0.7199	0.7187
TextRNN	0.6071	0.6051	0.7293	0.7304
TextCNN(Bert词向量)	0.6324	0.6313	0.7752	0.7745
TextCNN(Bert字向量)	**0.6357**	**0.6345**	0.7773	0.7782
TextRNN(Bert词向量)	0.6326	0.6327	0.7830	0.7828
TextRNN(Bert字向量)	0.6345	0.6336	0.7850	0.7851
Bert	0.6209	0.6316	**0.7869**	**0.7943**

表 5: 两个数据集在深度学习上的表现

我们以SVM和LogR的实验结果作为基线，对比分析了TextCNN模型，TextRNN模型以及基于Transformer的Bert模型在句子级汉语可读性评估任务上的表现，实验结果如表5所示。实验结果证明了深度学习方法在汉语可读性评估任务上的有效性。从以上结果可以看出，以百度百科词向量作为辅助的TextCNN和TextRNN模型的效果不及机器学习模型的效果，这在set2上表现得更加突出。以Bert-as-service生成的的预训练Bert字向量和Bert词向量作为TextCNN与TextRNN的输入的模型，则在之前实验的基础上表现出了显著的提升，整体

效果比机器学习模型的效果更好，说明普通的静态词向量对这两个模型的难度特征的捕获没有起到很好的辅助作用。以普通静态词向量作为输入的TextCNN和TextRNN的难度表征能力不如人工抽取的语言特征的难度表征能力强。同时对比以Bert向量作为输入的实验结果，可以发现以Bert字特征向量为输入的模型具有更优的性能，说明相比Bert词向量，Bert字向量在该数据集上能更大程度上地表征难度信息。同时，在Bert模型上进行微调的实验结果在set1和set2上的准确率分别为0.6209和0.7869，在set2上达到了最好的结果。说明Bert预训练语言模型自动捕获的特征向量在很大程度上代表了难度信息(Tovly Deutsch et al, 2020)。对比TextRNN和TextCNN的所有实验结果可以发现TextRNN的效果总是比TextCNN模型的效果更好，说明在汉语句子级可读性评估任务上TextRNN的难度表征能力比TextCNN更强。

5.3 探究语言特征能否提升深度学习模型的整体性能

将人工抽取的语言特征应用于机器学习模型，在可读性难度判别任务上达到了不错的效果。为了探究人工抽取的语言难度特征能否提升深度学习模型的整体性能，我们以深度学习模型作为特征提取器，向深度学习特征向量中融入不同层面的语言难度特征进行实验，以期这种组合的特征向量可以更好地表征难度。本实验中的特征提取器分别是TextCNN模型、TextRNN模型和基于Transformer的Bert预训练语言模型，TextCNN和TextRNN不使用中文静态词向量。语言难度特征采用吴思远(2020)的可读性特征体系，在实验中分别加入各个层面的语言难度特征进行实验。

Models—Datasets	set1		set2	
	ACC	F1	ACC	F1
TextCNN	0.6021	0.6018	0.7087	0.7027
TextCNN+汉字层面	**0.6135**	0.6119	**0.7592**	**0.7564**
TextCNN+词汇层面	0.6134	**0.6148**	0.7541	0.7542
TextCNN+句法层面	0.6128	0.6114	0.7347	0.7282
TextCNN+三个层面	0.6133	0.6123	0.7501	0.7479
TextRNN	0.6027	0.6013	0.7183	0.7084
TextRNN+汉字层面	0.6142	0.6147	**0.7643**	**0.7622**
TextRNN+词汇层面	**0.6168**	**0.6202**	0.7579	0.7615
TextRNN+句法层面	0.6086	0.6056	0.7303	0.7337
TextRNN+三个层面	0.6125	0.6144	0.7499	0.7528
Bert	0.6209	0.6226	0.7812	0.7724
Bert+汉字层面	0.6292	0.6240	0.7819	0.7830
Bert+词汇层面	0.6234	0.6212	0.7814	0.7789
Bert+句法层面	0.6341	0.6328	0.7848	0.7851
Bert+三个层面	**0.6390**	**0.6361**	**0.7874**	**0.7919**

表 6: 深度学习融合外部语言特征的表现

以三种深度学习特征提取器抽取的特征向量单独作为输入的模型作为基线，通过向TextCNN，TextRNN和Bert的特征向量中融入不同层面的语言特征，可以发现语言特征能不同程度地提高模型的效果(Tovly Deutsch et al, 2020)，实验结果如表6所示。横向对比，在set1上的提升不显著，在set2上的提升则更加显著。纵向对比，在Bert上的提升不显著，在TextCNN和TextRNN上的提升则更加显著。在TextCNN和TextRNN上，融入汉字层面的特征和融入词汇层面的特征得到的提升更多，而在融入句法特征的模型上提升更少，说明TextCNN和TextRNN模型捕获的难度特征更偏向于类似句法层面的特征，对于整句信息的保留能力更强，则与汉字和词汇层面特征的互补性更强。在Bert的一些列组合特征模型中，在融合句法层面特征的模型和融合所有层面特征的模型上提升更多，说明Bert模型自动捕获的难度特征类型更偏向于汉字和词汇层面的细粒度特征，所以与句法层面的特征的互补性更强。在整个实验中，TextCNN和TextRNN的基线效果比Bert的基线效果相差许多，其融入特征的最优模型尚且没有达到Bert的基线效果，说明Transformer作为特征提取器，其难度特征捕获能力在各个层面都优于CNN和RNN特征提取器。

6 总结

在本文中，我们首先提出了改进语料库构建的方法，基于改进的方法思路，我们构建了一个规模更大、噪点更低、质量更高的句子级可读性语料库，该数语料库包含37247条数据。通过在机器学习模型中加入汉字层面、词汇层面、句法层面以及三个层面的组合语言特征来探究在set2上的表现，并且与set1的结果进行对比。实验结果显示，set2的准确率比set1的准确率高出约0.15，验证了本文中改进方法的有效性，以及该数据集的有效性。说明扩充锚点句对提高数据集质量有直接影响，固定标注人员可以保证难度衡量标准的一致性和稳定性，标注次数越多，难度等级偏差越小。

将深度学习方法应用于汉语可读性评估，验证了深度学习方法在该任务上的有效性，并且得到比机器学习略胜一筹的效果，说明深度学习自动捕获的特征在很大程度上能够代表难度信息。在这两个非平行语义的数据集中，效果最好的模型都与Bert相关，说明非平行语义的可读性评估与语义不可分割。TextRNN的难度特征捕获能力比TextCNN更胜一筹。深度学习模型自动捕获特征的能力可以有效减少人工抽取语言特征的成本，促进可读性研究。我们探讨了语言难度特征的融入能否提升深度学习模型的难度表征能力，深度学习特征与语言特征的互补性关系。实验结果表明，在汉语可读性评估中，语言难度特征可以不同程度地提升深度学习模型的表征能力。TextCNN和TextRNN模型捕获的特征与汉字和词汇层面的语言特征互补性更强，Bert预训练语言模型捕获的特征与句法层面的语言特征互补性更强。

总的来说，深度学习方法在可读性评估上的应用是一条必由之路，语言特征对深度学习模型的难度表征能力的提升不显著，那么我们之后的工作似乎更应该关注到深度学习特征在多大程度上代表了难度信息，深度学习模型如何能捕获到更多区别于语言特征的难度信息。未来，我们的数据集会进行开放，便于更多学者的研究。同时，我们的研究也将在更具挑战性的两个数据集的合并集上进行。

参考文献

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, and Illia Polosukhin. 2017 *Attention is all you need.* In Advances in neural information processing systems 30, pages 5998–6008. Curran Associates, Inc.

Brunato,De Mattei,Dell'orletta,Iavarone,Venturi. 2018 *Is this Sentence Difficult? Do you Agree?* Conference on Empirical Methods in Natural Language Processing. pages：2690-2699.

Cho, K. , Van Merrienboer, B. , Gulcehre, C. , Bahdanau, D. , Bougares, Schwenk, H. 2014 *Learning phrase representations using rnn encoder-decoder for statistical machine translation.* Computer Science.

Collins-Thompson , Kevyn. 2014 *Computational assessment of text readability: A survey of current and future research.*, ITL - International Journal of Applied Linguistics，165(2)：97-135.

Collobert, Ronan, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011 *Natural language processing (almost) from scratch.* Journal of Machine Learning Research, 12(Aug):2493–2537.

Crossley, Scott A, Stephen Skalicky, Mihai Dascalu, Danielle S McNamara, and Kristopher Kyle. 2017 *Predicting text comprehension, processing, and familiarity in adult readers: New approaches to readability formulas.* Discourse Processes, 54(5-6):340–359.

Dale, Edgar and Jeanne S Chall. 1948 *A formula for predicting readability: Instructions.* Educational research bulletin,pages 37–54.

Davison, Alice and Robert N Kantor. 1982 *On the failure of readability formulas to define readable texts: A case study from adaptations.* Reading research quarterly,pages 187–209.

Dell'Orletta F,Montemagni S,Venturi G. 2011 *Read-it: Assessing readability of italian texts with a view to text simplification.* Proceedings of the second workshop on speech and language processing for assistive technologies. Association for Computational Linguistics,2011：73-83.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee,and Kristina Toutanova. 2018 *Bert:Pre-training of deep bidirectional transformers for language understanding.* arXiv preprint arXiv:1810.04805.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher,Matthieu Perrot, and Edouard Duchesnay. 2011 *Scikit-learn: Machine Learning in Python.* Journal of Machine Learning Research, 12:2825-2830.

Feng L, Huenerfauth M. 2009 *Cognitively motivated features for readability assessment.* Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics,2009:229-237.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016 *Deep Learning.* MIT Press.

Gunning, Robert. 1952 *The technique of clear writing.* McGraw-Hill, New York.

Karpov N,Baranova J,Vitugin F. 2014 *Single-sentence readability prediction in Russian.* International Conference on Analysis of Images,Social Networks and Texts. Springer,Cham,2014:91-100.

Kate R J,Luo X,Patwardhan S,et al. 2010 *Learning to Predict Readability using Diverse Linguistic Features.* Coling 2010 - 23rd International Conference on Computational Linguistics, Proceedings of the Conference. 546-556.

Kim Y . 2014 *Convolutional Neural Networks for Sentence Classification.* Eprint Arxiv, 2014.

Kincaid J P， Fishburn R P， Chisson B S. 1975 *Derivation of new readability formulas for navy enlisted personnel.* Adult Basic Education，1975：49.

Laughlin G H M. 1969 *SMOG Grading-a New Readability Formula.* Journal of Reading,12(8)：639-646.

Liu P , Qiu X , Huang X . 2016 *Recurrent Neural Network for Text Classification with Multi-Task Learning.*

Luo S,Callan J. 2001 *A statistical model for scientific readability.* Tenth International Conference on Information and Knowledge Man-agement. ACM，2001：574-576.

Matej Martinc, Senja Pollak, Marko Robnik-Šikonja. 2019 *Supervised and Unsupervised Nerual Approches to text readability.* Computational Linguistic journal.

Paszke A, Gross S , Massa F . 2019 *PyTorch: An Imperative Style, High-Performance Deep Learning Library.*

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi , Bingchen Li, Hongwei Hao, Bo Xu. 2016 *Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification* Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics,pages:229-237.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao. 2016 *Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification.* Meeting of the Association for Computational Linguistics.

Petersen, Sarah E and Mari Ostendorf. 2009 *A machine learning approach to reading level assessment.* Computer speech and language, 23(1):89–106.

Pilan I，Vajjala S，Volodina E. 2016 *A readable read：Automatic assessment of language learning materials based on linguistic complexity.*, arXiv preprint arXiv：1603.08868.

Schumacher E，Eskenazi M，Frishkoff G，et al. 2016 *Predicting the Relative Difficulty of Single Sentences With and Without Surrounding Context.* Conference on Empirical Methods in Natural Language Processing. pages：1871-1881.

Schwarm, Sarah E and Mari Ostendorf. 2005 *Reading level assessment using support vector machines and statistical language models.* In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pages 523–530. Association for Computational Linguistics.

Shen Li, Zhe Zhao, Renfen Hu, Wensi Li, Tao Liu, Xiaoyong Du. 2018 *Analogical Reasoning on Chinese Morphological and Semantic Relations.* Meeting of the Association for Computational Linguistics.

Sung Y T,Chen J L,Cha J H,et al. 2015 *Constructing and validating readability models：the method of integrating multilevel linguistic features with machine learning[J].* Behavior research methods,47(2)：340-354.

Tanaka-Ishii K, Tezuka S, Terada H. 2010 *Sorting texts by readability.* Computational Linguistics, 36(2):203-227.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Morgan Funtowicz, and Jamie Brew. 2019 *HuggingFace's transformers: state-of-the-art natural language processing.* Technical report.

Tovly Deutsch ,Masoud Jasbi, Stuart Shieber. 2020 *Linguistic Features for Readability Assessment* arXiv preprint arXiv:2006.00377.

Vajjala, Sowmya and Detmar Meurers. 2012 *On improving the accuracy of readability classification using insights from second language acquisition.* In Proceedings of the seventh workshop on building educational applications using NLP, pages 163–173. Association for Computational Linguistics.

Vajjala, Sowmya and Ivana Lucic. 2018 *Onestopenglish corpus: A new corpus for automatic readability assessment and text simplification.* In Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications,pages 297–304. Association for Computational Linguistics.

Xia, Kochmar, Briscoe. 2016 *Text Readability Assessment for Second Language Learners.* Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications.

Xu, Wei, Chris Callison-Burch, and Courtney Napoles. 2015 *Problems in current text simplification research: New data can help.* Transactions of the Association of Computational Linguistics, 3(1):283–297.

Yang, Zichao, Diyi Yang, Chris Dyer,Xiao dong He, Alex Smola, and Eduard Hovy. 2016 *Hierarchical attention networks for document classification.* In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1480–1489.

Yoon Kim. 2014 *Convolutional Neural Networks for Sentence Classification.* In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Yukun Zhu, Ryan Kiros, Richard Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015 *Aligning Books and Movies:Towards Story-like Visual Explanations by Watching Movies and Reading Books.* Computing Research Repository, arXiv:1506.06724.

Zhang, Xiang, Junbo Zhao, and Yann LeCun. 2015 *Character-level convolutional networks for text classification.* In Advances in neural information processing systems, pages 649–657.

郭望皓. 2016 基于*CRITIC* 加权赋值的汉语句子难度测定. 语文学刊(教育版),2016(12)：10-12.

江少敏. 2009 句子难度度量研究. 厦门大学硕士学位论文.

庞成. 2016 汉语句子难易度影响因素分析. 语文学刊(教育版),2016(1)：18-19.

吴思远，蔡建永，于东. 文本可读性的自动分析研究综述. 中文信息学报，2018,32(12)：1-25.

吴思远，于东，江新. 2020 汉语文本可读性特征体系构建及其效度验证. 世界汉语教学，2020(1):81-97.

王蕾. 2008 可读性公式的内涵及研究范式——兼议对外汉语可读性公式的研究任务. 语言教学与研究，2008(6)：46-53.

于东,吴思远,耿朝阳,唐玉玲. 2020 基于众包标注的语文教材句子难易度评估研究. 中文信息学报34(2):16-26.

基于预训练语言模型的案件要素识别方法

刘海顺[1] **王雷**[2] **陈彦光**[1] **张书晨**[1] **孙媛媛**[1†] **林鸿飞**[1]
[1]大连理工大学计算机科学与技术学院，辽宁大连 116024
[2]辽宁省人民检察院第三检察部，辽宁沈阳 110033
dahai@mail.dlut.edu.cn, 18804002266@163.com
{cygariel, camael}@mail.dlut.edu.cn, syuan@dlut.edu.cn, hflin@dlut.edu.cn

摘要

案件要素识别指将案件描述中重要事实描述自动抽取出来，并根据领域专家设计的要素体系进行分类，是智慧司法领域的重要研究内容。基于传统神经网络的文本编码难以提取深层次特征，基于阈值的多标签分类难以捕获标签间依赖关系，因此本文提出了基于预训练语言模型的多标签文本分类模型。该模型采用以Layer-attentive策略进行特征融合的语言模型作为编码器，使用基于LSTM的序列生成模型作为解码器。在“CAIL2019”数据集上进行实验，该方法比基于循环神经网络的算法在F1值上最高可提升7.6%，在相同超参数设置下比基础语言模型（BERT）提升约3.2%。

关键词： 案件要素识别；多标签文本分类；智慧司法；语言模型

A Method for Case Factor Recognition Based on Pre-trained Language Models

Haishun Liu[1] **Lei Wang**[2] **Yanguang Chen**[1] **Shuchen Zhang**[1]
Yuanyuan Sun[1†] **Hongfei Lin**[1]
[1]School of Computer Science and Technology, Dalian University of Technology
[2]The Third Procuratorial Department, People's Procuratorate of Liaoning Province
dahai@mail.dlut.edu.cn, 18804002266@163.com
{cygariel, camael}@mail.dlut.edu.cn, syuan@dlut.edu.cn, hflin@dlut.edu.cn

Abstract

Case factor recognition is an important research content in the domain of legal intelligence. The purpose of this task is to automatically extract the important fact descriptions from the legal case descriptions and classify them based on the factor system designed by the domain experts. Text encoding based on traditional neural networks is difficult to extract deep-level features, and threshold based multi-label classification is difficult to capture the dependencies between labels. So that a multi-label text classification model based on pre-trained language models is proposed. The encoder is the language model fine-tuned with the strategy of Layer-attentive, and the decoder is LSTM based sequence generation model. Experimented on the CAIL2019 dataset, the method can improve the F1 score by up to 7.6% over the traditional neural network algorithm based on Recurrent Neural Network, and about 3.2% over the basic language model under the same hyperparameter settings.

Keywords: Case factor recognition , Multi-label text classification , Legal intelligence , Language model

[†]通信作者

1 引言

2018年，司法部印发《“十三五”全国司法行政信息化发展规划》，明确提出我国到2020年全面建成智能高效的司法行政信息化体系3.0版，将大数据、人工智能、云计算、物联网等技术与司法工作进行实际融合，实现公共法律服务的便捷普惠化，实现政务管理水平的高效透明。随着我国司法行政信息化的不断推进，智慧司法研究领域兴起并日趋火热。智慧司法包括法律阅读理解、案件要素识别、相似案例匹配和司法判决预测等任务，旨在赋予机器理解法律文本的能力，促进司法智能的发展。其中，案件要素识别的具体研究内容为，给定裁判文书中的相关段落，针对文书中每个句子进行判断，识别其中的关键案情要素。案件要素抽取的结果不仅可以为要素式裁判提供技术支持，还可以应用到案情摘要、可解释性的类案推送以及相关知识推荐等司法领域的实际业务需求中。

前人在司法智能领域的研究工作主要集中在司法判决预测 (Luo et al., 2017; Zhong et al., 2018; Hu et al., 2018)、相似案例匹配和命名实体识别测 (Wang, 2018; Xie, 2018)等方面，直接针对案件要素识别的研究还相对较少，但它们在技术上具有共通性。与通用领域的自然语言处理（NLP）任务 (Kim, 2014; Bahdanau et al., 2014; Yang et al., 2016)类似，当前研究者们在智慧司法领域采用的方法多是基于神经网络的结构。具体地，网络底层使用预训练的词向量进行词嵌入，中层采用卷积神经网络（Convolutional Neural Networks, CNN）或者循环神经网络（Recurrent Neural Network, RNN）提取特征，上层应用分类器进行分类或应用条件随机场（Conditional Random Field，CRF）进行序列标注。这种结构存在一定的缺点，一是使用的静态词向量无法处理不同语境下的一词多义问题(Peters et al., 2018)，二是有监督方法的本质致使模型性能受限于标注数据集的大小。

不同于一般的多分类问题，案件要素识别是多标签分类问题，即一个样本可能同时属于0到N个类别。经统计分析，不计算负例，每个样本平均包含2.7个标签，最多可达7个。而且多个类别之间往往具有关联性，如Figure 1所示，在离婚类案件中，若一个样本属于“限制行为能力抚养子女”类，那么该样本有较大概率同时属于“婚后有子女”类，在借贷类案件中，“有借贷证明”多和“有书面还款协议”一起出现。解决多标签分类问题的主流方法是将其处理为多个二分类问题(Boutell et al., 2004)，通过设定阈值判断样本是否属于每个类。这种方法明显忽略了标签之间的相关性，性能有限。

针对上述问题，本文专门就案件要素识别任务进行了研究，提出了基于预训练语言模型的案件要素多标签分类方法。预训练语言模型支持上下文有关的词嵌入，可以从庞大的无标注数据中学习丰富的语法、语义等特征表示，捕获更长距离的依赖。BERT(Devlin et al., 2019)是预训练语言模型的一个基础模型，于公布之初横扫了11项NLP任务。结合Yang (2018)的工作，本文将BERT系列语言模型作为案件要素识别整体模型的编码器，且提出了Layer-attentive的多层特征的融合策略，将长短期记忆网络（Long Short-Term Memory，LSTM）作为解码器，并对比了与基于阈值算法的多标签分类的性能差异。最后，在公开的CAIL2019“要素识别”数据集上验证了模型的性能。

裁判文书句子描述	案件要素类别
原告王某某诉称：x 年 x 月 x 日，原、被告在 x 市民政局协议离婚，离婚时约定，长子王某甲由被告抚养，长女王某乙由原告抚养。	婚后有子女； 限制行为能力子女抚养
原告规划院向本院提出诉讼请求：1、判令原告不支付被告解除劳动合同的赔偿金 58288 元；	经济性裁员
原告诉称，被告侯 x 在 x 年 x 月 x 日向原告出具书面个人借款承诺书，提出向 x 银行贷款 70000 元，请原告担保。	借款金额 x 万元； 有借贷证明； 贷款人系金融机构； 有书面还款承诺

Figure 1: 案件要素识别实例

2 相关工作

智慧司法研究由来已久。早在上世纪五、六十年代，研究者就开始通过数学统计的方法对司法案件进行定量分析(Kort, 1957; Ulmer, 1963)，随后在八、九十年代，研究者们探索了基于规则的专家系统(Shapira, 1990; Hassett, 1993)。随着机器学习技术的发展，司法判决预测作为智慧司法研究的主要任务而备受关注，基于支持向量机（Support Vector Machine, SVM）的预测模型被提出来，预测对象包括罪名、案件类别和裁判日期等(Aletras et al., 2016; Sulea et al., 2017)。近年来，由于司法数据的公开和深度学习的发展，我国在司法判决预测方面出现了许多瞩目的工作。Luo (2017)通过BiGUR（双向门控神经单元）建模判决书文档及法条信息以进行罪名预测，CAIL2018(Xiao et al., 2018)提出了第一个用于司法判决预测的大规模中文法律数据集，Zhong (2018)以CNN和LSTM为基础构建了同时预测罪名、法条和刑期的多任务学习模型，Hu (2018)通过引入司法属性研究了少数罪名的预测问题。案件要素识别是司法智能领域的新兴任务，现阶段主要被当做文本分类问题进行处理，在技术上与司法判决预测最接近。

作为案件要素识别核心技术的文本分类，近几年，主流方法逐步从词向量加神经网络向语言模型转变。2013年开始，Word2Vec(Mikolov et al., 2013)以网络结构简单、易于理解、使用方便等特征成为最流行的词向量训练工具之一。随后，Kim (2014)结合词向量提出了多维度并行的单层卷积神经网络，模型表现优于传统机器学习方法和早期神经网络方法。紧接着，RNN也被引入文本领域，其变体LSTM(Hochreiter and Schmidhuber , 1997)以能捕获长距离信息依赖、善于编码序列信息而得到大量应用，Yang (2018)提出了基于LSTM序列生成模型的多标签文本分类算法。而后注意力机制被广泛研究(Bahdanau et al., 2014; Yang et al., 2016)，Lin (2017)提出Self-attentive，通过二维矩阵对序列信息进行加权。2018年，谷歌的研究人员提出了基于自注意力机制对Transformer框架(Vaswani et al., 2017)，并以Transformer为核心组件开发出了性能强大的语言模型BERT。

BERT的预训练及微调方法被不断进行改进 (Yang et al., 2019; Cui et al., 2019; Liu et al., 2019)。Qiao (2019)提出的BERT(MUL-Int)将每一层的[CLS]位置的编码进行加权求和，进而计算索引问题和答案文档之间的相似度。Sun (2019)基于BERT设计了更多的实验，不仅验证了每一层输出对分类结果的影响，还提出以简单平均的方式融合前四层或后四层输出。本文基于以上提到的文本分类模型进行了案件要素识别的相关实验和分析，对比了不同语言模型的性能差异，在Lin (2017)、Qiao (2019)和Sun (2019)等人工作的基础上提出了Layer-attentive特征融合策略。就多标签文本分类而言，本文使用LSTM序列生成模型，并对比了与阈值算法的性能差异。

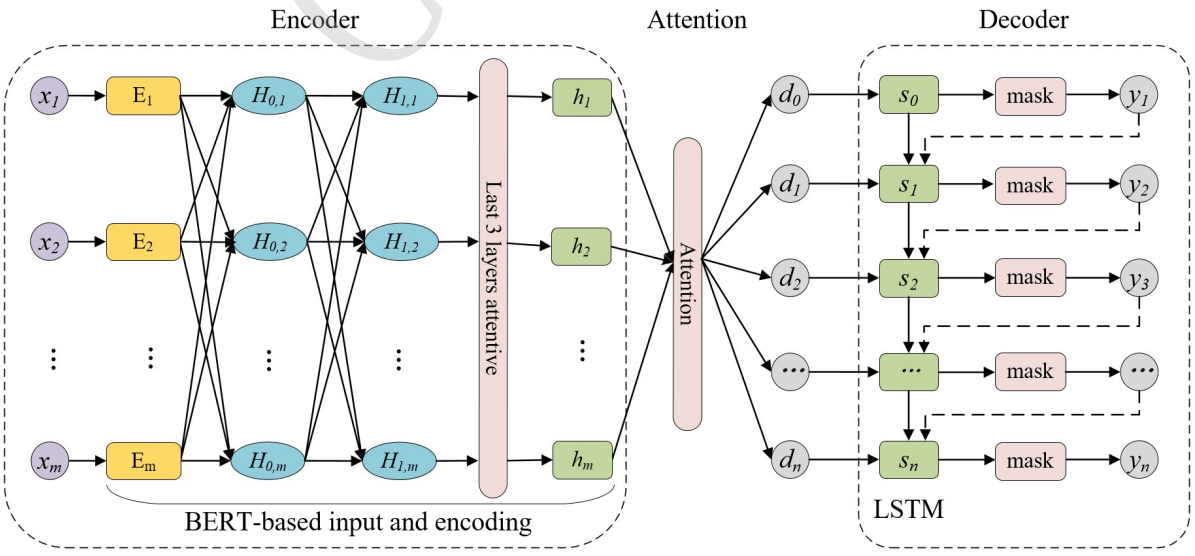


Figure 2: 基于预训练语言模型的案件要素识别模型

3 方法

本文将案件要素识别任务形式化描述为一个多标签分类任务，给定裁判文书中的一个句子序列$X = \{x_1, x_2, ..., x_m\}$，预测与$X$相对应的案件要素类别集合$\tilde{Y} \subseteq Y$。其中$m$是序列$X$的长度，$x_i$表示序列中的第$i$个词。$Y = \{y_1, y_2, ..., y_n\}$为要素标签集合，$n$为要素类别总数，因为一个样本可能同时归属于多个要素类别，所以$\tilde{Y}$是Y的子集。

本文所构建模型的结构如Figure 2所示，整体主要包括编码器（Encoder）和解码器（Decoder），中间通过注意力机制（Attention）操作进行交互。编码器部分以BERT为主体，将BERT最后三层的输出以Layer-attentive的方式进行加权融合得到输出H，对$H = [h_1, h_2, ..., h_m]$和$S = [s_1, s_2, ..., s_m]$进行Attention操作得到$D = [d_0, d_1, ..., d_n]$，将D通过基于LSTM的解码器得到标签预测集合$\tilde{Y}$。

3.1 BERT预训练语言模型

下面以BERT(Devlin et al., 2019)为例介绍BERT系列的预训练语言模型。BERT预训练语言模型的全称是基于Transformer的双向编码表示（Bidirectional Encoder Representations from Transformers, BERT）。其采用Transformer网络(Vaswani et al., 2017)作为模型基本结构，在大规模无监督语料上通过掩蔽语言模型和句对预测两个任务进行预训练（Pre-training），得到预训练的BERT模型。再以预训练模型为基础，在下游相关NLP任务上进行模型微调（Fine-tuning）。BERT模型的结构主要由三部分构成：输入层、编码层和任务层，其中输入层和编码层是通用的结构，对任何任务都适用。

BERT的输入层将每个词的词嵌入、位置嵌入和段嵌入相加得到每个词的输入表示$[E_1, E_2, ..., E_m]$。与原始Transformer不同的是，BERT模型的位置嵌入是可学习的参数，最大支持长度为512个位置。

对于编码层，base版本包含12层编码层，large版本包含24层编码层，每一层的输入都是基于上一层的输出，可抽象表示为：

$$H_i = \text{Transformer}(H_{i-1}), 0 < i < l \tag{1}$$

其中，$H_i \in R^{m \times d}$，m为序列长度，d为隐层维度。

在本任务中，任务层被Attention交互层和解码器替代。

3.2 基于预训练模型的编码器

一个神经网络的不同层可以捕获不同的语法和语义信息。因为BERT包含了l（12或24）个编码层，研究表明(Qiao et al., 2019; Sun et al., 2019)，选择BERT后3至4个编码层的输出进行特征融合，可以增强语言模型的特征表示。本文提出了Layer-attentive，以层次级别加权的方式对后三个编码层的输出进行融合，公式如下：

$$A_i = \text{softmax}(W_2 \text{tanh}(W_1 H_i^T)) \tag{2}$$

$$H = \text{SeLu}(A_{-1}H_{-1} + A_{-2}H_{-2} + A_{-3}H_{-3}) \tag{3}$$

其中，$W_1 \in R^{d \times d}$，$W_2 \in R^{d \times d}$，是两个权重矩阵，可以将向量的表示聚焦于不同层的不同元素。SeLU(Klambauer et al., 2017)是非线性激活函数。本文将以上特征融合方法命名为3Lattv。

为了证明以上方法的有效性，本文还设计了其他的特征融合方法。一是采用concat的方式对后三层的输出进行线性拼接：

$$H' = \text{SeLU}(W_c(H_{-1} \oplus H_{-2} \oplus H_{-3}) + b_c) \tag{4}$$

其中$W_c \in R^{d \times 3d}$，$\oplus$表示线性拼接。该方法被命名为3Lconcat。二是在上述两种方法中改后三层为后四层，相应的方法被命名为4Lattv和4Lconcat。

3.3 注意力交互

当模型预测不同的标签时，并非所有文本词都作出相同的贡献。Attention通过关注文本序列的不同部分并聚集那些信息丰富的词的隐层表示来产生上下文向量。特别地，注意力在时间步t上将权重α_{ti}分配给第i个单词，如下所示：

$$e_{ti} = v_a^T \text{tanh}(W_a s_t + U_a h_i + b_a) \tag{5}$$

$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{j=1}^{m} \exp(e_{tj})} \tag{6}$$

其中，W_a，U_a，v_a是权重参数，b_a是偏置项，s_t是解码器在时间步t的当前隐藏状态。在时间步t传递到解码器的最终上下文向量d_t的计算如下：

$$d_t = \sum_{i=1}^{m} \alpha_{ti} h_i \tag{7}$$

其中d_t的物理意义是预测第t个标签时的解码器的输入。

3.4 基于LSTM的解码器

本文使用LSTM(Hochreiter and Schmidhuber , 1997)作为多标签分类的解码器，解码器在时间步t的隐藏状态s_t的计算公式如下：

$$s_t = \text{LSTM}(s_{t-1}, y_{t-1}, c_{t-1}) \tag{8}$$

其中，y_{t-1}是时间步*t-1*在标签空间Y上的概率分布，其计算如下：

$$o_t = W_o \sigma(W_d s_t + U_d c_t + b_d) \tag{9}$$

$$y_t = \text{softmax}(o_t + I_t) \tag{10}$$

其中W_o，W_d，U_d是权重系数，$I_t \in R^Y$是用于防止解码器预测重复标签的掩码向量，即Figure 2中mask部分，σ是非线性激活函数。如果标签y_i在第*t-1*时间步被预测出来，则$I_t = -\infty$，否则$I_t = 0$。

最后，使用交叉熵损失函数进行训练：

$$J(\theta) = -\frac{1}{Nn} \sum_{i=1}^{N} \sum_{j=1}^{n} (y_{ij} \log p_{ij} + (1 - y_{ij}) \log(1 - p_{ij})) \tag{11}$$

其中N为样本个数，n为标签个数，y为实际标签，p为预测标签。

4 实验

4.1 数据集介绍

本文实验使用CAIL2019“要素识别”赛道提供的数据集[0]，该数据集来自“中国裁判文书网”公开的法律文书，由专家进行标注。数据的每一条由一个句子及其对应的要素标签组成，句子是从一篇裁判文书中的部分段落提取出来的，实例如Figure 1所示。本文将其按3:1:1的比例划分训练集、开发集和测试集，在测试集上评价模型性能。数据集涉及三类民事案件：劳动争议（Labor）、离婚纠纷（Divorce）和借贷纠纷（Loan），三类案件的数据各自分开，分别进行评价。每类案件各有20个要素类别，相应的类别样本数分布如Figure 3所示。可见数据集存在严重的数据分布不均衡的问题，每个案件的要素类别样本数从10_1级到10_3级不等。数据集的样本数据量统计及在样本的文本特点分析见Tanle 1。另发现平均60%以上的样本没有标签，即不是案件要素；一个样本最多可有7个标签，此种情况不足0.1%；具有1到3个标签的样本在三类案件中分别占约30%、25%、37%。

Table 1: 数据集的样本数量统计表

案件	案例数	每案例平均样本数	样本平均长度	样本数			
				训练集	开发集	测试集	合计
Labor	836	37.95	57.22	19038	6346	6346	31730
Divorce	1269	29.09	48.07	22152	7384	7384	36920
Loan	634	35.73	74.43	13615	4538	4538	22691

4.2 环境及参数设置

本文所有实验在如Table 2所示的环境下进行。对于BERT系列模型，均采用base-

[0]https://github.com/china-ai-law-challenge/CAIL2019

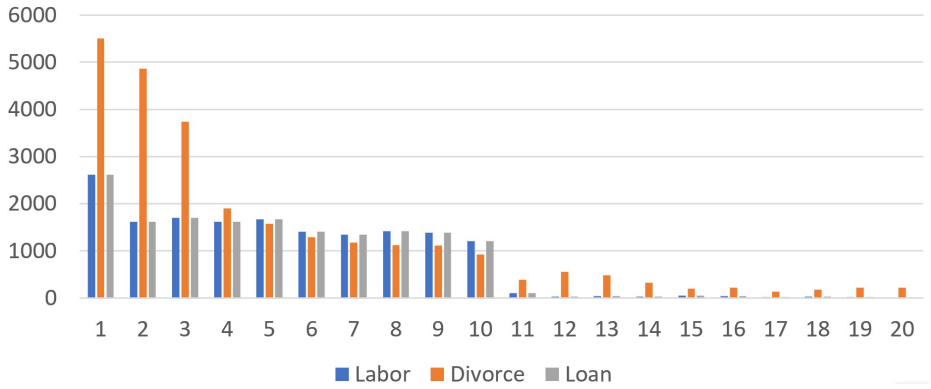


Figure 3: 各要素类别的数据量分布统计图

Table 2: 实验环境

环境名称	配置
操作系统	Ubuntu 16.04
CPU	Intel(R) Xeon(R) CPU E5-2678 v3 @ 2.50GHz
GPU	NVIDIA Tesla K80@11GB
Python	3.7.5
Pytroch	1.3.1
内存	128G

Chinese版本[1]进行微调，隐层维度d=768，序列长度m=512，编码层层数l=12，批处理大小batch_size=16，训练轮数epoch=3，学习率α=4e-5。对于BiLSTM模型，设置隐层维度hidden_size=256，序列长度m=256，学习率α=1e-3，批处理大小batch_size=64，训练轮数epoch=128，采用Word2Vec预训练的词向量的维度为300。

4.3 结果及分析

对于模型的表现，使用查准率(Precision, P)、查全率(Recall, R)和F1值作为衡量指标。具体使用宏平均查准率(Macro Precision, ma-P)、宏平均查全率(Macro Recall, ma-R)、宏平均F1值(Macro F1, ma-F)、微平均F1值(Micro F1, mi-F)、ma-F和mi-F的均值(Average F1, Ava)[0]。

4.3.1 编码器的作用

分别采用不同的编码器模型和解码器LSTM进行组合，在三个案件的数据上均进行实验。编码器模型列表如下：

BERT[1]：基础模型(Devlin et al., 2019)。

CNN-thre：Kim (2014)提出的卷积神经网络模型，底层使用预训练的词向量，使用多重一维卷积和最大池化提取特征。不使用解码器，输出层采用Algorithm 1所述方法。

BiLSTM：双向LSTM(Hochreiter and Schmidhuber , 1997)网络，底层使用预训练的词向量。

WWM[2]：基于Whole Word Masking训练样本生成策略训练的BERT(Cui et al., 2019)。

XLNet[3]：基于Transformer-XL(Dai et al., 2019)训练的最优自回归语言模型(Yang et al., 2019)。

RoBERTa[2]：采用多种技巧及更多数据训练的BERT(Liu et al., 2019)。

Table 3展示了在使用解码器LSTM的情况下，不同编码器模型在三类案件数据上的实验结果。比较CNN-thre、BiLSTM和BERT三个模型，BiLSTM优于CNN-thre，BERT优

[1]https://github.com/huggingface/transformers
[2]https://github.com/ymcui/Chinese-BERT-wwm
[3]https://github.com/ymcui/Chinese-XLNet

于BiLSTM，但该优势对语言模型而言提升并不特别明显。原因是一方面任务数据量达到万级，BiLSTM也能充分学习文本特征，另一方面训练BiLSTM所依据的词向量是根据数百万份裁判文书预训练的，Word2Vec在这里起到了很大的作用。为了详细比较BiLSTM和BERT在每个类别上的分类能力，Figure 4给出了BERT 和BiLSTM在Loan案件数据上每个要素类别的F1值。Figure 4表明，BERT对每个类别的分类能力均高于BiLSTM，在后10个类别，BERT的性能提升比较明显，结合Figure 3可知，Loan数据的后10个类别的样本数较前10个类别有数量级级别的差距，该结果也表明，以BERT为代表的语言模型处理小样本情况的能力较强。

纵向比较后四个模型，即四个BERT系列语言模型。BERT作为基础模型，性能较更先进的语言模型有一定的差距，XLNet和RoBERTa在该任务上具有最好的性能。RoBERTa比CNN-thre这一baseline模型绝对提升7.6%。另外，ma-F得分远低于mi-F得分，原因是数据分布极不均衡，每个类别的F1值相差很大，甚至有样本数量极少的类别的得分是0，这对ma-F影响较大，却对mi-F影响不明显。

Table 3: 不同编码器模型在三类案件数据上的实验结果

模型	Labor			Divorce			Loan		
	mi-F	ma-F	Ava	mi-F	ma-F	Ava	mi-F	ma-F	Ava
CNN-thre	75.03	50.26	62.65	80.05	66.34	73.20	72.08	47.03	59.56
BiLSTM	77.67	52.58	65.13	81.82	68.83	75.33	75.36	49.79	62.58
BERT	80.35	56.06	68.21	84.70	72.06	78.38	78.70	53.81	66.26
WWM	80.54	56.58	68.56	82.80	72.22	77.51	78.60	53.68	66.14
XLNet	81.24	58.48	69.86	**85.04**	**76.61**	**80.81**	79.33	55.77	67.55
RoBERTa	**81.29**	**58.66**	**69.96**	84.21	74.39	79.30	**79.82**	**57.06**	**68.44**

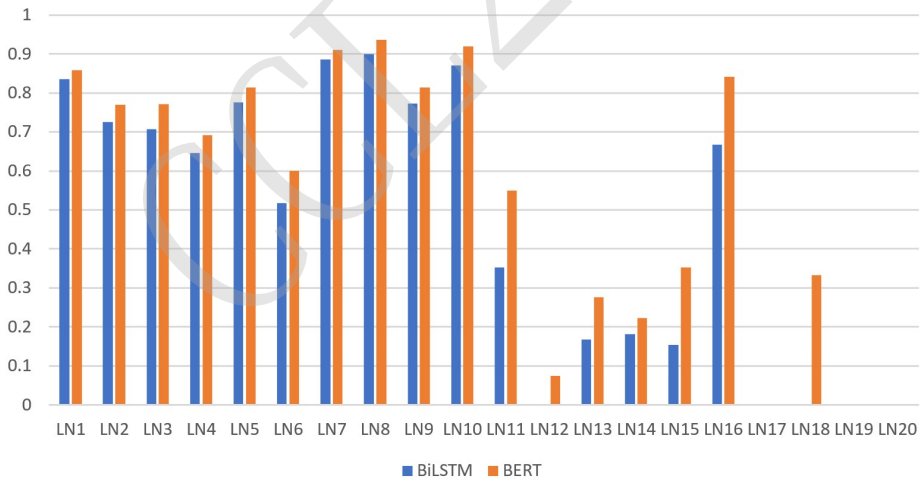


Figure 4: Loan数据上两个模型的类别F1值对比

4.3.2 解码器的作用

BERT-thre：该方法为只使用基于BERT的编码器，不使用Attention交互和解码器，相应的任务层换为softmax分类器，最后使用阈值设定函数对模型输出的概率值进行取舍，从而预测类别，标签概率计算公式如下：

$$p = \sigma(W_p \text{Pooler}(\text{H}) + b_p) \tag{12}$$

其中$W_p \in R^{d \times d}$，σ为sigmoid激活函数，Pooler是BERT对隐层输出进行pooling操作的函数(Devlin et al., 2019)。p中每个维度的数值对应每个类别的可能概率值，概率值介于[0,1]之间，仍使用二进制交叉熵损失函数进行训练。

Algorithm 1 类别阈值选择算法(Threshold selecting, thre)

Input: 在开发集上，样本属于第i类标签的概率p_i，样本在第i类的真实标签y_i。
Output: 第i类标签的阈值t_i。

```
t=arr[100], f=arr[100], s=arr[90], t[0]=0;
for j in 100 do
    t[j] = t[j-1] + 0.01;
    if p > t[j] then
        判断当前阈值t[j]下样本类别â_i = 1;
    else
        â_i = 0;
    计算当前阈值t[j]下的类别得分f[j] = F(ŷ_i, y_i);
    j递增1;
for k in 90 do
    保存每个区间下的得分均值s[k] = (∑_{l=0}^{9} f[k + l])/10;
    k递增1;
找到使得分最大的阈值区间z = argmax(s[z])，计算该区间的中值t_i = t[z + 5];
return t_i
```

本文Algorithm 1所示算法为每个类别设定阈值。

由于多标签分类的特殊性，具体的P、R值只能通过两者的宏平均或者微平均来体现，Tabel 4通过比较模型在三类案件数据上的ma-P、ma-R和ma-F，具体验证解码器对P值和R值带来的提升。

Table 4: 解码器与阈值算法的实验结果对比

模型	Labor			Divorce			Loan		
	ma-P	ma-R	ma-F	ma-P	ma-R	ma-F	ma-P	ma-R	ma-F
BERT-thre	57.41	53.10	55.47	71.87	72.63	71.95	54.15	51.52	53.14
BERT-LSTM	57.96	53.79	56.06	71.74	72.90	72.06	54.62	52.38	53.81
提升	+0.55	+0.69	+0.59	-0.13	+0.27	+0.11	+0.47	+0.86	+0.67
RoBERTa-thre	60.22	56.19	58.43	75.83	74.62	74.46	57.46	55.30	56.67
RoBERTa-LSTM	69.43	56.46	58.66	75.82	74.50	74.39	57.81	55.74	57.06
提升	+0.21	+0.27	+0.23	-0.01	-0.12	-0.07	+0.35	+0.44	+0.39

在Table 4中，同一案件下编码器较thre策略的主要提升体现在R值（召回率）上，尤其对Loan案件最为明显，经分析如Loan中两个要素类别“贷款人系金融机构款”和“有借贷证明”之间的相关性达到了0.729，其他类别也具有明显的相关性。基于LSTM的解码器正因为捕获了这种相关性，才在预测出来一个标签的情况下能连带着把与之相关的标签也预测出来。但是，准确率增益差说明这种解码器也存在不足，标签预测过程中会出现一定的错误累积，前一个标签预测错误可能导致后一个相关的标签预测错误，后续研究工作中将着重在这方面进行改进。RoBERTa-LSTM相对BERT-thre的提升为3.2%。

4.3.3 Layer-attentive策略的作用

为验证多层特征融合策略对模型性能的影响，以及对比不同的融合方法，本组对比实验以原始BERT为基础模型，在此基础上分别使用3Lattv、3Lconcat、4Lattv和4Lconcat的方法进行实验，五种方法均采用基于LSTM的解码器，不同方法在三类数据上的得分见Table 5。

由Table 5可知，除BERT-4Lconcat方法外，其他多层特征融合方法优于原始BERT的方法。其次，除Labor案件下三层特征融合外，Layer-attentive的方法均优于concat线性拼接的方法，最大提升可达到2.1%。分别比较BERT(4Lconcat)和BERT(3Lconcat)，比

Table 5: Layer-attentive策略的作用

模型	Labor			Divorce			Loan		
	mi-F	ma-F	Ava	mi-F	ma-F	Ava	mi-F	ma-F	Ava
BERT	79.45	54.79	67.12	84.13	72.64	78.39	78.23	45.21	61.72
BERT(4Lconcat)	81.22	53.40	67.31	84.44	71.75	78.10	76.42	49.78	63.10
BERT(4Lattv)	79.08	55.75	67.41	84.29	72.24	78.27	79.03	49.43	64.23
BERT(3Lconcat)	80.50	56.08	**68.29**	84.37	71.87	78.12	76.50	51.82	64.16
BERT(3Lattv)	80.35	56.06	68.21	84.70	72.06	**78.38**	78.70	53.81	**66.26**

较BERT(4Lattv)和BERT(3Lattv)，可发现三层特征融合均优于四层特征融合。最后，对三类案件的得分进行横向比较，相同模型在三类案件上性能差异明显，主要原因是三类案件的数据量有一定差距，而且分别具有不同的要素类别体系。

4.3.4 模型案例分析

Figure 5所示为BiLSTM、BERT、WWM、WWM-LSTM四种模型分别对三类案件预测结果的例子。第一个例子为Labor案件，实际标签有三个，BiLSTM模型预测出0个，BERT只能预测出其中一个，而WWM可以预测出来两个标签，WWM-LSTM因为能捕获LB3和LB6之间的依赖关系，可以将三个标签全部预测出来。说明语言模型相对传统神经网络具有更强的学习能力。WWM因为考虑了中文分词问题，比原始的BERT具有更强的语义解析能力。第二个例子也展示了WWM更强的性能。第三个例子是Loan案件，原本句子没有标签，BiLSTM却错误地预测了一个标签，因为句子中含有“债权”关键字，BiLSTM只捕获了这个特征，没有理解语义信息，而语言模型的强大之处在于不仅能捕获浅层的语法特征，更能学习到深层的语义信息。

裁判文书句子描述	标签	BiLSTM	BERT	WWM	WWM-LSTM	标签含义
因被告不给原告签订劳动合同也不交纳社保，为此原告要求解除合同并支付经济赔偿金。	['LB1','LB3', LB6']	[]	['LB1']	['LB1', LB6']	['LB1', 'LB3', LB6']	LB1：解除劳动关系 LB3：支付经济补偿金 LB6：未签订劳动合同
故对原告诉请赵二×由原告抚养的请求本院予以支持。	['DV1', 'DV2']	[]	[]	['DV2']	['DV1', 'DV2']	DV1：婚后有子女 DV2：限制行为能力子女抚养
另查明，二审中，信×公司自认其因参与主债务人江×公司破产债权分配，已分配得到债权金额为×元。	[]	['LN1']	[]	[]	[]	LN1：债权人转让债权

Figure 5: 不同模型的预测结果示例

5 结束语

本文提出了一个基于预训练语言模型的多标签分类模型，该模型可实现面向司法领域的案件要素识别。该模型主要分为编码器和解码器两大部分，两部分间通过注意力机制进行交互，其中编码器部分采用基于Layer-attentive特征增强的语言模型，解码器采用LSTM序列生成模型。实验结果表明，本文提出的案件要素识别模型比基于循环神经网络的模型在F1值上提高了7.6%，比基础语言模型BERT提升约3.2%。本文采用的基于LSTM的多标签分类策略具有较大的性能增益，Layer-attentive的微调策略也有一定的性能提升。未来工作将研究要素类别的含义对要素识别结果的影响。

致谢

本文工作受国家重点研发计划（2018YFC0830603）资助。

参考文献

Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyang Zhao. 2017. Learning to predict charges for criminal cases with legal basis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2727-2736.

Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. 2018. Legal Judgment Prediction via Topological Learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 3540-3549.

Zikun Hu, Xiang Li, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2018. Few-shot charge prediction with discriminative legal attributes. In *Proceedings of the 27th International Conference on Computational Linguistics*, 487-498.

Limin Wang. 2018. *Research on Chinese Named Entity Recognition for Legal Documents*. Suzhou University.

Yun Xie. 2018. *Reserch on Naming Entry Recognition for Chinese Legal Texts*. Nanjing Normal University.

Yoon Kim. 2014. Convolutional Neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 1746-1751.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv* :1409.0473.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, 1480-1489.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2227-2237.

Jacob Devlin, Ming W. Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understandin. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2227-2237.

Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. SGM: Sequence generation model for multi-label classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, 3915-3926.

Fred Kort. 1957. Predicting Supreme Court decisions mathematically: A quantitative analysis of the "right to counsel" cases. *American Political Science Review 51*, no. 1: 1-12.

Sidney S. Ulmer. 1963. Quantitative Analysis of Judicial Processes: Some Practical and Theoretical Applications. *Law and Contemporary Problems*, 28(1):164-84.

Monica Shapira. 1990. Computerized decision technology in social service. *International Journal of Sociology and Social Policy*, 10, 138–164.

Patricia Hassett. 1993. Can Expert System Technology Contribute to Improved Bail Decisions? *International Journal of Law and Information Technology*, 1(2):144-144.

Nikolaos Aletras, Dimitrios Tsarapatsanis, Daniel Preoţiuc-Pietro, and Vasileios Lampos. 2016. Predicting judicial decisions of the European Court of Human Rights: A natural language processing perspective. *PeerJ Computer Science*, 2:e93.

Octavia M. Sulea, Marcos Zampieri, Shervin Malmasi, Mihaela Vela, Liviu P. Dinu, and Josef V. Genabith. 2017. Exploring the use of text classification in the legal domain. *arXiv preprint arXiv* :1710.09306.

Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, and Yansong Feng. 2018. Cail2018: A large-scale legal dataset for judgment prediction. *arXiv preprint arXiv* :1807.02478.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv* :1301.3781.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8): 1735-1780.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *arXiv preprint arXiv* :1703.03130.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998-6008.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R. Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, 5754-5764.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-training with whole word masking for chinese bert. *arXiv preprint arXiv* :1906.08101.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv* :1907.116921.

Yifan Qiao, Chenyan Xiong, Zhenghao Liu, and Zhiyuan Liu. 2019. Understanding the Behaviors of BERT in Ranking. *arXiv preprint arXiv* :1904.07531.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune BERT for text classification? *China National Conference on Chinese Computational Linguistics*, 194-206. Springer, Cham, 2019.

Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. 2017. Self-normalizing neural networks. *Advances in neural information processing systems*, 971-980.

Matthew R. Boutell, Jiebo Luo, Xipeng Shen, and Christopher M. Brown. 2004. Learning multi-label scene classification. *Pattern recognition 37*, no. 9: 1757-1771.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2978-2988.

基于拼音约束联合学习的汉语语音识别

梁仁凤[1,2]，余正涛[1,2*]，高盛祥[1,2]，黄于欣[1,2]，郭军军[1,2]，许树理[1,2]
1.昆明理工大学，信息工程与自动化学院，昆明，650500
2.昆明理工大学，云南省人工智能重点实验室，昆明，650500
{liangrenfeng3，ztyu}@hotmail.com，gaoshengxiang.yn@foxmail.com
{huangyuxin2004，guojjgb}@163.com，xushulitony@outlook.com

摘要

当前的语音识别模型在英语、法语等表音文字中已经取得很好的效果。然而，汉语是一种典型的表意文字，汉字与语音没有直接的对应关系，但拼音作为汉字读音的标注符号，与汉字存在相互转换的内在联系。因此，在汉语语音识别中利用拼音作为解码约束，引入一种更接近语音的归纳偏置。基于多任务学习框架，提出一种基于拼音约束联合学习的汉语语音识别方法，以端到端的汉字语音识别为主任务，以拼音语音识别为辅助任务，通过共享编码器，同时利用汉字与拼音识别结果作为监督信号，增强编码器对汉语语音的表达能力。实验结果表明，相比基线模型，提出方法取得更优的识别效果，词错误率WER降低了2.24个百分点。

关键词： 端到端；汉语语音识别；联合学习；拼音

Chinese Speech Recognition Based on Pinyin Constraint Joint Learning

Renfeng Liang, Zhengtao Yu, Shengxiang Gao, Yuxin Huang, Junjun Guo, Shuli Xu
1. Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, 650500, China
2. Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and Technology, Kunming, 650500, China
{liangrenfeng3, ztyu}@hotmail.com, gaoshengxiang.yn@foxmail.com
{huangyuxin2004, guojjgb}@163.com, xushulitony@outlook.com

Abstract

Current speech recognition models have achieved good results in phonetic language such as English and French. However, Chinese is a typical ideographic writing, and there is no direct correspondence between Chinese characters and phonetics, but Pinyin, as a mark of the pronunciation of Chinese characters, has an internal connection with Chinese characters. Therefore, Pinyin is used as a decoding constraint in Chinese speech recognition, and an inductive bias closer to speech is introduced. Based on a multi-task learning framework, a Chinese speech recognition method based on pinyin constraint joint learning is proposed. The end-to-end Chinese character speech recognition is the main task, and the pinyin speech recognition is the auxiliary task. By sharing the encoder, the Chinese characters and the pinyin recognition are used at the same time. The result is used as a supervision signal to enhance the ability of the encoder to express Chinese speech. Experimental results show that compared with the baseline model, the proposed method achieves better recognition results, and the word error rate WER is reduced by 2.24 percentage points

*通信作者：余正涛 ztyu@hotmail.com
项目基金：国家自然科学基金（61732005，61761026）;云南省高新技术产业专项（201606）

Keywords: End-to-end , Chinese speech recognition , Joint learning , Pinyin

1 引言

自动语音识别（Automatic Speech Recognition，ASR）是把语音中包含的词汇内容转换为计算机可理解的文本。随着深度学习的快速发展，ASR系统主要分为两类：传统混合系统和当前主流的端到端模型。传统混合系统 (Sainath et al., 2013)基于深度神经网络隐马尔可夫模型（Deep Neural Networks - Hidden Markov Models ，DNN-HMM）对声学模型建模、使用发音字典将音素序列转换为词、再通过一个语言模型将词序列映射为句子，系统训练时，这些声学、发音和语言组件有不同的激活函数，通常单独训练和优化。为了弥补传统混合系统的不足，当前流行的端到端模型 (Liu et al., 2019)将传统混合系统折叠为一个单一的神经网络，去除传统框架中所有中间步骤和独立子任务，输入语音特征，直接输出源语言文本，具有容易训练、模型简单和联合优化的优势，取得很好的效果。当前端到端模型流行的方法主要有连接时序分类算法 (Graves et al., 2006)（Connectionist Temporal Classification，CTC）CTC和使用CTC与注意力机制的混合方法 (Moritz et al., 2019)。CTC不需要对训练语料预先分段和后处理输出标签，然而，CTC基于条件独立假设训练ASR模型，缺乏对输入序列间上下文关系的建模。对此，注意力对齐机制 (Bahdanau et al., 2014)第一次使用到基于序列到序列结构的语音识别模型中 (Chorowski et al., 2014)，但由于过度灵敏的关注对齐方式应用到真实的语音识别场景中表现出比较差的效果。于是， (Kim et al., 2017)结合CTC和注意力机制的优势提出基于两者的混合语音识别模型， (Moritz et al., 2019)和 (Sarl et al., 2020)在混合模型的改进取得不错的效果。

综上所述，端到端的模型主要在英语、法语等表音文字的语音识别中取得很好的效果，然而，汉语是一种典型的表意文字，每一个汉字表示个别词或词素的形体，不与语音直接发生联系，当前端到端的模型对汉字的识别存在一些不足。 (Chan et al., 2016)对汉字识别的研究工作中表明模型对汉字的识别收敛速度较慢。拼音作为汉字的读音标注文字，直接表示汉字语音，拼音与汉字存在内在转换关系，基于音节（拼音）的研究工作 (Zhou et al., 2018)持续至今。将语音特征识别为音节单元 (Qu et al., 2017)、再通过一个转换模型将拼音变换为汉字 (Liu et al., 2015)的级联模型存在错误传播，为了避免这种问题， (Chan et al., 2016)提出汉字-拼音识别模型，只在训练时使用拼音帮助对汉字的识别，但是这种方法识别字符错误率（Character Error Rate，CER）达到59.3，对此， (Zhou et al., 2018)提出基于Transformer (Vaswani et al., 2017)的贪婪级联解码器模型，取得相对较好的效果。

基于以上研究工作，在汉语语音识别中，引入拼音作为对汉字解码的约束，能够促使模型学习更好的语音特征。在汉语中，对汉字的识别类似于语音翻译（Spoken Language Translation，ST） (Di Gangi et al., 2019)，对拼音的识别可以视为对汉语的语音识别。在ST领域， (Weiss et al., 2017)提出将语音识别和语音翻译联合学习可以有效提高模型翻译性能。从 (Weiss et al., 2017)的研究工作中受到启发，在多任务学习框架下Caruana (1997)，提出基于拼音约束联合学习的汉语语音识别方法，在汉语语音识别中引入拼音语音识别任务作为辅助任务联合训练，共同学习，相互促进。在AISHELL-1 (Bu et al., 2017)中文训练语料上，词错误率WER值比基线模型降低2.24个百分点。

2 基于拼音约束联合学习的汉语语音识别方法

模型共享一个编码器，拼音语音识别和汉语语音识别分别有一个解码器，训练时，模型的交叉熵是两个解码器分别计算损失后正则求和；反向传播时，编码器的参数被两个任务同时更新，达到两个任务共同促进，相互增强的效果。模型结合 (Weiss et al., 2017)研究工作和 (Kim et al., 2017)提出的混合模型，并对其做了进一步改进。具体结构概览图如图1所示，包括三个部分：共享编码器、拼音语音识别和基于拼音约束联合学习的汉字识别，本节分别将从1.1、1.2、1.3节对以上部分进行介绍。

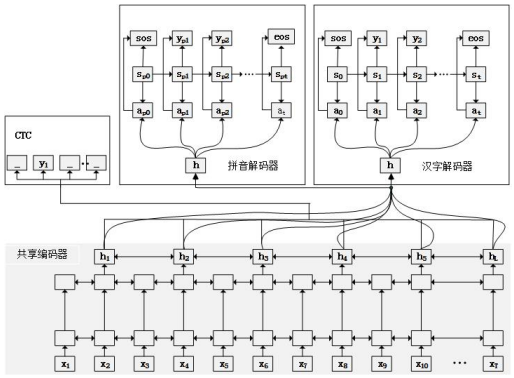


图1　基于拼音约束联合学习的汉语语音识别结构图

2.1　共享编码器

模型共享一个编码器，编码器采用双向长短期记忆网络（Long Short Term Memory networks, LSTM），双向LSTM结构见图1中的共享编码器部分。共享编码器将语音信号特征$x = (x_1, x_2, ..., x_T)$作为输入，使用VGG对x抽取特征转为高纬的隐表征，输出为$h = (h_1, h_2, ..., h_L)$。这里T表示语音特征的帧索引，L为对语音特征下采样后的帧索引$(L \leq T)$。编码器的编码过程表示为

$$h = Encoder\,(x) \qquad (1)$$

2.2　拼音语音识别

拼音语音识别模型采用当前流行的基于注意力机制的编码器-解码器框架，编码器采用2.1节介绍的共享编码器结构。解码器采用单向LSTM，见图1中的拼音解码器部分。解码器以共享编码器的输出h作为输入，基于时刻t前的输出标签序列，得到每一个t时刻预测拼音p标签y_{pt}的概率分布：

$$P\,(y_p|h) = \prod_t P\left(y_{pt}|h, y_{p(1:t-1)}\right) \qquad (2)$$

$$y_{pt} = LSTM\left(h, y_{p(1:t-1)}\right) \qquad (3)$$

对于每一时间步t，基于所有的输入语音特征h和注意力机制权重$a_{t,l}$产生文本向量c_t：

$$c_t = \sum_l a_{t,l} h_l \qquad (4)$$

这里的$a_{t,l}$通过softmax层计算：

$$a_{t,l} = \frac{\exp\left(\gamma e_{t,l}\right)}{\sum_l \exp\left(\gamma e_{t,l}\right)} \qquad (5)$$

$$e_{t,l} = \omega^T \tanh\left(W s_{t-1} + V h_l + U f_{t,l} + b\right) \qquad (6)$$

$$f_t = F * a_{t-1} \qquad (7)$$

这里，训练参数有ω、W、V、U、F、γ是模型的锐化因子，l为帧索引，$*$表示一维卷积，f_t通过$*$与卷积参数F计算得出。

解码器使用c_t、t时刻前的输出标签$y_{p(t-1)}$和隐状态s_{t-1}生成当前时刻的隐状态s_t和预测拼音标签y_{pt}：

$$s_t = LSTM\,(s_{t-1}, y_{t-1}, c_t) \qquad (8)$$

$$y_{pt} = Generate\,(s_t, c_t) \qquad (9)$$

这里LSTM代表单向循环神经网络，Generate代表前馈网络。

结合公式（2），拼音语音识别的损失函数可以通过以下公式计算：

$$L_p\,(h, y_p) = -\ln P\,(y_p|h) \qquad (10)$$

这里拼音序列$y_p = (y_{p1}, y_{p2}, ...y_{pt})$，其中$t \leq T$。

2.3 基于拼音约束联合学习的汉字识别

基于共享编码器的输出h作为输入，汉字解码器同样以h作为输入，结合时刻t前的输出标签序列，通过简单的前馈网络和softmax激活函数，得到每一个时刻t预测汉字标签y_t的概率分布$P(y|h)$，基于$P(y|h)$，汉字语音识别交叉损失熵可以通过以下公式计算：

$$L(h,y) = -\ln P(y|h) \qquad (11)$$

这里汉字序列$y = (y_1, y_2, ...y_t)$。

在多任务学习框架下，提出模型的交叉损失熵通过拼音解码器和汉字解码分别计算损失后的正则求和联合训练。拼音语音识别作为辅助任务帮助模型增强对汉字的识别能力，与此同时，汉语语音识别作为主要任务促进模型对拼音监督信号的感知。反向传播时，通过共享编码器，能同时感知拼音和汉字的监督信号，编码器的参数被拼音语音识别和汉字语音识别同时更新。结合公式（10）、（11），基于拼音约束联合学习的汉字识别交叉熵损失函数表示为

$$L_{hybrid}(h,y) = \lambda L_p(h,y_p) + (1-\lambda) L(h,y) \qquad (12)$$

这里λ为模型可微调的超参数：$\lambda \in (0,1)$。

考虑CTC具有使模型快速收敛的优势，且不需要对输入、输出序列做一一标注和对齐，因此，提出的模型结合了CTC。通常情况下，CTC与循环神经网络（Recurrent Neural Network，RNN）结合，RNN作为编码器，把语音特征序列x转为高纬的隐状态h，该编码器过程如公式（1）。基于语音隐表帧h，CTC假设输出汉字标签之间条件独立，标签之间允许插入空白表示（-），求出标签序列任何一条路径$\pi = (\pi_1, \pi_2, ..., \pi_T)$的概率分布$p(\pi/h)$，由于多条路径序列可能只对应一条汉字标签序列，通过定义一个多对一的映射函数F（$\pi \in F(y)$）将路径序列映射到标签序列y，采用前后向算法有效求得标签序列的最大概率分布$p(y/h)$，基于$p(y/h)$，可以计算CTC的负对数似然函数L_{CTC}。本文模型结合CTC的交叉熵损失函数通过以下公式计算：

$$L(h,y) = (1-\lambda_1) L_{hybrid}(h,y) + \lambda_1 L_{CTC}(h,y) \qquad (13)$$

$$L_{CTC}(h,y) = -\ln(P(y|h)) \qquad (14)$$

$$P(h,y) = \sum_{\pi \in F(y')} P(\pi|h) \qquad (15)$$

这里λ_1为模型可微调的超参数：$\lambda_1 \in (0,1)$，y'为映射标签序列。

3 实验

3.1 数据设置

见表1，使用由希尔贝壳中文普通话开源的语音数据库AISHELL-1 (Bu et al., 2017)证明了本文方法的有效性。该训练语料包括200个说话者，其中训练集有120098条语音（约150个小时），验证集有14326条语音（约10个小时），测试集有7176条语音（约5个小时）。通过Torchaudio[1]工具，提取以上训练语料步长为10毫秒、窗口大小为25毫秒、维度为40的梅尔倒频谱filter-bank特征。

分类	时长	男声	女声
训练集	150	161	179
验证集	10	12	28
测试集	5	13	7

Table 1: 实验训练集AISHELL-1

[1]https://pypi.org/project/torchaudio/

3.2 评价指标

本文使用词错误率作为模型的评价指标，词错误率简称WER（Word Error Rate），将模型预测的输出序列与监督信号序列进行比较，计算WER的公式：

$$WER = 100 * \frac{S + D + I}{N} \qquad (16)$$

这里S、D、I 表示替换、删除和插入的字数，N为监督信号字序列的总字数。

3.3 参数设置

对于未登录字，使用特殊字符“UNK代替”，超参数均设置为0.2时模型效果最好 (Kim et al., 2017)，dropout设为0.25。模型采用Adadelta算法Zeiler (2012)进行优化，batch-size设置为16，共享编码器采用4层的卷积网络和5层的双向LSTM，双向LSTM每个方向有512个隐状态单元，两个解码器均是一个单层的有512个隐状态单元的LSTM，Attention机制使用location-aware attention (Chorowski et al., 2014)。在词嵌入层，每个字表征为256纬的向量。拼音的字表大小为1400，汉语的字表大小为4500。

3.4 基线模型

本文共选择了3个基线模型，分别在训练数据集AISHELL-1进行试验，得到WER评分。模型包括基于音节的贪婪级联解码模型、S2S结合CTC的混合模型（S2S+CTC）和级联模型。

贪婪级联解码模型 (Zhou et al., 2018)，是使用两个beam search级联解码的Transformer模型。

混合S2S+CTC语音识别系统 (Kim et al., 2017)，是一种利用CTC和基于Attention序列到序列两者优势的模型，是当前常用的语音识别系统。

级联系统:将汉语语音特征序列识别为拼音文本序列，再采用一个额外的语言模型将拼音文本转写为汉语文本,采用由Pinyin2Hanzi3[2]将拼音文本序列转变为汉语文本序列。

3.5 本文方法有效性分析

对比基线模型，在AISHELL-1数据集上，验证了本文方法的有效性。使用WER值作为模型的评价指标，见下表2。

根据表2的实验结果分析：相比S2S+CTC（拼音识别），S2S+CTC（汉字识别）的WER值在验证集上高4.93个百分点，在测试集上高5.04个百分点，这说明当前的端到端语音识别模型对表意文字的识别效果不佳。相比基线模型S2S+CTC（汉字识别），提出模型在验证集上的WER值低2.5个百分点，在测试集上的WER值低2.24个百分点，说明在当前的汉语语音识别中引入拼音语音识别作为辅助任务联合训练，增强了模型对汉字的识别能力。相比级联系统+CTC，提出模型在验证集上的WER值低1.31个百分点，在测试集上低1.05个百分点，说明在汉语语音识别中引入拼音语音识别任务，提出的方法避免了级联系统导致的错误传播问题，取得比级联系统更好的识别效果。相比贪婪级联解码模型，提出模型在验证集上的WER值低6.1个百分点，在测试集上的WER值低4.95个百分点，这说明提出的模型在汉语语音识别中引入拼音取得相对较好的效果。

模型	λ, λ_1	WER（dev）	WER（test）
贪婪级联解码模型	-	16.16	17.64
S2S+CTC（拼音识别）	0,0.2	7.63	9.89
S2S+CTC（汉字识别）	0,0.2	12.56	14.93
级联系统+CTC	0,0	11.37	13.74
提出模型	0.2,0.2	10.06	12.69

Table 2: 提出模型对比基线模型的实验结果

为了讨论拼音语音识别任务和CTC对汉字识别的影响，对提出的模型去除CTC结构进行消融性实验结果分析，且分别将级联系统和S2S+CTC模型均消去CTC结构。三个模型训练

[2]https://github.com/letiantian/Pinyin2Hanzi

时间基本一致。相比S2S-CTC（拼音识别），S2S-CTC（汉字识别）在验证集集上的WER值高6.23个百分点，在测试集上的WER值高6.45个百分点，说明当前的端到端语音识别系统对表意文字的识别效果不佳。相比基线模型S2S-CTC（汉字识别），提出模型-CTC在验证集的WER值低2.61个百分点，在测试集上低2.57个百分点；相比级联系统-CTC，提出模型-CTC在验证集上低1.5个百分点，在测试集上低2.31个百分点，说明提出模型在不受CTC影响下，引入拼音约束联合学习，增强了模型对语音特征的表达。

模型	λ, λ_1	WER（dev）	WER（test）
S2S-CTC（拼音识别）	0,0	10.57	12.57
S2S-CTC（汉字识别）	0,0	16.80	19.02
级联系统-CTC	0,0	15.69	18.76
提出模型-CTC	0.2,0	14.19	16.45

Table 3: 消融性实验结果分析

结论

由于汉字与语音没有直接的联系，拼音与汉字、语音具有内在关系，提出基于拼音约束联合学习的汉语语音识别方法，通过多任务学习框架，联合拼音语音识别、汉字语音识别任务共同学习，取得了更好的效果。进一步研究工作可以将拼音序列变换汉字序列视为一个机器翻译任务，通过共享解码器方式的联合学习等。

参考文献

Ashish Vaswani,Noam Shazeer,Niki Parmar,Jakob Uszkoreit,Llion Jones,Aidan N. Gomez ,Łukasz Kaiser,Lukaszkaiser,Illia Polosukhin 2017. Alternation. *Attention is All you Need*, Neural Information Processing Systems.

Alex Graves, Santiago Fernández, and Faustino Gomez. 2006. Alternation. *Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks*, International Conference on Machine Learning ACM.

Alexander H. Liu, Tzu-Wei Sung, Shun-Po Chuang, Hung-yi Lee, Lin-shan Lee 2019. Alternation. *Sequence-to-sequence Automatic Speech Recognition with Word Embedding Regularization and Fused Decoding*, arXiv.

Dzmitry Bahdanau, Kyunghyun Cho,and Yoshua Bengio 2014. Alternation. *Neural Machine Translation by Jointly Learning to Align and Translate*, Computer Science.

Hui Bu, Jiayu Du, Xingyu Na, Bengu Wu, Hao Zheng. 2017. Alternation. *Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline*, 2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA). IEEE, 2017: 1-5.

Jan Chorowski, Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio 2014. Alternation. *End-to-end continuous speech recognition using attention-based recurrent NN: First results*, preprint arXiv.

Leda Sarl, Niko Moritz, Takaaki Hori,and Jonathan Le Roux 2020. Alternation. *Unsupervised Speaker Adaptation using Attention-based Speaker Memory for End-to-End ASR*, IEEE International Conference on Acoustics,Speech and Signal Processing.

Mattia A. Di Gangi, Matteo Negri and Marco Turchi 2019. Alternation. *Adapting Transformer to End-to-End Spoken Language Translation*, conference of the international speech communication association.

Matthew D. Zeiler 2012. *Adadelta: an adaptive learning rate method*, preprint arXiv.

Niko Moritz, Takaaki Hori,and Jonathan Le Roux 2019. Alternation. *Streaming end-to-end speech recognition with joint CTC-attention based models*, IEEE ASRU.

Niko Moritz, Takaaki Hori,and Jonathan Le Roux 2019. Alternation. *Triggered attention for end-to-end speech recognition*, in Proc. IEEE ICASSP, May2019, pp. 5666–5670.

Ron J. Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu,and Zhifeng Chen 2017. Alternation. *Sequence-to-sequence models can directly translate foreign speech*, Proceedings of Interspeech.

Rich Caruana 1997. *Multitask learning. Machine Learning.* 28(1):41–75

Shiyu Zhou, Linhao Dong, Shuang Xu, and Bo Xu. 2018. Alternation. *Syllable based sequence-to-sequence speech recognition with thetransformerin mandarin chinese*, Interspeech.

Suyoun Kim,Takaaki Hori,and Shinji Watanabe 2017. Alternation. *Joint ctc-attention based endto-end speech recognition using multi-task learning*, International Conference on Acoustics, Speech and Signal Processing (ICASSP).

Tara Sainath, Abdel-rahman Mohamed, Brian Kingsbury, and Bhuvana Ramabhadran. 2013. Alternation. *Deep Convolutional Neural Networks for LVCSR*, IEEE International Conference on Acoustics,Speech and Signal Processing.

William Chan,Ian Lane 2016. Alternation. *On online attention-based speech recognition and joint mandarin character-pinyin training*, Interspeech.

Yi Liu,Jing Hua,Xiangang Li,Tong Fu, and Xihong Wu 2015. Alternation. *Chinese syllable-tocharacter conversion with recurrent neural network based supervised sequence labelling*, Signal and Information Processing Association Annual Summit and Conference.

Zhongdi Qu,Parisa Haghani,and Eugene Weinstein 2017. Alternation. *Syllable-based acoustic modeling with ctc-smbr-lstm*, IEEE ASRU.

基于数据增强和多任务特征学习的中文语法错误检测方法

谢海华[1] ✉，陈志优[1]，程静[1]，吕肖庆[1,2]，汤帜[1,2]
1. 北大方正集团有限公司数字出版技术国家重点实验室，北京市海淀区，100871
2. 北京大学王选计算机研究所，北京市海淀区，100871
xiehh@founder.com.cn

摘要

由于中文语法的复杂性，中文语法错误检测（CGED）的难度较大，而训练语料和相关研究的缺乏，使得CGED的效果还远达不到能够实用的程度。本文提出一种CGED模型，采用数据增强、预训练语言模型和基于语言学特征多任务学习的方式，弥补训练语料稀缺的不足。数据增强能够有效地扩充训练集，预训练语言模型蕴含丰富的语义信息有助于语法分析，基于语言学特征多任务学习对语言模型进行优化则可以使语言模型学习到跟语法错误检测相关的语言学特征。本文提出的方法在NLPTEA的CGED数据集进行测试，取得了优于其他模型的结果。

关键词： 中文语法错误检测 ； CGED ； 数据增强 ； 多任务学习

Chinese Grammar Error Detection based on Data Enhancement and Multi-task Feature Learning

Haihua Xie[1] ✉, Zhiyou Chen[1], Jing Cheng[1], Xiaoqing Lyu[1,2], Zhi Tang[1,2]
1. State Key Laboratory of Digital Publishing Technology,
Peking University Founder Group Co. LTD., Beijing, China, 100871
2.Wangxuan Institute of Computer Technology,
Peking University, Beijing, China, 100871
xiehh@founder.com.cn

Abstract

Due to the complexity of Chinese grammars, Chinese grammar error diagnosis (CGED) is a challenging task, and the lack of training corpus and relevant study makes the current approaches of CGED still far from practical applications. In this paper, we propose a CGED model to compensate for the low-resource defect using data augmentation, pre-trained language model and multi-task learning of linguistic features. Data augmentation can effectively expand the training set, and pre-trained language models are rich in semantic information that is conducive to grammatical analysis. Meanwhile, enhancing language models based on multi-task learning of linguistic features enables the model to learn linguistic features useful for grammatical error diagnosis. The method proposed in this paper was tested on the CGED dataset of NLPTEA and obtained better results than other models.

Keywords: Chinese Grammar Error Detection , CGED , Data Enhancement , Multi-task Learning

1 引言

中文语法错误检测（Chinese Grammatical Error Diagnosis，CGED）的目标是自动检测出中文自然语句中的语法错误，例如：成分缺失或多余，语序不当等。CGED的检测任务一般包含：是否存在错误、错误类型、错误发生位置。虽然不能给出纠正错误的建议，CGED对于辅助写作和文档审校等场景依然十分有意义。在辅助写作中，CGED给出语法错误类型和位置以让作者针对性地修改文章，可以提升写作的质量和效率。另外，在出版行业的审校环节，由于正式出版物的格式要求十分严格，CGED自动检测出一些基础的语法错误有助于节省审校人员大量的时间，而直接纠正语法错误则可能造成文章的内容和逻辑发生变化。

目前，有关语法错误检测的研究大多数是针对英文的。与英文相比，中文的语法更加复杂和灵活。中文不存在词语的单复数和时态等明确的语法规则，其语法错误经常涉及隐晦的语义解析而不能基于字词形态来判断 (Fu, et al., 2018)。因此，现有的英文语法错误检测方法不能很好地适用于CGED。另外，目前研究者倾向于运用生成式的方法直接进行语法改错，跳过了语法错误检测的步骤 (Chris, Dolan and Gamon, 2018; Zheng and Briscoe, 2018; Zhou, et al., 2018)。只有少量的研究采用序列标注方法进行中文语法错误检测。然而，由于缺乏大规模高质量的标注语料作为训练集，CGED的准确率往往不高，达不到实用水平。如何在训练数据有限的情况下提高语法错误检测的效果是该类研究的一个难点。

针对上述问题，本文提出一种基于数据增强和语言学特征多任务训练方法来提升中文语法错误检测的效果。针对训练语料不充足的问题，本研究使用大量无标签的正确中文语料，通过词性规则、句法规则以及语言模型概率统计等方法来生成接近真实语法错误用例的样本，以扩充训练语料。此外，本研究采用预训练语言模型对字词进行表征，以利用大规模语料蕴含的语义信息，并将词法学习、句法学习、语法错误检测等任务结合进行多任务学习，进一步获取中文语义和语法信息。本文提出的方法在NLPTEA CGED评测任务数据集进行测试，准确率和召回率分别为85.16%和72.53%（F1值为0.783）,性能优于其他中文语法检测模型。

2 相关工作

中文语法错误自动检测模型采取的方法从最初的统计学习方法 (Chang, Wu and Prasetyo, 2012)和基于规则的分析 (Lee, et al., 2013)，到现在主流的深度学习算法 (Fu, et al., 2018; Yang, et al., 2017)，以及多种模型混合的方法 (Li, et al., 2018)。大多数研究采用序列标注模型来进行语法错误检测，并使用LSTM和CRF来实现 (Fu, et al., 2018; Yang, et al., 2017; Zhao, Li and Lin, 2018)。使用LSTM模型实现语法错误检测时，特征的选择十分重要，除了通常使用的字向量特征、词向量特征、词性POS特征，很多研究提出了许多新的特征 (Fu, et al., 2018; Li, et al., 2018; Zhao, Li and Lin, 2018)。例如：高斯互信息（ePMI）、向量词的共现(AWC)、依赖关系词语的共现（DWC）、基于语境的词表达等。也有一些研究针对LSTM模型结构进行改进，比如在LSTM模型中加入策略梯度 (Li and Qi, 2018)。这些研究的重点在于学习中文语法规律，基于无标注语料统计词语规律和词语用法，并提出相应的特征来提高检测效果。然而，统计特征不能捕获深层的语法和语义信息，因此一些隐晦的语法错误无法被发现。

针对训练语料不足的问题，一些研究者使用未标注的中文语料来构造错误用例，例如：通过随机增加、删除、替换字词和打乱字词顺序来生成错误样本 (Wang, et al., 2019)；统计已有训练语料中的语法错误分布，并构造相应的错误样本 (Zhang, et al., 2018)。前者采用随机方式构造的语法错误样本，往往显得不够真实，其语法错误分布与正常写作者所犯错误的分布相差较大。而后者构造的错误数据过于拟合已有的训练样本，不利于模型的泛化。

近年来，一些学者利用基于大规模语料预训练的语言模型来获取文本的语言学特征，以弥补训练语料的不足。基于预训练语言模型的语法错误检测模型，其效果优于通过融合多种特征构建的模型 (Bell, Yannakoudakis and Rei, 2019; Kaneko and Komachi, 2019)。不过这些方法都以英文为研究对象，它们尚未在中文数据集上进行试验或者测试性能。

大多数情况下，语法错误检测的目的是为了对语法错误进行纠正。在检测出语法错误的类型和发生位置之后，可以根据错误类型，采用相应的方法来修改语法错误。例如：错误提示为“成分冗余”，则直接删除该成分；错误提示为“用词不当”，则基于词语统计信息（例

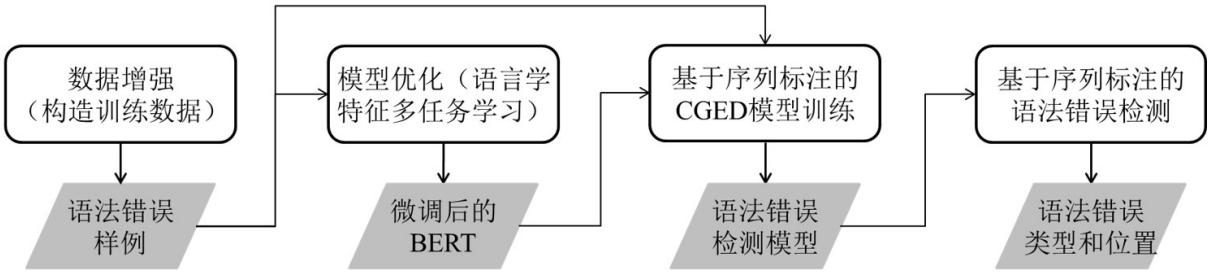


Figure 1: 基于数据增强和语言学特征多任务学习的中文语法错误检测系统框架

如：PMI）推荐候选词语以替换错误词语 (Fu, et al., 2018; Zhang, et al., 2018)。不过目前中文语法错误纠正的研究大多采用端到端的生成式方法，使用统计翻译模型 (Chris, Dolan and Gamon, 2018)、深度学习模型 (Zheng and Briscoe, 2018)、融合规则和统计的算法 (Zhou, et al., 2018)等，由错误句子直接生成正确的句子。但是生成的结果有时会改变原文的表达方式甚至语义和逻辑，在很多情况下不能产生令人满意的结果。

3 基于数据增强和语言学特征多任务学习的CGED模型

这一节将详细介绍我们提出的CGED模型。为了解决训练语料缺乏的问题，我们采用数据增强方法来扩充训练数据集，采用预训练语言模型BERT (Devlin, et al., 2019)作为基础的文本表征提取工具，并运用多任务训练数据来调整BERT参数以使它学习到更多的语言学特征。我们的语法错误检测系统框架见Figure 1。

我们的主要贡献是提出了基于句法分析与预训练语言模型采样的数据增强方法和基于语言学特征多任务学习的模型优化方法。以下章节将对Figure 1所示流程和上述两个贡献进行详细阐述。

3.1 基于句法分析与预训练语言模型采样的数据增强（构造训练数据）

中文语法错误检测研究的主要问题之一是训练语料的缺乏。我们使用大量未经标注的正确语句构造含有语法错误的训练样例，以弥补训练数据不足的问题。中文维基百科覆盖面广且表达方式丰富，人民日报表达方式规整规范，所以我们以维基百科和人民日报中文数据集为基础，抽取其中正确的语句，并对数据进行处理后构造训练样本。主要步骤的介绍如下。

1. 数据集预处理，主要的处理手段如下。
   - 增加数据的一致性和减少噪音，例如：将中文维基百科的繁体中文转化成简体中文，把全角字符转化为半角字符。
   - 运用中文处理工具对文本进行分词、词性标注、命名实体识别和依存句法分析。
   - 选择质量较高的句子，例如：去除过长（词数超过100个）和过短（词数小于3个）的句子。
2. 错误样例构造。本步骤将一些正确的语句改造为含有语法错误的语句。在语句经过分词、词性标注和依存句法分析之后，我们采用以下措施，构建不同类型的语法错误的训练样本。
   - 成分冗余构造：在语句的词语之间随机插入没有实际意义的词语。候选的插入词语选自停用词表。
   - 成分缺失构造：从主谓结构片段中删除主语或者谓语，从动宾结构片段中删除谓语或者宾语，从状中结构或者定中结构片段中删除被修饰成分。
   - 语序不当构造：修改动宾结构、状中结构、定中结构等结构片段中成分的顺序。
   - 用词不当构造：随机选取一个词语并将其遮盖（用MASK将其替换），然后用BERT的Masked LM预测出的候选字替换原来的字符。

为了保证改造后的句子在含有语法错误的同时，保持语句的基本语义和结构，以免发生意思改变，我们设计了以下规则。

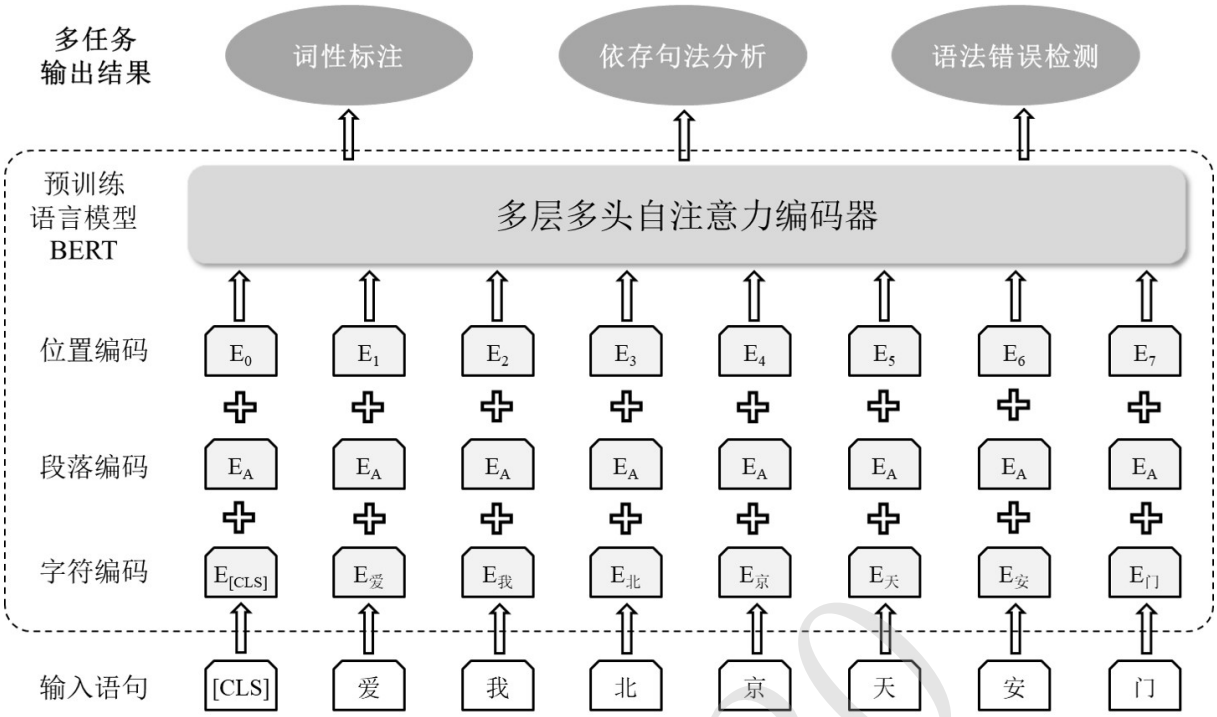


Figure 2: 基于语言学特征的多任务学习进行BERT模型优化

1. 不对命名实体进行修改。命名实体在句子中往往是主体成分，修改命名实体会改变句子的意思，例如：句子“协和医院是中国最好的医院之一，专治各种疑难杂症”，如果对“协和医院”进行修改，语句的意思就会发生变化。
2. 对于短句子，我们构造的样例中只含有一个错误。对于15个词语以上的句子，我们会随机增加错误。
3. 在成分缺失和语序不当构造时，规避修改依赖距离很远的结构成分，防止破坏语句结构。
4. 关于用词不当构造，除了构造“的地得”之间的误用情况，不对虚词、语气词之类无意义的词语进行修改以构造该类错误。实际样例中，虚词的使用错误主要是成分缺失和冗余。

以下是两个构造的错误样例。

1. 样例一：语序不当构造
   - 原句：**加速推广**菌草技术，将其列入国家开发计划。
   - 构造句：**推广加速**菌草技术，将其列入国家开发计划。
2. 样例二：用词不当构造
   - 原句：我跟朋友们经常用手机**打**电话聊天。
   - 构造句：我跟朋友们经常用手机**找**电话聊天。

3.2 基于语言学特征多任务学习的模型优化

在以往的CGED研究中，研究者使用的主流模型是BiLSTM-CRF结构。由于中文语法错误的复杂性和多样性，语法的正确使用与语言学特征高度相关，因此使用少量的训练数据很难训练出一个鲁棒性好的CGED模型，人们会在模型中加入词性、N-gram、PMI等语言学特征。但是，大量特征的使用使得模型结构繁琐，而且提取这些特征信息也大大降低了模型的运行速度。

我们使用BERT之类的预训练语言模型作为基础来构建CGED模型，以利用它们在预训练阶段学习到的深层语义信息。然后，我们采取多任务学习方式对BERT的参数进行调整，使模

	我	爱	北	京	天	安	门
我	0	SBV	0	0	0	0	0
爱	SBV	HEAD	0	0	VOB	VOB	VOB
北	0	0	0	0	ATT	ATT	ATT
京	0	0	0	0	ATT	ATT	ATT
天	0	VOB	ATT	ATT	0	0	0
安	0	VOB	ATT	ATT	0	0	0
门	0	VOB	ATT	ATT	0	0	0

HEAD： 主干词
SBV： 主谓关系
VOB： 动宾关系
ATT： 定中关系

Figure 3: 依存句法结构矩阵示例

型学习到各种语言学知识，并在预测阶段不必进行语言学特征提取，以提高模型的性能和效率。

多任务学习是指为模型设置多个训练目标，这些任务之间具有一定关联，并在训练阶段可以互相促进以达到更好的训练效果。多任务学习可以通过在模型上设置一些共享参数来实现。本文提出的方法使用BERT作为模型的共享部分，并使用不同结构来实现词性标注、句法分析和语法错误分类三个具体任务。基于语言学特征的多任务学习进行BERT模型优化的结构如Figure 2。

Figure 2所示模型的输出目标包括：词性标注，依存句法分析和语法错误检测。基于这三项任务的训练，可以对BERT的参数进行优化，以使BERT能学到更多的语言学知识。我们认为，这三个任务之间有互相促进的作用，词性和句法分析的结果能辅助判断语句是否有语法错误，例如：图2示例句是一个语法错误句，它的词性标注的结果是：动词-代词-名词，这个词性序列在中文语句中不常见，因此该句很可能含有语法错误。同样地，判断出语句含有语法错误，也有益于更准确地分析语句的词性和句法。这三个任务的详细描述如下。

1. 词性标注

   我们采用序列标注方法来实现词性标注任务，在BERT之后增加一个全连接层直接输出词性结果。由于BERT采用字符嵌入方式，对于多字符词语，我们采用“BI”的标注方式（‘B’表示词语开始位置，‘I’表示词语中间或结束位置）进行词性标注。在准备训练数据时，词性标注的标签可以由中文处理工具（例如pyltp (pyltp, 2020)）直接生成，标注示例如Table 1。

Character	爱	我	北	京	天	安	门
POS tag	B-v	B-r	B-n	I-n	B-n	I-n	I-n

Table 1: 词性标注示例

2. 依存句法分析

   依存句法分析的目的是确定语句的句法结构，通常以句法树的形式，用有向弧表示词语之间的修饰及指向关系（即依存关系）。在本文中，我们将句法结构（或词语之间的依存

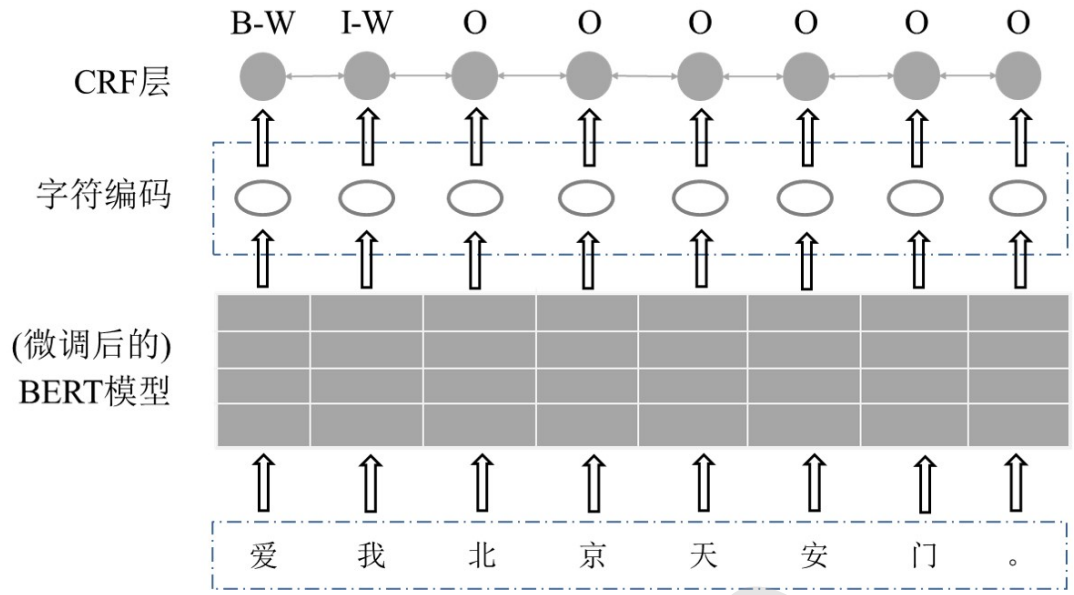


Figure 4: 基于BERT-CRF架构的中文语法错误检测模型

关系）用矩阵形式来表示。对于一个含有n个字的句子，用一个$n \times n$的矩阵表达词语之间的依存关系。为了避免关系矩阵（记为M）过于稀疏，我们将依存关系进行简化，取消修饰词和被修饰词之间的指向关系，所以M是一个对称矩阵。假设语句的第i个词（含有一个字符，在句子中的序号设为i^c）与第j个词（含有三个字符，在句子中的序号分别为j_1^c，j_2^c，j_3^c）之间的关系为动宾关系（VOB），则有$M_{i^c j_1^c} = M_{i^c j_2^c} = M_{i^c j_3^c} = VOB$，而且$M_{j_1^c i^c} = M_{j_2^c i^c} = M_{j_3^c i^c} = VOB$。我们将语句的主干词对应的对角线位置的值设置为Head，而对角线上其他位置的值设为0。以矩阵表示的句法结构示例如Figure 3。

在准备训练数据时，语句的句法结构矩阵可以由中文处理工具生成的句法树修改而成。在参数优化阶段，假设输入语句为S，其文本序列长度为t，经过BERT之后的语义表征为S_{BERT}，它的维度为$t \times 768$。然后采用以下公式产生两个中间变量H_1和H_2。

$$H_i = f(W_i S_{BERT} + b_i) \tag{1}$$

f表示对矩阵进行形变操作的函数，W_i和b_i是随机初始化并在训练中更新的参数。产生的H_1和H_2的维度都是$64 \times t \times 12$。然后基于以下公式产生句法结构分析结果。

$$M = Softmax[W(H_1 \cdot H_2^T)] \tag{2}$$

M的维度$64 \times t \times t$，对应$t \times t$矩阵的每个元素的数值（维度是1×64），即句法结构矩阵的结果。

3. 语法错误检测

我们采用多标签分类的方法完成语法错误检测任务，在BERT之后增加一个全连接层直接输出分类结果。分类的结果是句子含有的语法错误的类型。如果语句不含语法错误则输出“没有错误”，如果它含有多个语法错误则输出多个语法错误标签。语法错误检测的训练数据是由前文所述方法构造出来或者实际写作中产生。

上述三个任务模型的损失函数都用交叉熵来计算。多任务学习模型的损失函数是这三个模型的损失函数之和，模型训练的目标是最小化该损失函数。

Model	FPR	Detection level				Identification level				Position level			
		Acc.	Pre.	Rec.	F1	Acc.	Pre.	Rec.	F1	Acc.	Pre.	Rec.	F1
NLPTEA-18-HSK													
*B0	0.2138	0.7153	0.8395	0.6759	0.7489	0.6253	0.6518	0.4305	0.5185	0.4935	0.4395	0.248	0.3171
*B0+MTL	0.2438	0.7416	0.8354	0.733	0.7809	0.6293	0.6309	0.4854	0.5486	0.472	0.4062	0.2785	0.3304
*B0+MTL+DA	0.2106	0.743	0.8516	0.7253	0.7833	0.6462	0.6619	0.4827	0.5583	0.4843	0.4234	0.2769	0.3348
NLPTEA-16-HSK													
*B0	0.2182	0.7828	0.7771	0.7792	0.7782	0.7371	0.6993	0.582	0.6353	0.6492	0.5703	0.4173	0.482
*B0+MTL	0.2619	0.7768	0.7491	0.8173	0.7817	0.7176	0.663	0.6225	0.6421	0.617	0.5313	0.452	0.4884
*B0+MTL+DA	0.1975	0.795	0.7922	0.8074	0.7997	0.7457	0.7063	0.5996	0.6485	0.6543	0.5713	0.4341	0.4933
NLPTEA-16-TOCFL													
*B0	0.2132	0.6905	0.7512	0.6005	0.6676	0.618	0.5858	0.3753	0.4575	0.5024	0.3671	0.1986	0.2576
*B0+MTL	0.2666	0.7044	0.7514	0.6773	0.7124	0.608	0.5514	0.4322	0.4863	0.5066	0.3698	0.2301	0.2836
*B0+MTL+DA	0.2495	0.706	0.7498	0.6742	0.71	0.617	0.5761	0.4244	0.4886	0.5123	0.3718	0.2283	0.2828

Table 2: 中文语法错误检测模型的对比实验结果

3.3 基于序列标注的CGED模型训练和应用

我们把CGED视为序列标注问题，并选用BERT-CRF结构作为模型的基本架构，其中BERT的参数经过经过2.2节所述方法进行调整，见Figure 4。由于我们处理的对象是中文数据，我们使用中文BERT模型，它是基于大量中文维基百科语料预训练而成。在BERT之后使用CRF模型 (Sutton and McCallum, 2012)，一种经典的序列标注方法，直接生成语法错误检测的结果。语法错误标签使用“BIO”方式编码，“B”代表错误的开始位置，“I”表示中间或者结束位置，“O”表示当前字符没有语法问题。例如对于错误X，“B-X”代表“X”错误的第一个位置，“I-X”表示其他位置。

在训练阶段，训练数据集的部分数据来自人们在实际写作中出现的语法错误，而另一部分则来自前文所述方法构造出的数据。训练模型和预测模型的结构是一样的，输出的结果包含：是否存在错误，错误类型以及错误发生的位置。

4 中文语法错误检测实验

我们采用NLPTEA中文语法错误检测评测数据集[18]试验了我们的方法。NLPTEA提供一份标注过的语法错误数据集，语料来源是汉语非母语的汉语学习者在中文写作当中产生的错误样例。该数据集将语法错误分为四种类型：redundant errors（记为‘R’，即成分冗余），missing words（记为‘M’，即成分缺失），word selection errors（记为‘S’，即用词不当）和word ordering errors（记为‘W’，即词序不当）。数据集里的语句可能没有语法错误，也可能含有一个或多个语法错误。语法错误检测系统需要从以下三个方面对语句进行检测：

- Detection-level：检测语句是否含有语法错误。
- Identification-level：语句含有的语法错误的类型。
- Position-level：语句含有的语法错误的发生位置。

4.1 数据收集和处理

我们使用pyltp中文处理工具对语句进行分词、词性标注和依存句法分析，同时采用pyltp的标注体系。在多任务学习优化BERT时，我们使用了一些公开数据集来提升分词的准确性，以提高词性标注和依存句法分析的准确度。

我们收集了NLPTEA 2016，IJCNLP 2017和NLPTEA 2018的CGED任务的评测数据集，共有语句数量为20,451，按照句号、问号和感叹号拆分之后的语句数量为104,141。选择其中的80%数据作为训练数据，其余数据为校验数据。同时，我们收集和整理了中文维基百科数据集和人民日报数据集，使用2.1节介绍的数据构造方法生成训练数据（语句总数为138,825）并加入到训练集。为了维持正确语句和错误语句的比例，我们在数据集中加入了同等数量的不含语法错误的语句。

Model	FPR	Detection level				Identification level			Position level		
		Acc.	Pre.	Rec.	F1	Pre.	Rec.	F1	Pre.	Rec.	F1
B0+MFF+DA	0.2106	0.743	0.8516	0.7253	0.7833	0.6619	0.4827	0.5583	0.4234	0.2769	0.3348
HFL											
*run1	0.1613	0.7101	0.8276	0.609	0.7017	0.7107	0.4173	0.5259	0.5341	0.2729	0.3612
*run2	0.7554	0.6436	0.6171	0.9572	0.7504	0.3931	0.7331	0.5118	0.1441	0.3886	0.2102
*run3	0.1754	0.7278	0.8254	0.6517	0.7283	0.6874	0.4588	0.5503	0.4752	0.2906	0.3606
CMMC-BDRC											
*run1	0.5314	0.6889	0.6736	0.8621	0.7563	0.4834	0.5952	0.5335	0.2741	0.3177	0.2943
*run2	0.3574	0.6988	0.7266	0.7408	0.7336	0.5831	0.4955	0.5357	0.3839	0.2966	0.3346
*run3	0.347	0.663	0.7109	0.6709	0.6903	0.4853	0.4096	0.4442	0.2482	0.1814	0.2096
NCYU											
*run1	0.9987	0.5596	0.5598	0.9985	0.7174	0.2381	0.9749	0.3828	0.0030	0.0390	0.0056
*run2	0.9994	0.5599	0.5599	0.9995	0.7177	0.2382	0.9752	0.3828	0.0030	0.0384	0.0056
*run3	0.9994	0.5599	0.5599	0.9995	0.7177	0.2382	0.9752	0.3828	0.0030	0.0380	0.0055

Table 3: BERT+MFF+DA与NLPTEA 2018 CGED评测模型的对比

4.2 实验结果

我们按照2.2节介绍的方法，运用训练数据对BERT的参数进行调整。然后使用训练数据对语法错误检测的BERT+CRF模型进行训练，使用校验数据进行测试。我们同时使用不同的模型进行了对比实验，Table 2显示了对比实验的结果。其中，B0表示未经过优化的BERT模型，MTL表示多任务学习方法，DA表示数据增强，B0+MTL+DA则表示文本采用的方法。不同的模型分别在NLPTEA 2018 CGED任务的HSK测试集（NLPTEA-18-HSK）、NLPTEA 2016 CGED任务的HSK测试集（NLPTEA-16-HSK）和TOCFL（NLPTEA-16-TOCFL）测试集上进行了实验。

对比实验结果表明，使用语言学特征对BERT进行优化之后，语法错误检测的效果在各方面都有明显的提升，特别是检测的召回率得到很大提高。但是随着召回率的上升，检测精确率有一定程度的下降，不过数据增强的使用很好地弥补了这个问题，使得模型能够同时提高检测得召回率和精确率，并使F1指标提升。

我们与NLPTEA 2018 CGED评测结果进行了横向对比。我们没有采用模型融合以进一步提高检测效果，只用单一模型来与NLPTEA 2018评测效果较好的模型进行对比，结果见Table 3。HFL，CMMC-BDRC和NCYU是NLPTEA 2018评测结果里面准确率，召回率或者F1值较高的模型。在Detection Level和Identification Level这两个测试指标上，我们的单模型都取得了最优的F1值。但是在Position Level指标上，我们方法的效果不如HFL。经过分析，我们认为这可能是因为构造的错误案例与实际测试的错误案例在错误分布的不一致而造成的。

5 总结与展望

我们针对中文语法错误检测研究存在的主要问题之一，训练语料的缺乏，采用数据增强、预训练语言模型和语言学特征多任务学习的方式，有效地提高了语法错误检测的效果。使用语言学特征对语言模型进行优化能够使它学习到显式的语言学特征以及隐藏的语义信息，而语言学特征和语法使用是十分相关的，所以它对语法错误检测效果有明显的改善作用。

由于中文语法的复杂性，我们目前的工作依然存在很多不足，错误类型和位置的检测效果不好。在下一步工作计划中，我们将进一步提高数据构造的合理性，使构造的错误样本更符合人们实际所犯的语法错误。另外，我们会对语言学特征的多任务学习的结构进行改善，以进一步提高CGED任务的检测效果。

致谢

本项研究以下项目的支持：国家重点研发计划（No. 2019YFB1406302），国家自然科学基金项目（No. 61472014, No. 61573028, No. 61432020），北京市自然科学基金项目（No.

4142023），北京新星计划项目（XX2015B010）。感谢所有审稿人给出的宝贵意见和建议。

参考文献

Ruiji Fu, Zhengqi Pei, Jiefu Gong, et al. 2018. Chinese grammatical error diagnosis using statistical and prior knowledge driven features with probabilistic ensemble enhancement. *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications, NLP-TEA@ACL 2018*: 52-59.

Ru-Yng Chang, Chung-Hsien Wu, and Philips Kokoh Prasetyo. 2012. Error Diagnosis of Chinese Sentences Using Inductive Learning Algorithm and Decomposition-Based Testing Mechanism. *ACM Transactions on Asian Language Information Processing*, 3:1-3:24

Lung-Hao Lee, Li-Ping Chang, Kuei-Ching Lee, et al. 2013. Linguistic rules based Chinese error detection for second language learning. *Proceedings of the 21st International Conference on Computers in Education, 2013*: 27-29.

Yi Yang, Pengjun Xie, Jun Tao, et al. 2017. Embedding grammatical features into LSTMs for Chinese grammatical error diagnosis task. *Proceedings of the IJCNLP 2017, Shared Tasks, 2017*: 41-46.

Chen Li, Junpei Zhou, Zuyi Bao, et al. 2018. A hybrid system for Chinese grammatical error diagnosis and correction. *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications, NLP-TEA@ACL 2018*: 60-69.

Jianbo Zhao, Si Li, Zhiqing Lin. 2018. Contextualized character representation for Chinese grammatical error diagnosis. *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications, NLP-TEA@ACL 2018*: 172-179.

Changliang Li, Ji Qi. 2018. Chinese grammatical error diagnosis based on policy gradient LSTM model. *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications, NLP-TEA@ACL 2018*: 77-82.

Brockett Chris, William B. Dolan, and Michael Gamon. 2018. Correcting ESL errors using phrasal SMT techniques. *21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, ACL 2006*: 249-256.

Yuan Zheng, and Ted Briscoe. 2018. Grammatical error correction using neural machine translation. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016*: 380-386.

Junpei Zhou, Chen Li, Hengyou Liu, et al. 2018. Chinese grammatical error correction using statistical and neural models. *Natural Language Processing and Chinese Computing - 7th CCF International Conference, NLPCC 2018*: 117-128.

Yongwei Zhang, Qinan Hu, Fang Liu, et al. 2018. CMMC-BDRC solution to the NLP-TEA-2018 Chinese grammatical error diagnosis task. *Proceedings of the 5th Workshop on Natural Language Proceeding Techniques for Educational Applications, NLP-TEA@ACL 2018*: 180-187.

王辰成, 杨麟儿, 王莹莹, 杜永萍, 杨尔弘. 2019. 基于Transformer增强架构的中文语法纠错方法. *The Eighteenth China National Conference on Computational Linguistics, CCL 2019*.

Samuel Bell, Helen Yannakoudakis, and Marek Rei. 2019. Context is key: grammatical error detection with contextual word representations. *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, BEA@ACL 2019*: 103-115.

Masahiro Kaneko, and Mamoru Komachi. 2019. Multi-head multi-layer attention to deep language representations for grammatical error detection. *Computacion y Sistemas Vol. 23, No. 3*: 883–891.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, et al. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019*: 4171-4186.

pyltp: the python extension for LTP. 2020. https://github.com/HIT-SCIR/pyltp. *Last accessed 21 May 2020*.

Charles A. Sutton, and Andrew McCallum. 2012. An Introduction to Conditional Random Fields. *Foundations and Trends in Machine Learning 4(4)*: 267-373.

Gaoqi Rao, Qi Gong, Baolin Zhang, et al. 2018. Overview of NLPTEA-2018 Share Task Chinese Grammatical Error Diagnosis. *Proceedings of the 5th Workshop on Natural Language Processing Techniques for Educational Applications, NLP-TEA@ACL 2018*: 42-51.

基于有向异构图的发票明细税收分类方法

赵珮瑶[1,2]，郑庆华[1,2]，董博[3,4]，阮建飞[1,2]，罗敏楠[1,2]
[1] 西安交通大学计算机科学与技术学院
[2] 陕西省天地网技术重点实验室
[3] 大数据算法与分析技术国家工程实验室
[4] 西安交通大学继续教育学院

摘要

税收是国家赖以生存的物质基础。为加快税收现代化，方便纳税人便捷、规范开具增值税发票，国税总局规定纳税人在税控系统开票前选择发票明细对应的税收分类才可正常开具发票。提高税收分类的准确度，是构建税收风险指标和分析纳税人行为特征的重要基础。基于此，本文提出了一种基于有向异构图的税收分类方法 (Heterogeneous Directed Graph Attenton Network,HDGAT)，利用发票明细间的有向信息建模，引入外部信息，显著地提高了发票明细的税收分类准确度。

关键词： 税收分类 ； 图卷积网络 ； 有向异构图

Tax Classification of Invoice Details Based on Directed Heterogeneous Graph

Peiyao Zhao[1,2]，Qinghua Zheng[1,2]，Bo Dong[3,4]，Jianfei Ruan[1,2]，Minnan Luo[1,2]
[1] School of Computer Science and Technology, Xi'an Jiaotong University
[2]SPKLSTN Lab, Xi'an Jiaotong University
[3]National Engineering Lab for Big Data Analytics, Xi'an Jiaotong University
[4]School of Continuing Education, Xi'an Jiaotong University

Abstract

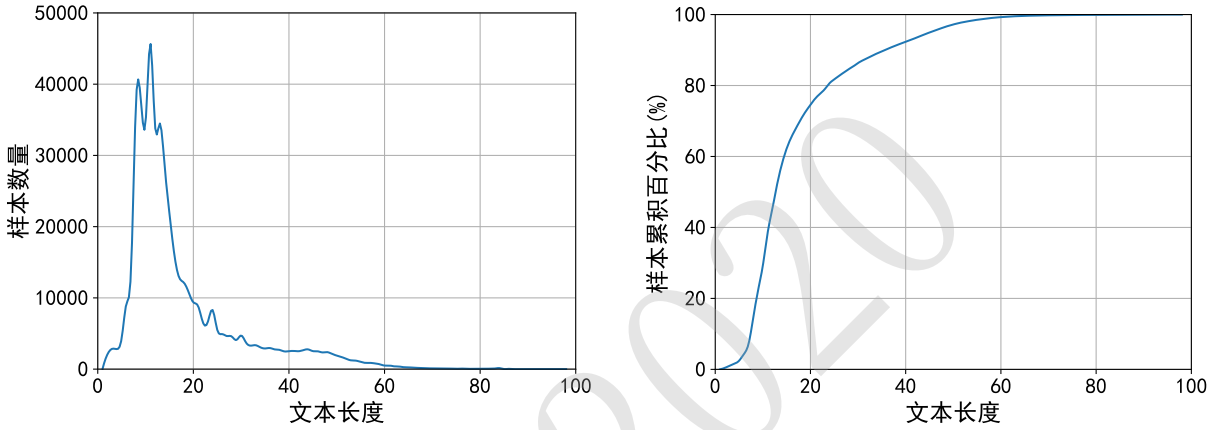
Taxation is the material basis for the survival of the country. In order to accelerate tax modernization and facilitate taxpayers to issue value-added tax invoices in a convenient and standardized manner, the State Administration of Taxation requires taxpayers to select the tax classification corresponding to the invoice details before issuing invoices in the tax control system Thus, improving the accuracy of tax classification has a crucial role in the construction of tax risk indicators and analysis of taxpayer behavior characteristics. This paper proposes a classification model based on the Heterogeneous Directed Graph Attention Network (HDGAT). The model significantly improves the accuracy of the tax classification of invoice details by using the directed information among the invoice details and external information.

Keywords: Tax classification , Graph convolutional network , Directed heterogeneous graph

1 引言

为加快税收现代化建设，方便纳税人便捷、规范开具增值税发票，国税总局自 2016 年 5 月 1 日起在全国范围内推行《商品和服务税收分类与编码》，纳税人在使用税控系统开票前，需要选择商品和服务（即发票明细）对应“税收分类编码”才能正常开具发票 (国家税务总局, 2016)。对于经济活动中不存在生产和加工环节的纳税人，购进和销售的商品原则上要保持一致，即进项货物和销项货物品类一致。通过对比商品与服务税收分类编码即可判断该类企业是否存在虚开或者开票不规范的行为。商品和服务的税收分类编码，作为各种税收风险指标的数据基础，其重要性不言而喻。提高商品和服务的税收分类精确度，是构建税收风险指标和分析纳税人行为特征的重要基础 (殷明霞, 2019)(孙懿, 2015)。

发票明细标注了对应的纳税人之间交易的商品和服务名称，选取中国某省纳税人数据进行分析，根据图（1）所示，发票明细平均文本长度为 17.62，在单词数量为 60 时，样本累计百分比达到了 98.76%，选取不同样本的统计结果有细微差别，但符合短文本的定义。因此，对针对发票明细的税收分类问题可转化为短文本的多分类问题。



(a) 样本数量统计　　(b) 样本累积百分比统计

图 1: 发票明细文本长度统计

因税务场景中发票上下游特有的方向性特征，现有的短文本分类方法并不适用于处理发票明细的税收分类问题。基于上述背景，本文提出了一种基于异构有向图的短文本分类模型 (Heterogeneous Directed Graph Attenton Network,HDGAT) 来进行发票明细的税收分类，有效地利用了发票明细间的有向信息。

2 相关工作

文本分类是自然语言处理中的经典应用，由于短文本的稀疏性和有限的标记数据，传统的文本分类方法不适用于短文本分类，目前主流方法主要包括基于主题模型的分类方法、基于深度学习的分类方法，根据数据集的不同，也可划分监督学习和半监督学习方法 (邓丁朋 et al., 2020)。

主题模型指通过推理策略获取短文本的主题特征，并将其与文档的原始特征进行融合，从而实现较好的分类效果。潜在语义分析模型（Latent Semantic Analysis，LSA）(Wiemer-Hastings et al., 2004) 通过奇异值分解将文档映射到低维语义空间里的进行向量表示，从概率生成的角度实现对文本的表示；概率潜在语义分析模型（probabilistic LSA，pLSA）(Hofmann, 1999) 从概率角度对 LSA 模型进行扩展，构建了一个以主题为隐变量的贝叶斯概率模型；隐含狄利克雷模型（Latent Dirichlet Allocation，LDA）(Blei et al., 2003) 是在 pLSA 的基础上扩展的一个文档、主题和词的三层贝叶斯概率模型，一定程度上克服了随着文档集增长 pLSA 参数过多而造成的过拟合问题。

随着神经网络的发展，RNN(Yin et al., 2017) 和 CNN(Lawrence et al., 1997) 等两个代表性的深度神经模型也开始在 NLP 任务中发挥了作用。TextCNN(Kim, 2014) 是将 CNN 用于句子分类的最初尝试，通过 word2vec 将单词转化为向量；循环神经网络 (Recurrent Neural Networks,RNN)(Yin et al., 2017) 是一类以序列数据为输入，在序列的演进方向进行递归且所有

节点按链式连接的递归神经网络；作为对 RNN 模型的改进，LSTM 网络模型 (Hochreiter and Schmidhuber, 1997) 被提出，通过长短期记忆单元来解决 RNN 梯度消失和指数爆炸问题。

针对短文本的特征词少，信息关联性不强以及存在大量样本的标注瓶颈问题，基于半监督学习的短文本分类问题越来越受到关注。预测文本嵌入 (Predictive Text Embedding,PTE) 模型 (Tang et al., 2015) 是一种半监督模型，该模型将有标签数据和无标签数据共同建模，然后将该网络降维到一个低维度的向量空间，得到文本的特征表示；分层注意力网络 (Hierarchical Attention Networks,HAN) 模型 (Yang et al., 2016) 是一种两层级的注意力模型。将文本分为单词-句子两层结构，分别建模后形成文本的向量表示。两层注意力模型分别应用于单词级别和句子级别，使得模型对于不同的单词和句子给予不同的权重，从而让文本分类更加精确；TextGCN 模型 (Yao et al., 2019) 将词和文本同时作为节点构建异构图，从而对文本进行表示；HGAT(Linmei et al., 2019) 模型通过在异构图上设立两层注意力机制，捕获不同节点的重要性。

目前金税三期的增值税发票管理系统中，会根据纳税人填写的发票明细，推荐其所属商品和服务税收分类编码，具体推荐规则为：前期通过各领域专家人工校定，得到大量有标签的样本，要推荐的发票明细通过与标签样本进行语义相似度计算，选取语义相似度最高的税收分类进行推荐。现有匹配规则忽略了发票间的关联关系以及文本间的共现关系，从而影响分类的质量。而上述短文本分类方法亦不能完全适用于税务领域，根据纳税人交易生成的发票信息可知，同一纳税人的进项发票和销项发票存在方向信息，而现有基于图的短文本分类方法是在无向图基础上建模，损失了发票的上下游信息。

本文提出了一种基于有向异构图的税收分类方法 HDGAT，通过融合标记数据和未标记数据间的关联关系，结合发票间的有向信息建模，引入标签概念和词语概念作为外部知识补充，设置双层注意力机制，通过多种粒度捕获关键信息，减少嘈杂信息的权重，显著地提高了发票明细的税收分类准确度。

3 基于有向异构图的税收分类

3.1 短文本异构图

为了充分利用发票间的上下游关系以及发票明细文本间的共现信息，本文将发票明细、标签、词语构建为有向异构图，如图（2）所示。

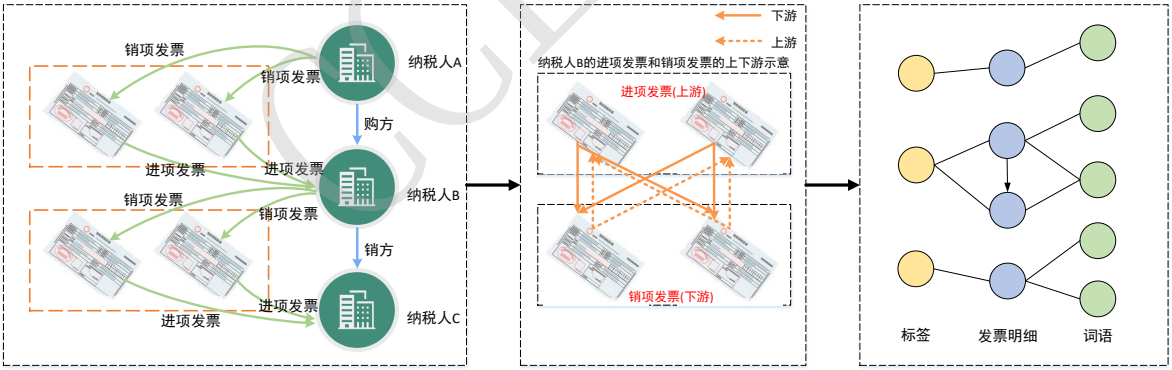


图 2: 税务异构图构建

发票间的上下游关系产生于纳税人间有向的交易关系，在图2中，纳税人 B 既可以作为与纳税人 A 交易中的购方纳税人，也可以作为与纳税人 C 交易中的销方纳税人，其交易产生的进项发票和销项发票间就存在上下游关系，可由此分析发票明细间的语义信息传递。

共现信息，指不同文本中的关联信息和特征项隐含的知识 (于游 et al., 2019)。例如，“七匹狼”这一发票明细表示的语义可以指香烟，也可以指衣物，在具体的分类任务中，如果没有其它信息的引入，很难得到正确的分类结果。若当前“七匹狼”的上下游发票明细中，含有衣物或是香烟的关联信息，则可对该文本的分类起到辅助作用。

除此外，本文考虑引入外部知识，通过处理《商品和服务税收分类编码》，提取具体类别的相关概念作为标签的外源信息；同时引用 Wikipedia 语料集，将词语对应的概念作为词语的外

源信息。

令 $\mathcal{G}=(\mathcal{V},\mathcal{E})$ 表示短文本异构图，节点 $\mathcal{V}=D\cup T\cup E$，包含短文本 $T=\{t_1,\ldots,t_m\}$，标签 $L=\{l_1,\ldots,l_k\}$ 和词语 $W=\{w_1,\ldots,w_n\}$ 。边集 $\mathcal{E}$ 代表三类节点间的关系，包括标签与文本间的关系 $Label-Text$、文本与文本间的关系 $Text-Text$、文本与词语间的关系 $Text-Word$。

短文本异构图细节描述如下。短文本间的关系通过纳税人的交易关系获得，并根据发票的方向信息得到短文本的上下游关系，因此，短文本间的边为有向边。不同的短文本可能带有税收分类标签，也可能没有税收分类标签。在 HDGAT 模型构建中，以基于谱域的方式构造有向图的拉普拉斯矩阵，利用发票明细间的上下游关系及三类实体间的关联关系对所有实体进行网络表示，包括有标签文本和无标签文本，最后对有标签的数据进行分类验证。

词语是从短文本数据预处理过程中得到的，通过分词工具，对短文本进行分词、去停用词，统计所有词语出现的次数，设定提取阈值 $\lambda=5$，提取出现次数大于 5 的词语加入异构图中。如果短文本包含某个词语，则将它们间进行连接，建立有向边。

标签来自短文本的标记信息，与短文本间的边为有向边，由标签指向短文本。

3.2 有向异构图卷积

图卷积网络 (Graph Convolutional Networks,GCN) 是一个多层神经网络，根据节点的邻域属性来推导其嵌入向量，最初应用于同构图 (Kipf and Welling, 2016)。根据更新方式可将 GCN 分为基于空间域的 GCN 和基于谱域的 GCN(Zhou et al., 2018), 本文构建的模型是以谱域的角度进行建模，模型整体架构如图（3）所示。

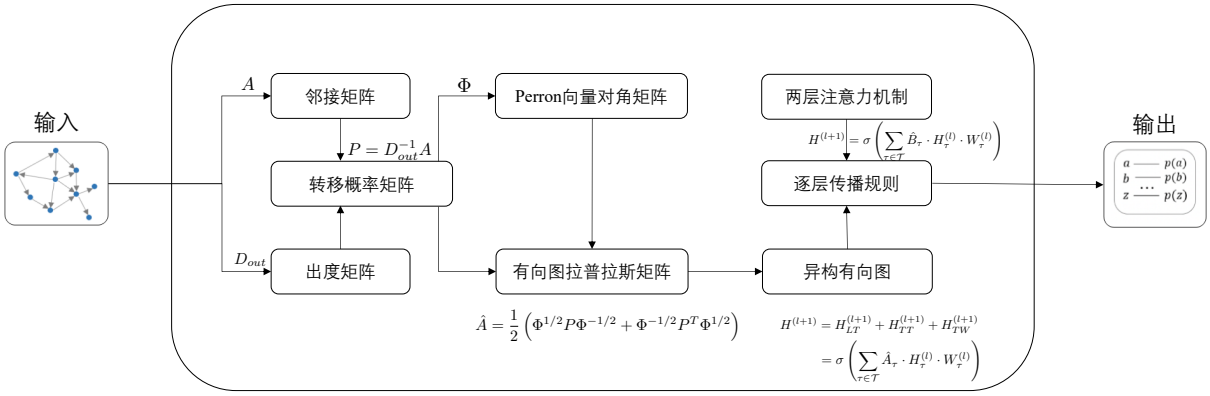


图 3: HDGAT 模型示意

定义有向图 $\mathcal{G}=(\mathcal{V},\mathcal{E})$，其中 $\mathcal{V}$ 和 $\mathcal{E}$ 分别代表节点和边的集合。如果图 G 的每对顶点之间在每个方向上都有一条路径，则此有向图 G 称为强连通。假设节点 u 指向节点 v，则这条边表示为 (u,v)，节点 u 和节点 v 互为一阶邻居。节点的出度指从该节点出发的边的条数，入度指进入该节点的边的条数。定义有向图的出度矩阵为 D_{out}，邻接矩阵为 A，则转移概率矩阵为表示为公式所示

$$P=D_{out}^{-1}A \tag{1}$$

根据 Perron-Frobenius 定理 (De Lathauwer et al., 2000)，具有非负项的不可约矩阵具有唯一的左本征向量，所有项均为正。将这一定理应用到有向图中，令 ρ 表示一个强连通有向图的转移概率矩阵 P 所有正特征向量的特征值，则 P 具有唯一的左本征向量 ϕ，其中 $\phi(v)>0$，满足等式 $\phi P=\rho\phi$，ϕ 表示行向量。根据 Perron-Frobenius 定理，令 $\rho=1$ ，而 P 的所有其它特征值的绝对值不大于 1。

定义有向图的拉普拉斯矩阵为 L_{dir}，具体表示为

$$L_{\text{dir}}=I-D^{-1/2}AD^{-1/2}=I-D^{1/2}PD^{-1/2}=I-\Phi^{1/2}P\Phi^{-1/2} \tag{2}$$

其中，$\Phi=\text{diag}\left(\phi_{\text{norm}}(v)\right)$ 为 P 的 Perron 向量对角矩阵，因上述公式（2）中, 有向图转移概率矩阵 P 为非对称矩阵，进一步修改为以下公式保证拉普拉斯算子的对称性 (Horn and

Johnson, 2012)。

$$L_{dir} = I - \frac{1}{2}\left(\Phi^{1/2} P \Phi^{-1/2} + \Phi^{-1/2} P^T \Phi^{1/2}\right) \tag{3}$$

定义有向图的分层传播规则如下

$$H^{(l+1)} = \sigma\left(\hat{A} \cdot H^{(l)} \cdot W^{(l)}\right) \tag{4}$$

其中，$\hat{A} = \frac{1}{2}\left(\tilde{\Phi}^{1/2}\tilde{P}\tilde{\Phi}^{-1/2} + \tilde{\Phi}^{-1/2}\tilde{P}^T\tilde{\Phi}^{1/2}\right)$ 表示有向图的拉普拉斯矩阵，$H^{(l)} \in \mathbb{R}^{|\mathcal{V}|\times q}$ 表示 l^{th} 层中节点的隐藏表示，第一层设置 $H^{(0)} = X$，$W^{(l)}$ 表示使用梯度下降训练神经网络权重矩阵。$\sigma(\cdot)$ 表示激活功能，例如 ReLU。

在短文本异构图中，有三类节点：短文本、词语、标签。将边类型集合表示为 $\mathcal{T} = \{\tau_1, \tau_2, \tau_3\}$，具体来说，$\tau_1 = Label - Text$，$\tau_2 = Text - Text$，$\tau_3 = Text - Word$，则对于短文本异构图的传播函数定义为各部分的加和，将它们各自的权重矩阵投影到一个隐式的公共空间中，如公式 (5) 所示:

$$H_{LT}^{(l+1)} = \sigma\left(\hat{A}_{LT} \cdot H_{LT}^{(l)} \cdot W_{LT}^{(l)}\right) \tag{5a}$$

$$H_{TT}^{(l+1)} = \sigma\left(\hat{A}_{TT} \cdot H_{TT}^{(l)} \cdot W_{TT}^{(l)}\right) \tag{5b}$$

$$H_{TW}^{(l+1)} = \sigma\left(\hat{A}_{TW} \cdot H_{TW}^{(l)} \cdot W_{TW}^{(l)}\right) \tag{5c}$$

$$H^{(l+1)} = H_{LT}^{(l+1)} + H_{TT}^{(l+1)} + H_{TW}^{(l+1)} = \sigma\left(\sum_{\tau\in\mathcal{T}} \hat{A}_\tau \cdot H_\tau^{(l)} \cdot W_\tau^{(l)}\right) \tag{5d}$$

其中 $\hat{A}_\tau \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}_\tau|}$ 是 $\hat{A}$ 的子矩阵，$\hat{A}$ 的行代表所有节点，列代表类型为 τ 的相邻节点，$H^{(l+1)}$ 的表示第 $l+1$ 层的传播函数，是通过聚集三部分 $H_\tau^{(l)}$ 的特征的信息而获得的，第一层设置 $H_\tau^{(0)} = X_\tau$。权重矩阵 $W_\tau^{(l)} \in \mathbb{R}^{q(l)} \times q^{(l+1)}.$，需考虑不同特征空间的差异。

3.3 注意力机制

通常，对于指定节点，不同类型的邻居节点可能会对它产生不同的影响。相同类型的邻居节点可能会携带更多有用的信息，并且，相同类型中的不同邻居节点也可能具有不同的重要性。例如，在短文本异构图中，短文本节点、标签节点和词语节点可能会对指定短文本节点具有不同的影响，而与该节点相连的其它短文本节点也可能具有不同的重要性。为了同时捕获节点级别和类型级别的不同重要性，本文引入双层注意力机制 (Velikovi et al., 2017)。

1）类型级别的注意力机制

给定特定节点 v，类型级别的注意将学习不同类型的相邻节点对 v 的影响权重。具体来说，首先将类型 τ 的嵌入表示为 $h_\tau = \sum_{v'} \hat{A}_{vv'} h_{v'}$，$h_\tau$ 是相邻节点特征 $h_{v'}$ 的总和，其中节点 $v' \in \mathcal{N}_v$ 并且类型为 τ。然后，基于当前节点 h_v 和嵌入表示 h_τ 来计算类型级别的注意力得分，如公式（6）所示。

$$\hat{a}_\tau = \sigma\left(\mu_\tau^T \cdot [h_v \| h_\tau]\right) \tag{6}$$

其中 μ_τ 是类型 τ 的注意力向量，$\|$ 表示“连接”，$\sigma(\cdot)$ 表示激活函数，例如 ReLU。

使用 softmax 函数对所有类型的注意力得分进行归一化，从而获得类型级别的注意力权重，如公式（7）所示。

$$\hat{\alpha}_\tau = \frac{\exp\left(\hat{a}_\tau\right)}{\sum_{\tau'\in\mathcal{T}} \exp\left(\hat{a}_{\tau'}\right)} \tag{7}$$

2）节点级别的注意力机制

对于特定节点 v, 节点级别的注意力机制将捕获不同相邻节点的重要性并减少噪声节点的权重。形式上，给定类型为 τ 的特定节点 v 及为不同类型 τ' 的相邻节点 $v' \in \mathcal{N}_v$，基于类型级别

的嵌入 τ'，计算节点 v 和节点 v' 基于节点级别的注意力得分，注意力权重表示为 $\alpha_{\tau'}$ ，如公式（8）所示。

$$\hat{b}_{vv'} = \sigma\left(\nu^T \cdot \hat{\alpha}_{\tau'}\left[h_v \| h_{v'}\right]\right) \tag{8}$$

其中 ν 是注意力向量。使用 softmax 函数对节点级别的注意力得分进行归一化，如公式（9）所示。

$$\hat{\beta}_{vv'} = \frac{\exp\left(b_{vv'}\right)}{\sum_{i\in\mathcal{N}_v}\exp\left(b_{vi}\right)} \tag{9}$$

最后，通过替换等式，我们将包括类型级别和节点级别注意的双重注意机制集成到有向异构图卷积中。整体的分层传播规则如公式所示：

$$H^{(l+1)} = \sigma\left(\sum_{\tau\in\mathcal{T}}\hat{B}_\tau \cdot H_\tau^{(l)} \cdot W_\tau^{(l)}\right) \tag{10}$$

其中，$\hat{B}_\tau$ 表示注意力矩阵，$\beta_{vv'}$ 表示在 v^{th} 行 v'^{th} 列的元素。为了实现后续的分类任务，将短文本嵌入 $H^{(L)}$ 馈送到 softmax 层进行分类，在模型训练过程中，通过 L2 范数交叉熵损失训练数据，表示为：

$$\mathcal{L} = -\sum_{i\in D_{\text{train}}}\sum_{j=1}^{C} Y_{ij}\cdot\log Z_{ij} + \eta\|\Theta\|_2 \tag{11}$$

其中，C 表示分类数目，D_{train} 表示的短文本索引集合，用于训练，Y 表示标签矩阵，Θ 表示模型参数，η 表示正则化因子。训练过程采用梯度下降算法优化模型。

4 实验设计

针对本文提出的 HDGAT 模型，本节进行相应的对比实验，说明文本间的共现信息和发票间的有向信息对于发票明细分类的有效性，同时，引入的两层注意力机制能够进一步提高模型的分类效果。

4.1 数据集描述及预处理

本文实验评测的数据集采用从我国某省国税局获取的纳税人交易数据、国税总局推出的《商品和服务税收分类编码》以及中文维基百科语料库。国税总局 2018 年发布的《商品和服务税收分类编码》，包含 4206 种商品和服务分类，每种分类都有具体的概念说明，例如，货物“甲类卷烟”，属于“烟草制品”，对应的概念描述为“每标准条 (200 支) 调拨价格在 70 元 (不含增值税) 以上 (含 70 元) 的卷烟”。概念描述为判断货物的商品和服务类别提供了依据。

中文维基百科语料库是由原始的维基百科网页处理得到，提取网页中的纯文本信息进行词模型训练，得到大量词条的向量表示，用于对短文本异构图中词语节点进行外源信息的补充。

根据我国某省国税总局中的纳税人虚开名单，根据其交易关系，选取一阶邻居节点，包含 284 种商品和服务类别的共计 50000 条发票明细文本，涉及到的纳税人数量为 4987 个，经过数据预处理后得到 12733 个词语，作为构建短文本异构图的原始数据。数据集描述如表1所示。

表 1: 数据集基本描述

数据集	文本	标签比例	词语	标签	边
TAX50K	50000	14%	12733	284	941488

短文本异构图中有向边的度分布呈幂律分布，如图4所示，大部分的节点度极小，小部分的节点度极大，且入度和出度的幂律分布有细微差别。

通过计算可知，异构图的边密度为 1.6865×10^{-5}。根据 (Goswami et al., 2018) 的研究，边密度表示图实际具有的边与其潜在边的比率，并不能很好的展现加权图的稀疏性特征，因此

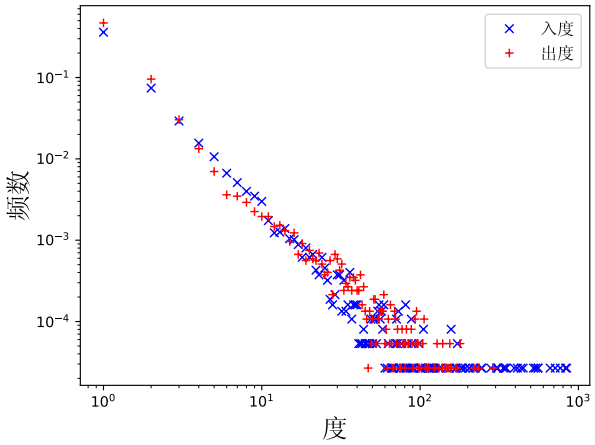


图 4: 税务异构图的度分布

引入 *Gini* 系数分析图的稀疏性。Gini 衡量一定数量的节点之间连接度的不平等程度，公式为 $GI = 1 - 2 \cdot \left[\sum_{i=1}^{n} \frac{b_i}{T}\left(\frac{n-i+\frac{1}{2}}{n}\right)\right]$，其中 T 表示所有节点度的和，b_i 表示节点的度按升序排列后第 i 个值。由表2可知，构造的税务异构图 Gini 系数为 0.6530，稀疏性较高，因此 HDGAT 通过学习局部结构的文本表示来构建发票明细的税收分类模型。

表 2: 数据集稀疏性描述

数据集	边密度	平均集聚系数	度平均值	Gini 系数
TAX50K	1.6865×10^{-5}	3.1×10^{-3}	1.7902	0.6530

表 3: 样本集数量表

总样本集 (10)	训练集 (6)	验证集 (2)	测试集 (2)
51000	31000	10000	10000

数据集按照 3:1:1 的比例划分为训练集、验证集、测试集，各样本数量如表3所示。数据集中的边共有 941488 条，其中，边类型为 $Text - Label$ 的有 50000 条，边类型为 $Text - Text$ 的有 723451 条，边类型为 $Text - Word$ 的有 168037 条。对词语补充概念文本、税收分类补充概念文本进行长度统计，统计结果如图5所示。由图可知，概念类文本长度的数值分布与发票明细的有明显区别，概念类文本长度分布不连贯，且比发票明细文本更长。

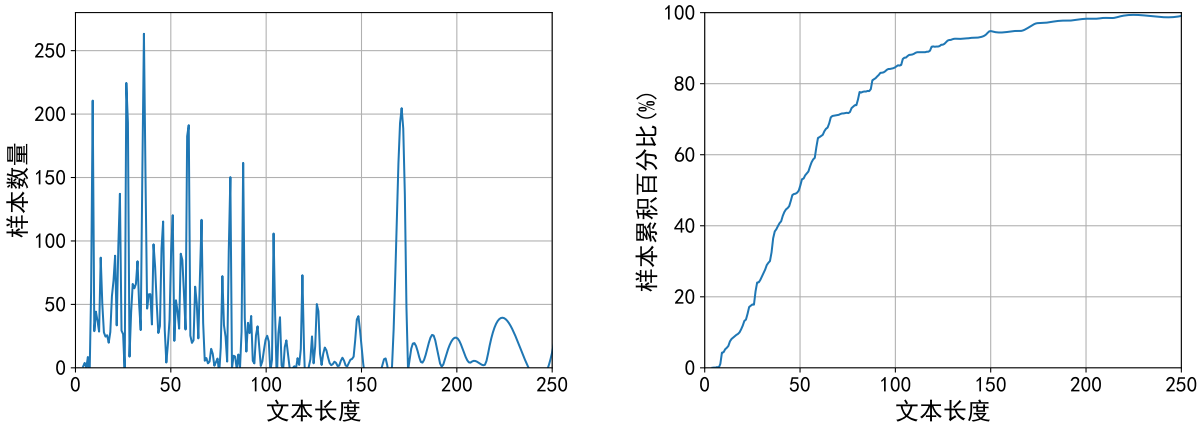
对于文本的数据预处理过程，主要包括发票明细短文本的预处理以及标签、词语的外源信息预处理两部分，描述如下。

预处理过程尝试采用 Word2Vec+TF-IDF 和 Bert 两种方法分别获取文本的向量表示，并分别进行实验对比。

1）Word2Vec + TF-IDF

Step1: 分词。根据《商品和服务税收分类编码》中对税收分类的说明构建交易明细专有词典，并根据货物名称人工添加词条作为补充，构建的交易明细专业词典包括“五金”、“浇铸”、“锡锭”、“绝地求生”、“晨光”等各行业交易明细专业性词汇共计 4467 个单词；基于 Jieba 分词工具，将交易明细专业词典设为自定义词典，作为对原有词典的补充，对发票明细短文本进行分词处理；

Step2: 去停用词。根据实际货物描述特点，将表示规格、体积的描述词添加到停用词词典中，例如：“52° 五粮液 1618 瓷瓶 500ml”中，添加“52°”、“500ml”至停用词词典，利用 Jieba



(a) 样本数量统计　　(b) 样本累积百分比统计

图 5: 概念类文本长度统计

工具对分词后的发票明细进行去停用词处理；

Step3: 获取词向量。基于 Python 的 gensim 库中的 Word2Vec 工具对纳税人的文本特征进行 Word Embedding 处理，采用 Skip-gram 模型，以词向量上下文最大距离为 5，词向量维度为 100 进行文本特征的向量化，得到词语的 Embedding 结果；

Step4: 计算 TF-IDF。TF 表示词频，词频 = 某个词出现次数/总词数；IDF 表示逆文档频率，逆文档频率 =log(语料库的文档总数/(包含该词的文档数 +1))，由 TF-IDF=TF×IDF 计算得到不同词的权重；

Step5: 获取句向量。根据词向量和词权重进行加权平均，作为该条发票明细整体的向量表示。

2)Bert

Bert 是谷歌于 2018 年提出的基于 Transformer 的双向编码器的端到端表示模型 (Devlin et al., 2018)。旨在通过联合调节所有层中的上下文来预先训练深度双向表示，并通过后期微调的训练策略提高对不同文本的向量表示能力。本文选用 Bert 旨在对比其表示效果，对细节原理不做详述。

Step1: 获取中文预训练模型，根据模型所在路径设置加载地址；

Step2: 使用 Python 的第三方库 bert-as-service，分别完成客户端 bert-serving-client 和服务端 bert-serving-server 的配置；

Step3: 服务端开启服务后，客服端根据服务端 ip 进行连接；

Step4: 依次遍历发票明细、词语补充概念、税收分类概念，根据 BertClint.encode() 函数获取对应的 768 维句向量。

4.2 评价指标和参数设置

实验任务为实体分类，选取精准率 Accuracy、F1 值作为评价指标。Accuracy 的计算方法如公式12所示，F1 的计算方法如公式13所示。

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{13}$$

其中，TP 表示正例被判定为正例，FP 表示负例被判定为正例，FN 表示正例被判定为负例，TN 表示负例被判定为负例。

针对 HDGAT 模型，我们探索了不同参数对实验结果的影响。实验基于随机梯度下降进行模型训练，在训练中，使用固定学习率 λ，在 $\lambda = 0.001, 0.002, 0.005$ 的范围内进行尝试，训练时的 dropout 在 $0.5, 0.6, 0.7, 0.8, 0.9$ 中进行尝试。最终选择如下最优参数：$\lambda = 0.005$，$dropout = 0.8$，

网络的隐藏层维度设置为 512，根据 Word2Vec+TF-IDF 和 Bert 训练得到向量维度的不同，分别将输入维度设置为 100 或 768。正则化因子设置为 $\eta = 5e-6$，避免模型过拟合。

4.3 实验结果与分析

1) 有效性分析

本节将 HDGAT 与 LSTM、PTE、TextGCN、HAN 四类基准算法在税务数据集上进行实验对比，用来评估 HDGAT 用于半监督短文本分类的效果。其中，LSTM 是长短期记忆网络(Hochreiter and Schmidhuber, 1997)，将发票明细的向量表示直接输入到模型中，进行文本分类任务；PTE(Tang et al., 2015) 是最早的异构网络的表示学习方法，可用于文本数据的半监督分类任务；TextGCN(Yao et al., 2019) 是将最早将异构 GCN 模型应用于文本分类任务的，通过将文本和词语作为节点，学习二者之间的关联，进而输出向量表示进行半监督的文本分类任务；HAN(Yang et al., 2016) 通过元路径，将文本异构图转化为几个同构子网络的加和，然后在应用图注意力机制进行文本分类任务。HGAT(Linmei et al., 2019) 将注意力机制引入异构图中进行文本分类的任务。我们用 W 表示模型输入的向量是通过 Word2Vec+TF-IDF 生成的，用 B 表示模型输入的向量是通过 Bert 生成的。“-directed”表示在 HDGAT 模型中去除有向信息，观察实验结果；“-attention”表示在 HDGAT 模型中去除双层注意力机制，观察实验结果。实验结果如表4所示。

表 4: 实验结果对比

评价指标	Accuracy(%)	F1(%)
LSTM	43.67	42.19
PTE	45.32	42.53
TextGCN	72.61	60.98
HAN	65.64	59.77
HGAT	79.49	74.12
HDGAT(W)	84.45	77.67
HDGAT(B)	**89.32**	**81.55**
HDGAT(W-directed)	79.33	72.34
HDGAT(B-directed)	80.98	73.76
HDGAT(W-attention)	73.67	61.97
HDGAT(B-attention)	75.56	62.69

由表4可知，HDGAT 模型的分类效果比基准方法有明显提升，其中 HDGAT 基于 Bert 获得向量表示比基于 Word2Vec+TF-IDF 获得向量表示对后续模型训练分类的效果更好，说明优质的词向量模型对于后续模型训练的重要性，词向量模型的训练语料越丰富，得到词语的向量表示越准确。对于基准方法的实验效果，PTE 模型的分类结果较差，原因可能是，PTE 仅依靠词语间的共现信息来学习文本嵌入的，而短文本的共现信息相对于长文本来说较少，使得 PTE 分类效果不理想。基于图神经网络的模型 HGAT、TextGCN 和 HAN 模型较 CNN、LSTM、PTE 分类效果理想。

针对 HDGAT 的消去实验的分类结果，去除有向信息的分类结果与 HGAT 分类效果基本一致，优于其它基准方法，说明引入发票明细的上下游信息捕获了短文本间的语义信息传递，同时也说明注意力机制捕获了不同相邻节点和不同类型节点的重要性（减少了嘈杂信息的权重）；去除注意力机制的分类结果仍优于基准方法，说明将发票明细间的上下游的方向信息输入到模型的有效性。

HDGAT 模型是基于谱域的图卷积模型，模型训练过程中需计算全部图的邻接矩阵，不适合超大规模图的计算，相对于基于空间域的图卷积模型，不会随机选取邻居节点进行建模，可解释性更强。

2）可解释性分析

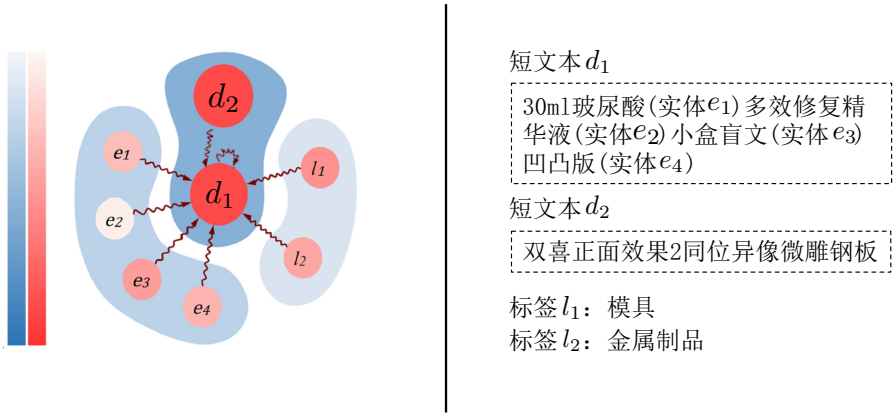


图 6: HDGAT 可视化分析

以发票明细“30ml 玻尿酸多效修护精华液小盒盲文凹凸版”为例，其正确分类为“印刷专用设备”。该发票明细通过分词、去停用词、超过阈值条件被选定关联的词语依次为“玻尿酸”、“精华液”、“盲文”、“凹凸版”。通过下游发票明细“双喜正面效果 2 同位异像微雕钢板”关联的标签为“模具”，“陶瓷玻化砖钻头”关联的标签为“金属制品”等等，构成的局部短文本异构图如图（6）所示。

通过分析纳税人间的交易信息可知，发票明细间的上下游关系可以分为通用类货物销向关系和加工类货物销向关系，通用类货物指电脑、中性笔等不局限于纳税人的经营范围而购入的货物，加工类货物指根据纳税人的经营范围购入经过加工或散装销出的货物。根据短文本 d_1 关联到的上游节点 d_2 可知，二者不属于通用类货物。d_2 作为上游节点，可能与 d_1 存在加工制作的关联性。

类型级别的注意权将高权重（0.75）分配给短文本本身，而将低权重（0.2 和 0.05）分配给实体和主题，这意味着短文本本身的语义比实体和主题对分类的贡献更大，即上游发票明细“双喜正面效果 2 同位异像微雕钢板”对于分类的影响大于其它类型节点，d_2 自身的标签为“模具”，关联为 d_1 的二阶邻居。节点级别的注意力为短文本关联的节点分配了不同的权重，属于同一类型的节点的节点级权重之和为 1，如图所示，实体 e_3（盲文）、e_4（凹凸版）的权重比 e_1（玻尿酸）、e_2（精华液）的权重更高。经过 HDGAT 模型中 softmax 层选出可能性最大的两个类别依次为“印刷专用设备”、“美容护肤品”，选定可能性最大的“印刷专用设备”作为分类结果。该发票明细的二阶邻居节点，标签 l_1（模具）和 l_2（金属制品）对于将短文本分类为“印刷专用设备”具有近似相等的影响。该案例表明，模型中引入发票明细间的有向信息及双层注意力机制可以以多种粒度来捕获关键信息，并减少嘈杂信息的权重，从而影响模型的分类结果。

5 总结

本文提出了一种基于有向异构图的税收分类方法 HDGAT。该方法通过信息传播有效利用了标记数据和未标记数据间的关联关系，构建短文本异构图，并集成标签概念和词语概念作为外源信息补充，利用了发票明细间的方向信息及双层注意力机制，有效提高了模型的分类效果。本文的实验部分利用了中国某省税务数据，证明了以下结论：

1）利用发票明细间的上下游信息，并引入双层注意力机制，通过多种粒度捕获关键信息，减少嘈杂信息的权重，有效提升了分类效果；2）通过对比基于 Word2Vec+TF-IDF 与基于 Bert 获得向量表示对后续模型训练的分类效果，说明优质的词向量模型对于后续模型训练的重要性，词向量模型的训练语料越丰富，得到词语的向量表示越准确。3）HDGAT 是一种基于谱域的图卷积模型，训练过程中需用到全部图的邻接矩阵进行计算，不适用于超大规模图计算。

该模型是根据税务场景进行构建，若其它场景中也存在短文本间具有有向信息传递的现象，可尝试进行迁移，不适用于一般类型的短文本分类任务。未来可结合空间域的图卷积模型进行改进，并从采样方式中寻求可解释性，以适应于超大规模图的计算。

致谢

本研究得到如下项目资助：国家重点研发计划“云计算与大数据”重点专项课题（2016YFB1000903），教育部创新团队 (IRT_17R8)，国家自然科学基金（61721002、61532015），西安交通大学-税友集团人工智能联合实验室项目。

参考文献

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. 2000. A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications*, 21(4):1253–1278.

Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Swati Goswami, CA Murthy, and Asit K Das. 2018. Sparsity measure of a network graph: Gini index. *Information Sciences*, 462:16–39.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. *international acm sigir conference on research and development in information retrieval*, 51(2):50–57.

Roger A Horn and Charles R Johnson. 2012. *Matrix analysis*. Cambridge university press.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Steve Lawrence, C Lee Giles, Ah Chung Tsoi, and Andrew D Back. 1997. Face recognition: A convolutional neural-network approach. *IEEE transactions on neural networks*, 8(1):98–113.

Hu Linmei, Tianchi Yang, Chuan Shi, Houye Ji, and Xiaoli Li. 2019. Heterogeneous graph attention networks for semi-supervised short text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4823–4832.

Jian Tang, Meng Qu, and Qiaozhu Mei. 2015. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1165–1174.

Petar Velikovi, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2017. Graph attention networks.

Peter Wiemer-Hastings, K Wiemer-Hastings, and A Graesser. 2004. Latent semantic analysis. In *Proceedings of the 16th international joint conference on Artificial intelligence*, pages 1–14. Citeseer.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7370–7377.

Chuanlong Yin, Yuefei Zhu, Jinlong Fei, and Xinzheng He. 2017. A deep learning approach for intrusion detection using recurrent neural networks. *Ieee Access*, 5:21954–21961.

Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2018. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434*.

于游, 付钰, and 吴晓平. 2019. 中文文本分类方法综述. 网络与信息安全学报, (5).

国家税务总局. 2016. 国家税务总局关于开展商品和服务税收分类与编码试点工作的通知. [EB/OL]. http://www.chinatax.gov.cn/n810341/n810755/c2417702/content.html Accessed Feb 25, 2016.

孙懿. 2015. 大数据时代对税务工作的挑战与对策. 学术交流, (2015 年 06):133–139.

殷明霞. 2019. 基于金税三期下的企业税务风险管理研究. 纳税, (33).

邓丁朋, 周亚建, 池俊辉, and 李佳乐. 2020. 短文本分类技术研究综述. 软件.

半监督跨领域语义依存分析技术研究

毛达展　　李华勇　　邵艳秋*
国家语言资源监测与研究平面媒体中心，信息科学学院
北京语言大学，北京市海淀区学院路 15 号，100083，中国
maodazhan@foxmail.com　lihuayong@blcu.edu.cn　yqshao163@163.com

摘要

近年来，尽管深度学习给语义依存分析带来了长足的进步，但由于语义依存分析数据标注代价非常高昂，并且在单领域上性能较好的依存分析器迁移到其他领域时，其性能会大幅度下降。因此为了使其走向实用，就必须解决领域适应问题。本文提出一个新的基于对抗学习的领域适应依存分析模型，我们提出了基于对抗学习的共享双编码器结构，并引入领域私有辅助任务和正交约束，同时也探究了多种预训练模型在跨领域依存分析任务上的效果和性能。

关键词： 语义依存分析 ； 领域适应 ； 对抗学习 ； 预训练模型

Semi-supervised Domain Adaptation for Semantic Dependency Parsing

Dazhan Mao　　Huayong Li　　Yanqiu Shao*
Language Resources Monitoring and Research Center,
Information Science School, Beijing Language and Culture University,
15 Xueyuan Road, HaiDian District, Beijing, 100083, China
maodazhan@foxmail.com　lihuayong@blcu.edu.cn　yqshao163@163.com

Abstract

Recently, although deep learning has brought significant progress to semantic dependency parsing, the semantic annotation data is very expensive to label, and when a dependency parser with better performance in a single domain is migrated to other domains, its performance will decline largely. Therefore, in order to make it practical, it is necessary to solve the problem of domain adaptation. This paper proposes a new domain adaptation dependency parsing model based on adversarial learning. We proposed a shared dual encoder structure based on adversarial learning, and introduced domain private auxiliary tasks and orthogonal constraints. At the same time, we also explored a variety of pre-training models in the cross domain dependency parsing task about the effectiveness and performance.

* 通讯作者　Corresponding Author

Keywords: Semantic dependency parsing , Domain adaptation , Adversarial learning , Pre-training model

1 引言

依存分析是一种句子结构的解析方式，其将句子的句法或语义结构解析为一系列二元、非对称依存关系，这些依存关系构成了句子的依存树（或依存图）。不同于句法依存树分析，语义依存图分析是一种深层次的语义解析，其描述的是句子各个组成部分间的语义关系 (Che et al., 2012)，如图 1 所示，其允许更复杂的依存结构（如多父节点、非投射等等）。由于其能够直接表达深层语义信息，因此应用价值更大。然而，现有的语义依存分析研究使用的数据集往往来自课本或者新闻等单个领域，这样即使依存分析器在数据集上取得了较高的性能，当迁移到其他目标领域时，分析器的性能也会大幅度下降。

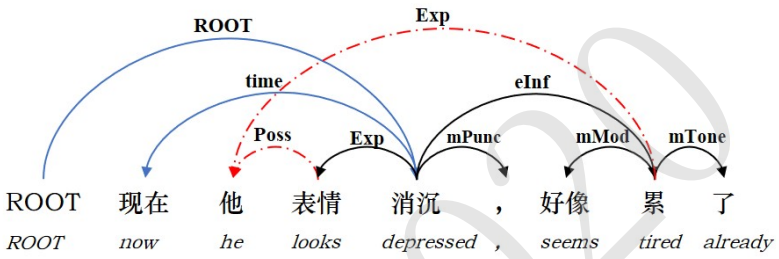


图 1. 语义依存分析示例。图中红色依存弧为多父节点现象，蓝色依存弧为非投射现象

根据目标领域的数据有无标注，领域适应可以划分为无监督领域适应（目标领域完全没有标注数据）和半监督领域适应（目标领域存在少量标注数据，同时也有大量无标注数据）(Kouw and Loog, 2018)。由于语义依存分析本身的复杂性，目前纯粹基于无监督的跨领域语义依存分析的研究进展相对滞后。而半监督的领域适应虽然仍需要少量的数据标注，但是其可以利用一定的监督信号指导领域适应，领域迁移效果更好，迁移后的模型实用价值更大，也能更好地和语义依存分析任务结合。因此本文关注于针对语义依存分析任务的半监督领域适应。本文的主要工作总结如下：

- 本文提出了一个新的基于对抗学习的领域适应框架。该框架支持一个模型同时解决面向多个目标领域的领域适应问题。该框架在实验数据集上明显优于基线模型。
- 本文将预训练语言模型融合到了对抗领域适应框架中，从而进一步提升了模型的领域适应能力。同时我们详尽讨论分析了应用预训练语言模型解决语义依存分析任务以及领域适应时的一系列细节问题。

2 相关工作

2.1 依存分析

现有的依存分析方法主要有两种，分别是基于转移的算法 (Chen and Manning, 2014);(Dyer et al., 2015) 和基于图的算法 (Chen et al., 2013);(Wang and Chang, 2016)。早期的这两种依存

分析器需要手动定义复杂的特征模板，这费时费力且需要很强的背景知识，限制了分析器的进一步发展 (Koo and Collins, 2010);(Koo and Collins, 2010)。

近年来，神经网络方法被广泛应用在依存分析中 (Chen and Manning, 2014);(Dozat et al., 2017)。在这些基于神经网络的依存分析器的研究工作当中，(Dozat and Manning, 2016) 双仿网络依存分析器取得了目前最优的性能。因此，双仿网络依存分析器在本文中，将作为后续对依存分析进行领域适应研究的基础。

2.2 领域适应

最近，随着 (Peters et al., 2018)ELMO;(Devlin et al., 2018)BERT 等上下文表示的兴起，大量工作开始研究基于预训练上下文表示的领域适应方法，并且取得了较好的结果，展示了预训练上下文表示在领域适应任务上的巨大潜力。(Liu et al., 2019a) 分析了上下文表示中的语言学知识和可迁移性。(Mulcaire et al., 2019) 使用上下文表示提升了跨语言任务的迁移效果。受到这些工作的启发，本工作将把预训练模型融入依存分析的领域适应模型中，探究上下文信息对跨领域依存分析是否有帮助。

对抗学习已经被证明可以明显提升跨领域依存分析器的性能 (Bousmalis et al., 2016);(Ganin and Lempitsky, 2014)。但是大部分的工作为了抽取不同领域之间的无关特征，都是把领域无关的特征和领域私有的特征混合在一起，这就不可避免地损失一些领域私有的信息 (Sato et al., 2017)。(Chen et al., 2017) 针对中文多粒度分词任务，提出了一个 Shared-Private 模型。在这个模型的基础上，本文对私有编码器进行简化，不同领域的私有编码器统一成一个，并增加领域预测的辅助任务。(Liu et al., 2017);(Shi et al., 2018) 引入了正交约束来消除共享空间和私有空间之间的冗余信息。在本文中也将把正交约束应用到领域无关编码器和领域私有编码器之间。

3 基于对抗学习的领域适应依存分析模型

与一般基于对抗的跨领域依存分析做法一样，都是混合源领域和目标领域的数据输入到 Biaffine 编码器，但本模型增加了 BERT 通用编码层、领域共享双编码器、领域分类辅助任务以及正交约束等可能对模型性能有提升作用的组件。

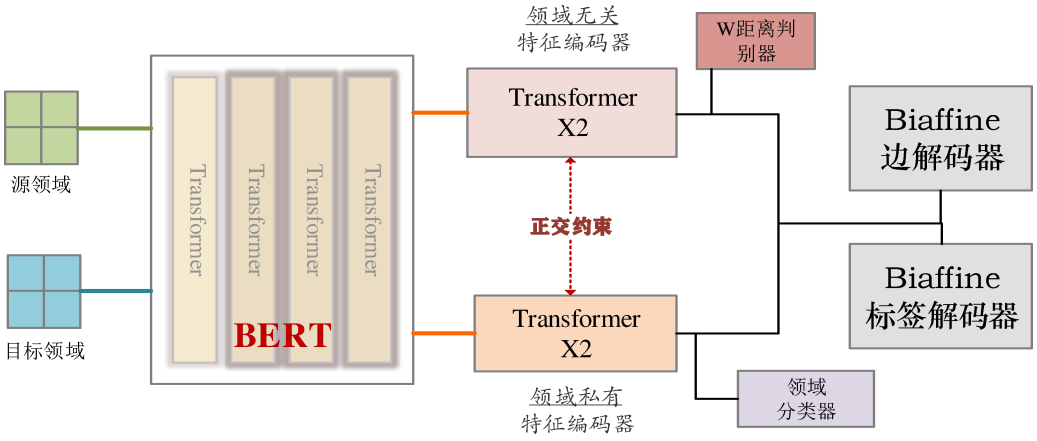


图 2. 基于对抗学习的领域适应依存分析模型结构

3.1 BERT 通用编码层

经典的依存分析器采用词向量加词性向量的静态表征，有时也会以字符向量表示加以辅助，这种经典的组合方式不能为每个词提供基于上下文的正确表示，也无法很好地解决未登录词问题。近年来，随着 BERT 等预训练语言模型的涌现，越来越多的研究开始使用预训练语言模型替换经典的词向量输入，同时也有大量研究表明 BERT 等预训练语言模型对于跨领域迁移有着很好的帮助，因此本文使用 BERT 作为底层编码。BERT 是多层 Transformer 神经网络 (Vaswani et al., 2017) 的堆叠，形式化地，BERT 每层的处理过程可表示为：

$$h_{i,j} = BERT_j(x_i) \tag{1}$$

其中，i 表示第 i 个输入，j 表示第 j 层 BERT，x_i 是输入的字符。

BERT 默认选择使用最后一层 BERT 输出作为整体输出，但是已有大量研究表明 BERT 等预训练语言模型每层的编码信息并不相同，一般 BERT 底层涉及一些语言基础知识，BERT 中层编码了一定的句法结构知识，BERT 高层则编码了语义知识，且 BERT 和训练时的任务相关度很高。因此直接使用最后一层 BERT 输出可能不是最好的方案，为此，本文引入了层加权机制，以一种可训练的方式加权平均不同 BERT 层的输出。层加权机制可形式化为：

$$h_i = c\sum_i BERT_{j,i} \cdot softmax(w_j) \tag{2}$$

其中 w_j 是一个可训练的“权重”标量，用来对应每一层 BERT 输出；c 是一个可训练的缩放标量，用了缩放最后的加权表示；$BERT_{j,i}$ 表示第 j 层 BERT 在第 i 个位置的输出。

经过层加权机制后，可以得到对应输入的字符序列表示，由于依存分析是基于词语级别的，所以需要从字符序列映射到词语序列，我们采用简单的尾字表示法完成映射，即对于每个词语只选择尾字对应的表示来作为整个词语的表示。

3.2 领域共享双编码器

在预训练语言模型之后，又连接了两个领域共享编码器，一个是领域无关特征编码器 $f_{share}^{E(x)}$，一个是领域私有信息编码器 $f_{private}^{E(x)}$，分别负责提取领域无关特征和领域私有特征。两个编码器均使用两层 Transformer 神经网络实现，每层 Transformer 网络可以形式化地表示为：

$$Transformer(X) = Skip(FF, Skip(MultiHead, X)) \tag{3}$$

$$Skip(f, h) = LayerNorm(h + Dropout(f(h))) \tag{4}$$

$$FF(h) = GELU(hW_1^T + b_1)W_2^T + b_2 \tag{5}$$

其中，(Hendrycks and Gimpel, 2016)GELU 代表高斯误差线性单元激活（Gaussian error linear units）函数。为了保证领域无关特征编码器可以提取到领域共享的特征，我们在领域无关编码器上额外连接了一个对抗判别器，基于对抗学习的方式强制编码器编码领域无关特征。同时为了保证领域私有编码器可以提取到每个领域的私有信息，我们在领域私有编码器上也额外连接了一个领域分类辅助任务。

3.3 对抗判别器

“领域无关”特征编码器除了连接依存任务所需的解码器 $Biaffine^{edge}$ 和 $Biaffine^{label}$ 外，还额外连接一个对抗判别器 $D_{adv}(x)$，负责提取领域之间的不变特征。

参考 WGAN 的实现 (Arjovsky et al., 2017);(Arjovsky and Bottou, 2017)，本文采用基于 Wasserstein 距离的对抗判别器。在使用基于 Wasserstein 距离的损失作为对抗损失时，对抗判别器实际上是一个 Wasserstein 距离回归网络。

形式化地，对于源领域的输入数据 X_{source} 和目标领域的输入数据 X_{target}，经过领域特征编码器后，我们分别得到对应的表示分布 P_s 和 P_t，则 P_s 和 P_t 之间的 Wasserstein 距离等于：

$$W(P_s, P_t) = \sup_{\|f\|_L \leq 1} E_{x\sim P_s}[f(x)] - E_{x\sim P_t}[f(x)] \tag{6}$$

其中，f 是一个 Lipschitz-1 连续函数，注意为了求解 Wasserstein 距离，这里依据 WGAN 对其定义公式做了转换。根据 WGAN 中的要求，我们使用一层全连接神经网络 f^W 近似该 Lipschitz-1 连续函数，同时将该网络的参数取值范围固定到 [-0.01,0.01] 之间。

进而可以计算得到 Wasserstein 距离对抗损失 L^W_{adv}：

$$L^W_{adv}(S^s, S^t) = f^W(S^s) - f^W(S^t) \tag{7}$$

在训练时，一方面我们需要优化“判别器”以产生最准确的 Wasserstein 距离，为此需要在“判别器”的参数上最小化 Wasserstein 距离对抗损失 $L^W_{adv}(S^s, S^t)$；另一方面，本文需要使领域无关特征编码器产生的两个表示分布尽可能“迷惑”Wasserstein 距离判别器，为此，本文需要在领域无关特征编码器的参数上最大化 Wasserstein 距离对抗损失 $L^W_{adv}(S^s, S^t)$。

由上述可知基于 Wasserstein 距离的对抗学习过程是一个 minmax 训练，即：

$$\min_{\Theta^{dis}} \max_{\Theta^{share}} L^W_{adv} \tag{8}$$

其中，Θ^{dis} 表示判别器的参数，Θ^{share} 表示判别器的参数。

在训练时我们通过先进行 $\min_{\Theta^{dis}}$ 训练，然后再进行 $\max_{\Theta^{share}}$ 训练的方式交替完成整个训练过程

3.4 Biaffine 解码层

本文使用双仿解码器来分别预测两个词语之间的依存弧关系和依存标签。首先将编码输出的词语级别的表示向量 h_i^{lstm} 传入两个前馈神经网络层 (FNN)，分别得到该词语的“头表示“和“尾表示”：

$$h_i^{edge-head} = FNN^{edge-head}(h_i^{lstm}) \tag{9}$$

$$h_i^{edge-dep} = FNN^{edge-dep}(h_i^{lstm}) \tag{10}$$

随后使用双仿变换整个句子中可能存在的依存弧的得分矩阵 $s_{i,j}^{edge}$：

$$Biaffine(x_1, x_2) = x_1^T U x_2 + W(x_1 \otimes x_2) + b \tag{11}$$

$$s_{i,j}^{edge} = Biaffine^{edge}(h_i^{edge-dep}, h_j^{edge-head}) \tag{12}$$

$$p_{i,j}^{edge} = sigmoid(s_{i,j}^{edge}) \tag{13}$$

训练时，依存弧解码器的损失定义为：

$$J_{edge}(\Theta^p) = -p_{i,j}^{edge} \log p_{i,j}^{edge} - (1 - p_{i,j}^{edge}) \log(1 - p_{i,j}^{edge}) \tag{14}$$

依存标签的方式和预测依存弧的方式非常相似，唯一不同的就是两个词语之间的依存标签的分类空间比较大，因此这里使用 softmax(Grave et al., 2016) 而不是 sigmoid 函数处理，最终得到依存标签概率 $p_{i,j}^{label}$。

$$p_{i,j}^{label} = softmax(s_{i,j}^{label}) \tag{15}$$

训练时，依存标签解码的损失定义为：

$$J_{label}(\Theta^p) = -\sum_{label} \log p_{i,j}^{label} \tag{16}$$

最后将依存弧概率和依存标签概率传给解码算法，就能得到最后的依存图。

在训练时，通过最小化依存损失 $J_{parser}(\Theta^p)$ 从而训练得到一个领域内依存分析器，依存分析损失由依存弧损失和依存标签损失相加得到：

$$J_{parser}(\Theta^p) = \beta J_{label}(\Theta^p) + (1 - \beta) J_{edge}(\Theta^p) \tag{17}$$

其中，β 是一个超参数，用来控制最终损失中两个解码器损失的相对大小。

3.5 领域分类辅助任务

我们希望私有编码器能够提取领域私有的信息，但仅通过最小化依存任务的损失 L_{parser} 无法保证私有特征编码器真正提取到了对应领域的私有信息，因此本工作又额外引入了一个私有辅助任务，即领域分类任务，负责判断编码器编码的特征属于哪一个领域。这一辅助任务类似于文本领域分类，由一个领域分类器 $f^c(x)$ 实现，其包括一层全连接神经网络和一个 softmax 层：

$$f^c(p^T, \theta_C) = softmax(b + UP^T) \tag{18}$$

其中，b 和 U 代表全连接层的参数，P 代表私有信息编码层 $f_{private}^E$ 的输出特征。

训练时，领域分类器的交叉熵损失 $L_{classify}$ 定义为：

$$L_{classify} = -\sum_{i=1}^{N} \sum_{j=1}^{2} y_i^j \log(\hat{y}_i^j) \tag{19}$$

其中，$\hat{y}_i^j$ 为 softmax 层的预测标签，$\hat{y}_i^j$ 为真实标签。

通过通过最小化 $L_{classify}$，可以迫使领域私有特征编码器编码对应领域的私有特征。

3.6 正交约束

加入辅助任务后可以保证领域私有特征编码器学习到了领域的私有信息，但是私有特征编码器可能会学习到一部分领域无关特征，造成特征冗余表达。为了确保这两个编码器之间不存在冗余的特征，本工作为两个编码器之间增加了一个正交约束，在训练时惩罚领域私有编码器中和“领域无关”编码器重合的特征，从而促使领域私有信息编码器不提取领域间的不变特征。正交约束损失的定义如下：

$$L_{diff} = \|S^T P\|_F^2 \tag{20}$$

这里 S 代表领域无关编码器 f^E_{share} 的输出，P 代表领域私有信息编码层 $f^E_{private}$ 的输出，$\|\cdot\|^2_F$ 代表平方 Frobenius 范数。

矩阵 A 的 Frobenius 范数 $\|A\|_F$ 定义为：

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{i,j}|^2} \tag{21}$$

由上述公式可知，Frobenius 范数代表了矩阵的所有元素平方和的开方。因此，通过最小化正交约束 L_{diff}，就迫使 S^TP 的乘积最小化，进而等价于迫使两个矩阵相互“正交”，而从使得两个编码器的输出特征互不重叠。

3.7 联合训练

通过将上述的所有任务损失整合起来，得到了总共 4 个损失，分别是对抗损失 L^C_{adv}（或者 L^W_{adv}）、依存分析任务损失 L_{parser}、领域私有信息编码层辅助任务损失 $L_{classify}$、领域无关信息编码器和领域私有信息编码器之间的正交损失 L_{diff}。我们定义最终的训练目标损失 L 为：

$$L = L_{parser} + \lambda L_{adv} + \gamma L_{classify} + \eta L_{diff} \tag{22}$$

其中，依存分析的任务损失定义为：

$$L_{parser}(\Theta^p) = \beta L_{label}(\Theta^p) + (1-\beta)L_{edge}(\Theta^p) \tag{23}$$

上述 β、λ、γ、η 均为控制损失大小的超参数。注意，当使用目标领域的无标注数据时，L_{parser} 只在源领域的数据上计算。

4 实验部分

4.1 数据集介绍

本研究的源领域数据集来自 the SemEval-2016 task9(Che et al., 2012) 和《博雅汉语》。经过调研，选择两大类四小类目标领域，一大类是文学风格，主要包括散文（《文化苦旅》）、小说（《小王子》、《少女小渔》）、剧本（《武林外传》）三个子目标领域。另一大类是下游应用，主要包括医疗诊断文本子目标领域。

依据中文语义依存图标注规范，依托语义依存图标注平台，我们组织了 6 名语言学专业的学生做了数据标注。对于每个目标领域，我们只标注了少部分数据，并将其划分为训练集、验证集、测试集，并对剩余的无标注数据做了清洗和筛选，如表 1 所示。

表 1. 数据集划分

领域说明			人工标注数据集			无标注数据
			训练集	验证集	测试集	
源领域		平衡语料	38000	2000	2000	0
目标领域	文学	散文	3000	1000	1000	20000
		小说	3000	1000	1000	30000
		剧本	3000	1000	1000	8000
	应用	医疗	2000	500	500	30000

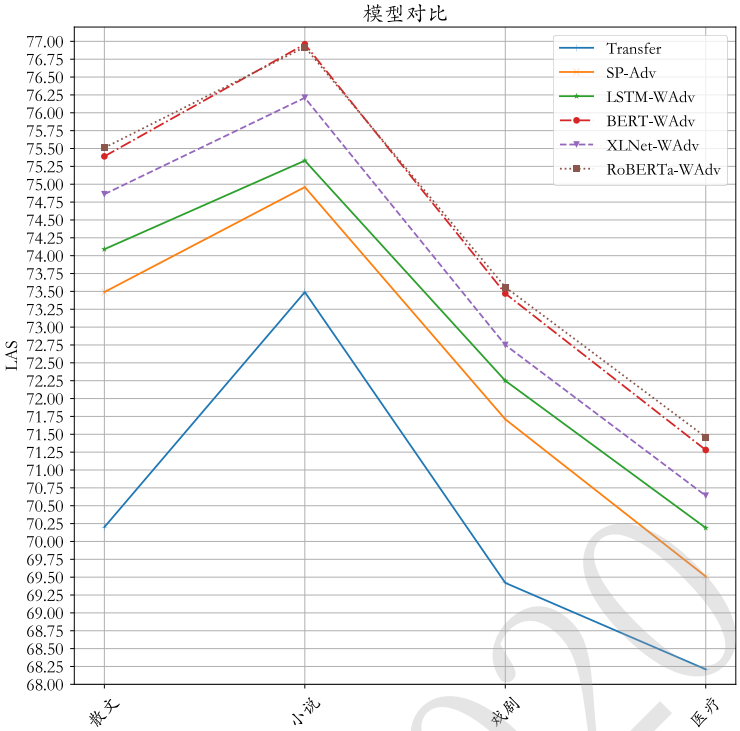


图 3. 本工作的模型和基线模型的对比

4.2 实验设置

我们尝试了多种预训练语言模型，其层数均为为 12，隐层向量维度均为 768。领域私有特征编码器和领域无关特征编码器都使用两层 Transformer 神经网络，其中 Transformer 层的注意力头数为 8，隐层向量维度为 768，dropout 比例为 0.2，使用 Relu 激活函数。对抗损失的控制参数 λ 为 0.5；领域分类辅助任务损失的控制参数 γ 为 0.05；正交约束损失的控制参数 η 为 0.001；依存损失的控制参数 β 为 0.5。对抗判别器的学习率设置为 0.0001，模型的其他部分的学习率设置为 0.001。在训练时使用带 L2 正则的 Adam 优化算法，min 训练和 max 训练的交替比例为 5:1。输入最大句长为 100，超过此长度的句子将被跳过。本文使用 4 张 NVIDIA Tesla V100-16GB 的显卡完成训练，单卡的批量大小设置为 32。

4.3 基线模型

为了更好地比较提出的模型的领域适应能力，我们选择了两个基线模型，分别是迁移模型**Transfer**和基于领域分类对抗损失的“共享 -私有”模型 **SP-Adv**：

- **Transfer**：Transfer 使用基于 LSTM+Biaffine 的单领域依存分析模型，在训练时，Transfer 模型先在源领域的数据上预训练，然后再在对应的目标领域上进一步微调。
- **SP-Adv**：模型使用经典的“共享 -私有”框架，同样使用对抗训练，但是其不采用正交约束，也不采用领域预测的辅助任务。

此外，为了进一步对比基于预训练语言模型的动态表征和传统的基于词向量的静态表征之

间的差别，我们将预训练语言模型替换为词向量加词性向量，模型其他部分保持不变，得到另一个基线模型，称为 **LSTM-WAdv**。

4.4 实验结果

4.4.1 与基线模型的对比

表 2 展示了我们的模型和基线模型在 4 个目标领域上的 LAS 指标，其中 **Transfer**、**SP-Adv**分别代表两个基线模型的结果，**LSTM-WAdv**代表在本文提出的模型上去掉预训练语言模型之后的结果，**BERT-Wadv**(Devlin et al., 2018);**XLNet-WAdv**(Yang et al., 2019);**RoBERTa-WAdv**(Liu et al., 2019b) 分别代表使用 **BERT**、**XLNET**、**RoBERTa**预训练语言模型的结果。

为了更加直观地比较差异，我们绘制了模型之间的对比折线图（如图 3），由图 3 可以看出，我们提出的基于预训练语言模型和对抗学习的领域适应框架都明显优于两个基线模型。同时使用预训练语言模型的领域适应框架也要优于使用词向量的框架。同时在三种预训练语言模型中，**RoBERTa**展现了最好的领域适应性能。

表 2. 本工作的模型和基线模型在 4 个目标领域上的 LAS 指标

模型	散文	小说	戏剧	医疗
Transfer	70.20	73.49	69.42	68.21
SP-Adv	73.49	74.96	71.71	69.51
LSTM-WAdv	74.09	75.33	72.25	70.19
BERT-WAdv	75.39	**76.96**	73.47	71.28
XLNet-WAdv	74.86	76.21	72.75	70.64
RoBERTa-WAdv	**75.51**	76.92	**73.56**	**71.46**

4.4.2 无标注数据对领域适应的影响

为了进一步探索无监督数据量在半监督学习中的影响，我们又做了两组对比实验。这两组实验分别选择前述实验中 LAS 最高的小说目标领域和 LAS 最低的医疗目标领域。本文将这两个领域的所有无标注数据划分为相等的 10 份，从不使用无标注数据到使用全部的无标注数据，逐步增加无标注数据的数量训练模型，并记录对应的 LAS 指标。如图 3 所示，无论是医疗领域还是小说领域，LAS 指标都随着无标注数据量的增加呈现接近线性关系的增长。注意，在小说领域上，当无标注数据使用超过七成的时候，LAS 指标的提升已经非常微弱，这说明此时两个编码器已经基本收敛，无法进一步提升。

4.4.3 消融实验

为了进一步分析本文提出的不同组件对最终模型领域适应性能的影响，我们在 **LSTM-WAdv**的基础上又做了相应的消融实验，如表 3 所示，分别记录了去掉对抗损失、去掉正交约束、去掉领域预测辅助任务以及去掉私有特征编码器时的实验结果。

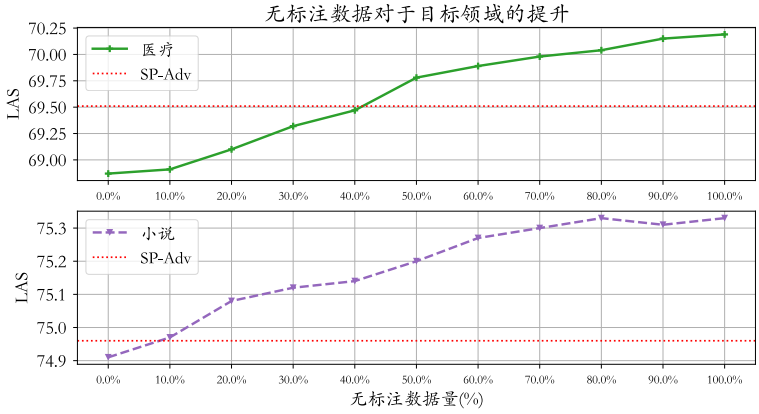


图 4. 无标注数据量对领域适应的影响

表 3. 消融实验

实验	散文	小说	剧本	医疗	平均下降
LSTM-WAdv	74.09	75.33	72.25	70.19	——
去掉对抗	72.82	74.90	71.10	69.48	0.890
去掉正交约束	73.90	75.16	71.81	69.84	0.288
去掉辅助任务	73.62	75.20	72.15	69.91	0.245
去掉私有特征	73.41	75.01	71.60	69.74	0.525

从表中可以看出，以上四个组件中，对模型最终效果影响最大的是对抗损失，去掉其之后模型在 4 个目标领域上平均 LAS 下降了 0.89，这再次证明了对抗学习技术在领域适应任务中的重要作用；其次影响模型性能的组件是私有特征，去掉其之后模型 LAS 平均下降了 0.525，这里需要注意一旦去掉私有编码器，正交约束和辅助任务也相应地失去了作用，因此私有特征的影响要大于其他两个组件。同时从表中可以看出，四个组件均对模型最终的性能有积极作用，其中影响最小的辅助任务也有 0.245 的平均共享。上述实验充分证明了本章提出的模型方法是有效的。

5 结论

在之前提到的跨领域分析数据集上，本文提出的基于预训练语言模型和对抗学习的领域适应框架都明显优于两个基线模型，在尝试的三种预训练模型中，RoBERTa 展现了最好的领域适应性能。在消融实验中，也验证了本文提出的领域适应框架的各个组件对模型最终性能的提升是有积极作用的。

致谢

本成果受国家自然科学基金项目（61872402），教育部人文社科规划基金项目 (17YJAZH068)，北京语言大学校级项目（中央高校基本科研业务费专项资金）（18ZDJ03），模式识别国家重点实验室开放课题基金资助。

参考文献

Martin Arjovsky and Leon Bottou. 2017. Towards principled methods for training generative adversarial networks. *Stat*, 1050.

Martin Arjovsky, Soumith Chintala, and Leon Bottou. 2017. Wasserstein gan.

Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. 2016. Domain separation networks. *CoRR*, abs/1608.06019.

Wanxiang Che, Meishan Zhang, Yanqiu Shao, and Ting Liu. 2012. Semeval-2016 task 9: Chinese semantic dependency parsing. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*.

D. Chen and C. D. Manning. 2014. A fast and accurate dependency parser using neural networks.

Wenliang Chen, Zhang Min, and Haizhou Li. 2013. Utilizing dependency language models for graph-based dependency parsing models. In *Meeting of the Association for Computational Linguistics: Long Papers*.

Xinchi Chen, Zhan Shi, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-criteria learning for Chinese word segmentation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1193–1203, Vancouver, Canada, July. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing.

Timothy Dozat, Peng Qi, and Christoper Manning. 2017. Stanford's graph-based neural dependency parser at the conll 2017 shared task. pages 20–30, 01.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. 2015. Transition-based dependency parsing with stack long short-term memory. *Computer Science*, 37(2):321–C332.

Yaroslav Ganin and Victor Lempitsky. 2014. Unsupervised domain adaptation by backpropagation.

Edouard Grave, Armand Joulin, Moustapha Cisse, David Grangier, and Herve Jegou. 2016. Efficient softmax approximation for gpus.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus).

Terry Koo and Michael Collins. 2010. Efficient third-order dependency parsers. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*.

Wouter M. Kouw and Marco Loog. 2018. An introduction to domain adaptation and transfer learning.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-task learning for text classification.

Nelson F Liu, Matt Gardner, Yonatan Belinkov, Matthew E Peters, and Noah A Smith. 2019a. Linguistic knowledge and transferability of contextual representations.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach.

Phoebe Mulcaire, Jungo Kasai, and Noah A. Smith. 2019. Polyglot contextual representations improve crosslingual transfer. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3912–3918, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations.

Motoki Sato, Hitoshi Manabe, Hiroshi Noji, and Yuji Matsumoto. 2017. Adversarial training for cross-domain universal dependency parsing. In *Conll Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*.

Ge Shi, Chong Feng, Lifu Huang, Boliang Zhang, and Heyan Huang. 2018. Genre separation network with adversarial training for cross-genre relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Wenhui Wang and Baobao Chang. 2016. Graph-based dependency parsing with bidirectional lstm. In *Meeting of the Association for Computational Linguistics*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding.

汉英篇章衔接对齐语料库构建研究

李艳翠*　冯继克　来纯晓　冯洪玉
河南科技学院信息工程学院，新乡 453003
liyancui@hist.edu.cn

摘要

篇章衔接性分析是理解篇章的基础，汉语和英语在指代、连接和省略等主要衔接方式上存在差异。本文旨在创建汉英篇章衔接对齐语料库，给出包括子句、连接词、指代和省略的汉英篇章衔接对齐标注策略，建立包含相应信息的对齐信息的语料库资源，最后对标注语料进行评估并讨论了标注中的难点问题及解决方法。对语料库标注质量评估及简单实验结果表明，本文研究语料标注策略方法切实可行，所标注的资源一致性满足实际需要。

关键词：篇章衔接；对齐语料标注；指代；省略；连接

Research on the Construction of Chinese-English Discourse Cohesion Alignment Corpus

Yancui Li　Jike Feng　Chunxiao Lai　Hongyu Feng
College of Information Engineering, Henan Institute of Science and Technology, Xinxiang 453003
liyancui@hist.edu.cn

Abstract

Discourse cohesion analysis plays a critical role in discourse understanding, in which there exist differences in cohesion between English and Chinese, including anaphor, ellipsis and connective. This paper aims to create a corpus containing corresponding cohesion alignment information. First, we explore proper strategies in annotating discourse cohesion, including clause, conjunction, reference and ellipsis. Then, we create resources which contains the information of alignment. Finally, this paper evaluates the corpus, discusses the problems and solutions in the annotation. The evaluation of corpus labeling and simple experimental results show that the method of corpus labeling strategy in this paper is feasible and the consistency of labeled resources meets the actual needs.

Keywords: Discourse Cohesion; Alignment Corpus Annotation; Anaphor; Ellipsis; Conjunction

1 引言

自然语言的单位可以从小到大分为词、短语和句子，最后形成一个篇章。在实际应用中，自然语言处理大都要在篇章上进行，不可断章取义，要正确理解篇章，就需要了解篇章中的衔接。衔接是一个语义概念，当篇章中的某个成分的含义需要依赖于另一个成分解释时，就会出

现衔接，汉语和英语中都有多种衔接手段。衔接主要有指代、省略和连接：指代是指用代词、冠词等表示特定的事物或已被提及过的事件；省略是指在事理逻辑上应有在字面上却没有的成份；连接主要指连接不同篇章并表达语义关系（例如因果、并列、转折等）的词语。汉英篇章衔接手段有差异，如例 1 和例 2。

例 1a: (他)^{r1} 脱下衣服的时候 c1，他 ^{a1} 听得外面很热闹，阿 Q^{a2} 生平本来最爱看热闹，(他)^{r2} 便 ^{c2} 即寻声走出去了。(他)^{r3} 寻声渐渐的寻到赵太爷的内院里，虽然 ^{c3} 在昏黄中，(他)^{r4} 却 ^{c4} 辨得出许多人，赵府一家 ^{a3} 连两日不吃饭的太太也在内，还有 ^{c5}(他们)^{r5} 间壁的邹七嫂，(也有)^{c6}(他们)^{r6} 真正本家的赵白眼，赵司晨。 (鲁迅：阿 Q 正传)

例 1b: **While**$^{c1'}$ **he**$^{r1'}$ was taking off his shirt **he**$^{a1'}$ heard uproar outside, **and since**$^{c2'}$ **Ah Q**$^{a2'}$ always liked to join in any excitement that was going, **he**$^{r3'}$ went out in search of the sound, **he**$^{r4'}$ traced it gradually right into Mr. Chao's inner courtyard. **Although** $^{c3'}$ it was dusk **he**$^{r4'}$ could see many people there: **all the Chao family** $^{a3'}$ including the mistress who had not eaten for two days. **In addition** $^{c5'}$, **their**$^{r5'}$ neighbor Mrs. Tsou was there, **as well as** $^{c6'}$ **their** $^{r6'}$ relatives Chao Pai-yen and Chao Szu-chen. (杨宪益、戴乃迭译：The True Story of Ah Q)

例 2a：尽管 ^{c1} 减轻污染 ^{a1} 的呼声不断，(并且)^{c2} 公众日渐愤怒，污染 ^{a2} 还是变得更糟糕了，(这)^{r1} 越发显出环保的紧迫性。

例 2b:**Despite**$^{c1'}$frequent calls for cutting **pollution**$^{a1'}$, **and**$^{c2'}$ growing public anger, **the problem**$^{a2'}$ has only got worse, **which**$^{r1'}$ increasingly shows the urgency of environmental protection.

例 1 中的篇章衔接方式主要有指代、省略和连接。例 1a 省略了四个主语"他"(r1-r4)，由于省略的主语在上下文中是暗含的，因此并未给读者在阅读上造成困难，省略的"他"和阿 Q 形成省略衔接；但在同样的情况下，如例 1a 的对照翻译例 1b，在英语中，主语是不能省略的，否则句子的结构上将不完整，翻译时被省略的主语 he(r1'-r4')都补充上。例 1a 中的他和阿 Q 和例 1b 中的 He 和 Ah Q 形成指代衔接。例 1a 中的连接成分"虽然"(c3)、"还有"(c5)、"也有"(c6)分别和例 1b 中的"Although"(c3')、"In addition"(c5')、"as well as"(c6')相对应，它们的功能相同，其中，连接词"也有"(c6)在汉语中是省略的，而相应的翻译中却根据意义补充了"as well as"(c6')。例 1 给出的例子反映了汉英衔接的实际情况，例 2 是(Tu Mei Tu Mei et al. ,2014)文中的实例，在翻译时，连接词"尽管"(c1)和"Despite"(c1)相应，"污染"(a2)在翻译时变成了"the problem"。综合分析例 1 和例 2 可知，汉英篇章中都存在各种衔接，衔接手段略有差异。

本文开展的汉英篇章衔接研究具有非常重要的理论意义和应用价值，形成的汉英篇章衔接对齐标注策略可用于构建语料库，所构建的语料库既可用于汉英篇章衔接的对比、翻译、教学等研究，又有助于推动汉英篇章衔接对齐分析研究及平台建设。

2 相关工作

2.1 汉英篇章衔接理论研究

Halliday and Hasan(1976)、Werth(1984)和 Cook(1989)分别将衔接进行了分类，文章中均指出主要的衔接手段包括连接、省略和指代。胡壮麟(1994)在《语篇的衔接与连贯》第一次系统地介绍了汉语篇章衔接与连贯，这本书是胡壮麟先生对 Halliday and Hasan(1976)衔接理论的继承和发展，除了保留 Halliday and Hasan(1976) 以语法和词汇为重点的衔接模式外，该书还包含了英语和汉语实例，这对汉英篇章衔接的研究具有很大的启示。周利芳(2018）和曹继阳(2019）分别对汉语篇章衔接的成分和手段进行了研究和分析。理论研究方面，汉英语篇的衔接基本都包括指代、省略、连接等，汉英语篇的衔接对比也多从这几个方面展开。奚雪峰等(2019）从篇章意图性角度探讨了篇章话题结构，并在此基础上分析了篇章的连贯性和衔接性。朱永生等(2001)的《英汉语篇衔接手段对比研究》将衔接理论用于汉英篇章对比，该书基于 Halliday and Hasan(1976)的衔接理论，运用大量的语料分析了英汉衔接手段的异同。由于汉语是一种意合型语言，人们在选择词语和句子方面通常能省则省，英语中大多数的省略都带有形式上的标记，

而汉语的省略是在不用考虑语法，甚至不用考虑逻辑的情况下表达其含义。此后许多研究者将衔接理论用于汉英语篇对比研究（常阳,2015; 钟书能,2016; 张献丽,2017;王菲,2018; 张易男和李燕鸿,2019），这些论文大多数采用 Halliday and Hasan(1976)对衔接手段的分类结合汉英语料分析汉英篇章衔接方式的异同。以上汉英对比研究取得了一定的效果，但选择的样本均较少，往往难以排除随机性对结果的影响。英汉对比研究应着眼于两种语言的特色，各自不同的趋势，需要选择有代表性且较多的样本。

2.2 汉英衔接语料库

语料库在自然语言处理技术的发展过程中起到了非常重要的作用。下面介绍包含指代、连接信息标注的语料库，以及汉英平行语料库。

包含指代信息的语料库。目前较知名的标注了指代信息的语料库主要有 MUC(Message Understanding Conference) 、ACE(Automatic Content Extraction) 、OntoNotes 语料库。MUC 语料通过指向形成指代链。ACE 中具有相同指代关系的实体位于同一指代链，且该指代链拥有唯一的编号。但 MUC 和 ACE 只标注了实体指代，并且没有考虑省略的指代标注。OntoNotes 包括词汇层面，句子层面和篇章层面多层次的标注，在篇章层面主要包含空语类信息、实体间以及事件的共指关系(Pradhan et al.，2007)。OntoNotes 中包含汉语和英语，汉语部分还标注了部分零指代信息，但零指代仅标注了主语位置，而汉语的零指代种类很多，且每一类别都有其自身的特点，这就制约了汉语零指代消解的研究。Kong and Zhou(2010)在 CTB6.0 语料标注的空语类(Empty Category)基础上进行了汉语零指代信息的标注，该语料有 150 篇文本。

包含连接信息的语料库。包含连接信息的语料库主要有宾州篇章树库(Penn Discourse Tree Bank)、汉语复句语料库、清华汉语树库、哈工大中文篇章关系语料以及苏州大学和河南科技学院合作完成的汉语篇章结构语料库（CDTB）。以上对于篇章的标注多采用英语篇章体系，李艳翠等(2015)提出一种基于连接依存树的汉语篇章结构表示方法，连接依存树的主要特征是叶子节点为子句，内部节点为连接词，连接词通过其层级地位(管辖范围)表示篇章结构的层次，通过其语义(具体与抽象)表示篇章关系。在此基础上，作者标注了 500 个文档的汉语篇章语料，其中有 24.8%的篇章关系有显式连接词。以上语料中虽然都涉及了连接词的相关标注，但均针对单语，篇章关系中汉语仅 25%左右有连接词，英语有 45.5%，可见英语连接词使用频率大于汉语。

平行语料库主要是指语料中的两种语言文本构成互译关系，目前的汉英衔接语料库主要针对单语，现有的汉英平行语料库除了做了一般性段落、句子等对齐工作外，很少进行语义等深度加工，特别是篇章层面的标注加工。因此，很难利用现有平行语料库进行基于篇章衔接的自动分析和应用研究。

综上，由于汉英衔接理论不同，衔接方式也有差别，汉英衔接对比多从指代、省略和连接方面进行，但目前的对比选择的样本均较少，不具有统计学意义。目前的汉英衔接语料库主要针对单语，现有的平行语料库只做了段落、句子等对齐工作，很少进行篇章衔接等深度加工，特别是衔接信息的对齐。这严重制约了基于篇章衔接对齐语料的语言对比及自动对齐分析工作。

3 汉英篇章衔接对齐标注策略

在充分分析现有汉英衔接理论、衔接对比分析理论和汉英衔接自动分析研究内容的基础上，本文制定了标注策略。词汇衔接由于有明显的词语指示，不是汉英衔接研究的难点，所以本文重点标注语法衔接，包括指代（本文将衔接理论中的指称和替代合并为指代）、连接和省略信息。杨传鸣（2008）对红楼梦及其英译本的衔接进行定量统计，发现在所有衔接手段中(包括词汇衔接和语法衔接)，汉语中指代、省略和连接手段占 59.6%，英语占 77.0%。本文的标注内容包括全部语法衔接，且包含大部分衔接手段，具有一定的代表性。

现有的对齐语料库中，仅仅有句子等单位对齐，而没有衔接的对齐，这直接影响汉英衔接对齐知识的获取。本文标注了子句、指代、省略和连接及其对齐信息。如例 2 的标注内容见图

1，图 1 中用相同颜色表示对齐的子句，用连线表示衔接对齐的信息，如连接词“尽管”和“Despite”对齐；用括号表示省略的信息，省略的内容可以是连接词，也可以是指代词，如：省略的内容“并且”和“and”对齐；同一语言中的指代链，用虚线表示，如“污染”和“污染”，“pollution”和“the problem”在分别在同一指代链上。实际标注中，指代、省略和恢复是相互指导，交叉进行的。

图 1. 例 2 的标注信息

汉英篇章衔接对齐语料库的对齐标注总原则是“单位对齐，词对齐”。标注语料的整体策略是源语为主，目标语为辅，即以汉语为主，英语为辅。标注目标是实现双语衔接中的子句、指代、连接的对齐标注。所以它实质上是一个“标注中有对齐，对齐中有标注”的对齐与标注合二为一的过程。

汉英篇章衔接的对齐标注，包括切分（子句）对齐、连接词对齐、指代对齐这几个关键对齐标注任务，由于本文考查的省略主要是连接词省略和指代省略，因此将其标注合并到相应的项目中，在标注时体现省略信息。下面详述其标注策略。

3.1 子句对齐标注

基本假设是具有对译关系的篇章，其内部的子句是一一对应的，参考李艳翠等（2013）的子句定义。英汉双语篇章子句的对齐，为了保持结构的一致性，一般采用“源语优先”即汉语优先的划分子句的方法，首先按既定的汉语基本篇章单位进行切分，然后以英语对齐（最终可根据结果归纳英语基本篇章单位）来保证汉英篇章的对应关系。根据子句定义，英语的从句或句子和子句对应，子句对齐后便于衔接信息的对齐标注。本文子句以汉语为主，将英语相应的组块（英语从句或短语）和汉语子句对应，事实上，这种分析对于汉语是子句分析，对于英语则主要是子句对齐。这种分析机制，可以保证所研究的问题是篇章层面的问题。

在实际操作中，主要依据三点：第一主要看英汉的句意。对于一个优质的翻译文本，源语中的因果、转折、并列等逻辑语义关系必然在目的语中得到反映，根据逻辑语义关系，可以分别从英汉平行语料库中相邻的子句中找出其对应关系，从而进行英汉的对齐划分；第二看结构，结合源语与目的语的结构，英汉中主谓宾的顺序是一致的，一些名词性从句、状语从句的对译也较为一致，找出英汉中相应的词汇从而找出英汉相对应的句子成分进行划分。比如看源语中结尾的动词、非谓语动词、宾语、各种从句或是其他成分在汉语中是否得到了体现；第三是看标点，在对译的中文语料库中，英文的标点大部分会和汉语一致，根据标点情况，可以更清楚地推测文意及翻译的中文文本。

如例 3，汉语子句“比开放前的一九九一年增长九成多。”和英语子句“growing more than 90 % compared to 1991 , before they had opened .”对应。

例 3a：中国十四个边境对外开放城市一九九五年经济建设取得可喜成果。|据统计，这些城市去年完成国内生产总值一百九十多亿元，|比开放前的一九九一年增长九成多。

例 3b: In 1995 , the economic construction of China 's fourteen border municipalities that are open to the outside attained gratifying results .| According to statistics ,these municipalities last year fulfilled more than 19 billion yuan of the gross domestic product ,| growing more than 90 % compared to 1991 , before they had opened .

3.2 连接词对齐标注

句子之间或子句之间存在如条件、转折、因果等语义连接关系，连接词指具有子句及其以

上语法单位连接和关系提示作用的语言单位，可以根据连接词的管辖范围(连接的子句)和篇章关系两方面确定连接词。李艳翠等（2015）将连接词作为篇章关系的关键因素在汉语中已进行了标注，参考汉语篇章结构中的做法，在汉英连接词对齐标注时，对连接词是否可添加和可删除进行标记，为便于操作，本文仅对在汉语、英语或汉英中都出现的连接词进行标注，对双语均省略的连接词，由于添加时所选择的词范围较大，容易导致对齐标注不一致，且在实际应用中意义不大。汉英对译篇章由于意义相同，所以对于连接词的汉英对齐标注主要为管辖范围和逻辑功能的对齐，标注时如连接词缺省则根据意义对连接词进行添加，对汉英都无法添加连接词的情况不进行标注。

李艳翠等（2015）在汉语连接词分类中认为，连接词可分为四大类：并列类、转折类、解说类和因果类，在此基础上又可分为 17 种不同的关系类型。例如，并列类可分为并列关系、顺承关系、递进关系、选择关系和对比关系五种关系类型。每种关系类型又包含多个连接词，而某些连接词可属于不同的关系类型。标注时主要考虑三种连接词对齐关系，如例 4 汉语没有连接词而英语有连接词，如例 5 汉英均有连接词，如例 6 汉语有而英语没有连接词。

例 4a:其中，台湾对祖国大陆输出值为一百七十八亿美元，比上一年增长百分之二十；|输入值为三十一亿美元，比上年增长百分之七十四。

例 4b:The number of investment projects dropped by 444 as compared with last year ,| **but** the value of investments rose by more than 130 million US dollars as compared with last year .

例 5a：并投资一千三百多个亿，加强基础设施和基础产业建设，|**为**扩大对外开放创造良好环境。

例 5b:It has invested more than 130 billion yuan to strengthen the construction of infrastructures and basic industries| **so as to** create a sound environment for expanding the opening up to the outside world.

例 6a:在投资项目上比上年减少四百四十四件，|**但**投资金额却比上年增加一点三亿多美元。

例 6b:Last year, the number of investment proposals presented by Taiwanese businesses and approved by Taiwan authorities totaled 490 ,| with a value of 1.092 billion US dollars.

在翻译时，允许出现不是一对一的情况，如例 7：

例 7a:在社会主义市场经济体制建设不断推进，对外开放进一步扩大的新形势下，海关的职能**不能**削弱，|**只能**加强。

例 7b:Under the new circumstances in which the construction of a socialist market economy mechanism is continually being promoted and the opening up to the outside world is further expanding, the functions of Customs **should not be** weakened, |**and should only be** strengthened .

3.3 指代对齐标注

经过反复的研究和实践，最终确定汉英篇章衔接对齐标注总原则，以篇章为单位将 ACE 实体类型为人名、地名、机构名、时间等具有代表性的且在文章中出现频率较高的指代实体词进行汉英对齐标注。标注原则是单语中的指代信息构成指代链，汉英指代链中的项目两两相互对应。标注过程中采用边标注指代链边进行双语对齐，标注和对齐同时进行，这样可以全面考察双语的各种信息。

本文标注实体指代和事件指代信息，如例 8 的“金川公司”是实体代词，“这里”“这家企业”是事件指代。例 8a 中的“金川公司”“这里”“金川公司”和“这家企业”分别对应例 8b 的“Jinchuan Company”、“this place”、“the Jinchuan Company”和“this enterprise”，同时形成指代关系，在本篇章中都指的是“金川公司”，因此将有指代信息汉英指代词标注在同一指代链。

例 8a: 一九六四年，**金川公司**产出第一批电解镍。从此以后，逐步改变了中国镍、钴及铂族金属长期依赖进口的局面。如今，**这里**已成为中国最大的镍钴生产基地和铂族金属提炼中心，镍和铂族金属产量分别占全国的百分之八十八和百分之九十以上，被誉为中国的“镍都”。一九

七八年，**金川公司**被中国政府列为全国矿产资源综合利用三大基地之一，作为中国镍工业代表的**这家企业**由此踏上依靠科技进步求振兴的发展之路。

例 8b: In 1964 , **Jinchuan Company** produced the first batch of electrolytic nickel .From then on , the situation of China 's long time dependence on import for nickel , cobalt and platinum family metals has been changed gradually .Up to now , **this place** has become China 's largest nickel and cobalt production base and platinum family metals refining center , with an output of nickel and platinum family metals that respectively account for more than 88 % and 90 % of the whole country respectively , being praised as China 's " Nickel Capitol " .In 1978 , **the Jinchuan Company** was listed by the Chinese government as one of the top three bases of integrated utilization of national mineral resources .Since then , **this enterprise** , as a representative of China 's nickel industry , began to step onto its vigorous development road by relying on advances in science and technology .

3.4 省略对齐标注

省略可以包含代词的省略、名词的省略以及连接词的省略等，本文认为指代和连接都可以省略。由于对篇章的理解是主观的，因此特别是翻译者的主观理解将会添加到翻译后的文本中，以更好的反映原文，因此省略处理的原则是汉英都省略的不做处理，主要处理汉语或者英语省略。由于汉语省略较多，标注时以英语为主，在汉语中寻找对应，如不存在则补充，存在则对齐，如不能补齐，则对空。如图 2 例 3a 中，根据英语对照补充两个省略的代词“他”(例 3a 中用“()”标示)，“(他)-he”、“他-he”、“阿 Q-Ah Q”以及“(他)-he”依次对齐。如图 1 中的例子“and”在是翻译时补充的内容，可以分析得出汉语中省略了对应的词“并且”。当然，也有一些词是汉语中有，而英语在不影响理解的情况下省略，此时英语中也补充并对齐。

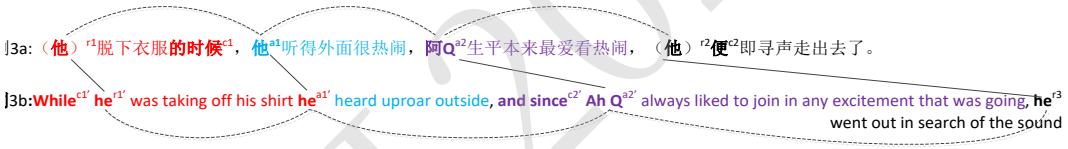


图 2. 例 1 省略和指代的对齐标注

4 汉英衔接对齐资源创建

本文将充分利用已有的汉语篇章级资源，在 OntoNotes 的汉英平行文本上追加与篇章衔接性相关的指代、省略和连接标注信息并进行汉英标注内容的对齐。为了便于标注，基于标注策略，制定了标注规范，开发了辅助标注平台，并以人工和计算机结合的方式进行语料标注。

4.1 语料选择

在 OntoNotes 中已经包含实体、部分省略信息。但这些信息是单语标注，没有体现双语对齐关系。本文在此基础上添加其他衔接信息，同时考虑双语，标注的同时完成对齐，具体包括：1）将汉英篇章中的子句标注扩展到双语；2)连接词及其对齐标注，以前期研究为指导，标注连接词属性和对齐信息，包含添加的连接词和连接词是否可删信息，连接词的管辖范围，连接词所连接的篇章单位是否调序等；2)种类齐全的汉英省略信息：OntoNotes 语料中仅包含了主语位置的零指代关系，而汉语省略涉及多个种类，这里主要标注指代和连接两种省略信息。

4.2 标注规范

根据篇章衔接分析机制和对齐策略，针对子句、连接词、指代、省略的标注及对齐分别提出具体的标注规范。标注规范注重可操作性，分别从判定原则、对齐方法等方面入手制定，并制定了标注规范。

4.3 标注方法

在标注规范的指导下进行标注，标注工作参考了之前汉语篇章结构语料资源构建积累的方

法和经验，分阶段进行：在第一阶段，由于语料库处理工作量大，为了确保质量和通用性，制定了初步的标注规范，同时开发了标注工具，并对参与标注的人员进行了培训；第二阶段，为保证标注的一致性，将标注者分为三组，分别标注了若干篇相同的文档，然后在一起讨论所有标注内容，包括指代、省略和连接的属性和对齐方式等，形成统一的标注思想，而后得到修订后的标注规范；第三阶段，标注者分组完成 60 篇相同文档的标注，用标注完的文档两两计算标注的一致性。选取一致率高的两组语料，由标注成员共同参与讨论，经过多次研究形成最终的标注规范；第四阶段，根据最终的标注规范，由标注一致率高的两组成员继续完成剩下语料的标注，另一组成员负责完成语料校对和一致性的计算，形成最终的汉英篇章衔接对齐语料库。

对于子句、指代、省略和连接及其对齐信息的标注，本文开发了辅助标注平台，根据用户选择记录需要添加的词、标注信息的类型，对齐的位置等信息，使用人机结合的标注策略，提高标注质量和效率。

4.4 标注结果

完成了 200 个平行文档的汉英篇章衔接对齐语料标注。标注了 200 个平行文档的子句切分、连接词对齐和指代词对齐语料。根据制定的汉英子句对齐切分标准，通过汉英子句对齐的标注规范，即对平行语料库进行汉英子句对齐语料标注。目前平行语料中共有效标注汉语句子 899 句，英语句子 1281 句，汉英 2153 个子句对，汉语子句平均长度是 11 个词语，英语子句平均长度是 20 个单词。汉语子句对应的英语子句主要句法结构有 S、VP、NP、PP 等。连接词对齐标注中，共标注了 817 对连接词，如“但”和“nevertheless”对应，共标注显式连接词 462 次，出现次数较多的连接词（并 and）占 50.9%，汉语中隐性连接词达到 60%。指代对齐标注中，目前共标注文档有效文档 193 篇，标注了 1613 个指代链，平均每篇文档有 8.4 个。共标注了 3657 个指代词，平均每个指代链上有 2.3 个指代词。省略情况主要是连接词省略和指代省略，在连接词省略中，中文省略 122 个词，英文省略 3 次，中文省略现象明显多于英文。指代省略 114 次，其中中文省略 92 次，英文 22 次。

5 实验结果与分析

5.1 标注质量评估

一致性评估主要考察标注者标注的一致内容与所有标注内容之比，本文从汉语一致性、英语一致性和汉英对齐一致性三方面进行考察。其中，汉英对齐一致性指的是标注者对相同语料的汉语标注一致并且汉语相对应的英语对齐标注也一致的情况。标注工作有 6 名同学参与，前期将 6 名同学两两分为 A、B 和 C 三组进行标注，对其标注的 60 篇文档进行逐一探讨并两两计算一致性，得出 A-C 小组在在汉语一致性、英语一致性和汉英对齐一致率等方面明显高于其他两个小组，因此由 A-C 小组继续完成剩下文档的标注工作，B 小组成员负责校验。由于标注内容不同，针对子句、连接词和指代词分别采用了不同的计算方法。目前共完成 200 篇文档的标注工作，其子句对齐、连接词对齐和指代对齐语料评估结果如表 1 所示。

		汉语一致性	英语一致性	汉英对齐一致性
子句对齐	切分对齐 I	0.972	0.992	
	切分对齐 II	0.968	0.930	0.909
连接词对齐	显隐对齐	0.962	0.987	0.974
	显式连接词对齐	0.800	0.950	0.876
	全部连接词对齐	0.678	0.690	0.684
指代词对齐		0.933	0.932	0.920

表 1.标注一致性计算结果

子句对齐亦可称作切分对齐，切分对齐的方法有两种：切分对齐方式 I：汉语子句的切分位置均标有标点符号，并计算了用作切分标记的标点符号（，；：。）一致性。英语子句切分不一定使用标点符号作为切分标记，可以使用空格（基本上是任意单词或标点符号）的形式作为切分标记，以及是否可以使用任何空格作为一致性计算的切分标记；切分对齐方式 II：计算不同标注者的所有切分（AUB）之间的共同切分（A∩B）的一致性。对于句子位置 SentencePosition = “ X1 ... X2 | Y1 ... Y2”，计算 A 和 B 的切分位置相同的情况。与切分对齐方式 I 相比，该方法的评估更准确，可以统一中英文切分评估标准。

从表 1 可以看出，子句切分对齐 I 在汉语和英语一致性上较高，主要是每个切分位置都进行计算，计算的无歧义切分位置较多。采用子句切分 II 计算出汉英对齐一致性为 90.9%，说明子句完全对齐还有待提高，可以从提高英语切分对齐标注的位置精准性和在汉语指导下进一步实现英语切分对齐这两方面的改进可以有效改善切分对齐标注准确率。

由于连接词总是有一定的管辖范围，且连接词有显隐之分。连接词对齐标注评估，从显隐对齐、显式连接词和对齐全部连接词对齐三个方面进行评估。由表 1 一致性结果可知，显隐对齐一致率较高，其中英语一致率达 0.987，同时英语普遍高于汉语的一致率。这是因为英语显式连接词明显较汉语的多，相比汉语，英语对于连接词有比较共性的认识，汉语的认识却有较大分歧。这也证明英语在关系对齐标注时作为指导性标准的可靠性。显式连接词对齐的一致性高于全部连接词，主要是表示同中连接关系所添加的隐式连接词不固定，如表因果可以是“因为”、“因”等词。为提高连接词对齐标注的准确率，可从两方面入手：第一，进一步明确汉语连接词的定义，从而增强汉语显式连接词的对齐标注效果。第二，规范隐式连接词的添加，指定添加连接词的范围，减少隐式连接词添加的分歧。

指代词对齐主要计算标注者选择指代词的一致性，由于指代词通常比较明显，添加的指代词多为名词且固定，所以一致性高于连接词对齐。汉英指代词对齐标注的一致性达 0.920 在指代对齐标注一致性计算中除对汉语一致、英语一致、汉英对齐一致率进行计算，还加入了汉语位置一致、英语位置一致、属性一致、指代词个数一致和指代链个数一致率的计算，其对应的一致率分别为：0.926 、0.925 、0.931 、0.932 、0.872，其一致率的计算对汉英篇章衔接对齐语料库的构建具有重要的参考意义。由于两小组同学进行双盲标注，标注结果存在一定差异。讨论过后，将进一步规范标注策略，对一些文档标注完善，个别误差大的文档进行重新标注。在对结果进行一致性评估时，将考虑去除一些无效指代链，将进一步提高一致率的精确性。

5.2 简单实验结果

李艳翠等（2013）在基于逗号的汉语子句识别研究中，手工标注了 100 篇文档。实验结果表明，具有最佳识别效果的最大熵分类器模型使用 CTB6.0 提供的标准语法树，最高准确率为 92.8%，使用 Berkeley 自动语法分析树，最高准确率是 89.9%。本文开发了汉英子句切分平台和英语子句切分平台，利用最大熵、决策树、贝叶斯等模型进行训练，然后分别进行汉语、英语子句的自动切分。得到中文自动切分准确率 90%，英文 93%。在此基础上，进行基于 BiLSTM-CRF 模型进行分析，汉英子句切分 P、R、F 分别为 92.3%、94.4%、93.4%和 95.5%、93.4%、94.4%。中文连接词自动识别准确率为 92.5%，中文 95.7%。

汉英连接词的自动识别实验，中文连接词自动识别准确率为 92.5%，中文 95.7%。李艳翠等（2015）在标注了 CTB6.0 中 500 个文档的实验结果表明，具有最佳识别效果的解说类的准确率为 82.5%，连接词自动识别并分类的总正确率为 89.1%。本文的关系识别采用英文连接词本身和对应的中文连接词作为特征，通过实验统计，给定连接词中，并列类有 332 个，占 71.86%；解说类 43 个，占 9.31%；转折类 23 个，占 4.98%；因果类 64 个，占 13.85%。通过表 2 中结果看到，并列类，因果类和解说类分类结果较好，转折类识别效果较差。

关系类别	正确率	召回率	F1 值
并列类	95.45	90.00	92.59
因果类	94.43	77.38	83.52
转折类	30.00	88.14	44.28
解说类	95.80	67.03	72.79

表 2.连接词识别结果

实验发现，由于在关系类别分布中并列类所占比例最高，训练实例最多，并且连接词的集中度较高，因此识别率相对较高。但是存在连接词一对多的现象，如“and”在并列类中出现 217 次，在解说类和因果类中各出现 1 次，所以会导致一些识别错误。转折类识别效果最差，一是因为关系类别分布中转折类出现次数最少，只有 23 次，二是因为有的转折类连接词同时对应了其它的关系类别，如“but” 在转折类中出现 15 次，在并列类中出现 1 次。而且观察测试结果发现也将该“but”归为了转折类。解说类虽然比例较低，在训练集中共出现 43 次，但是解说类连接词比较明显，如出现时多集中为“among”，“among which”，“of which”等词语，所以解说类识别准确率较高。根据实验分析，与同一连接词对应的关系类别越少，该词的歧义性越小，每个类别的连接词越集中，出现频率越高，连接词类别识别准确率就越好。以后通过增加训练集规模，训练结果会得到大幅的提升。

6 标注中的难点问题及解决

6.1 标注对象问题

在最初的标注过程中，发现标注结果中真正形成指代链的实体词较少，并且存在较多指代词单独成链的现象，最终造成不同标注者的标注结果存在较大差异。经过反复的实践和讨论，最终统一标注规范，将有较多指代词的 ACE Type 为 GPE、ORG、LOC、PERSON 和 DATE 的实体词标注处理，存在较少实体词甚至往往仅有单独一个实体词的 ACE Type 为 MONEY、PERCENT、EVENT、QUANTITY 和 CARDINAT 等实体词不再单独标注成链。

例 9a: （中国）h1 羽绒及其制品行业是（八十年代中期）d1 开始快速发展的，全行业利用（中国）h2 资源、人力优势，加上注重引进国外先进技术与设备，产品产量和质量得以大幅度提高。据不完全统计，目前（中国）h3 已有羽绒及制品加工企业（三千余家）c1，其中上规模的达（六百多家）c2，从业人员约（三十万）c3，形成年产羽绒制品（五千多万件）c4 生产能力，年工业总产值达（八十亿元）c5。通过（十余年）d2 市场开拓，（中国）h4 现已成为世界主要羽绒生产国和羽绒制品出口国，年出口羽绒近（三万吨）c6、羽绒制品（二千多万件）c7，创汇达（八点二亿美元）c8，其中羽绒服装出口额占行业出口总额（百分之五十）c9 以上。

例 9b:**China 's**[h1] down and down products industry started its rapid development **in the mid '80s**[d1] .The entire industry makes use of **China 's**[h2] resources and manpower advantage , and additionally stresses introducing advanced foreign technology and equipment , thus increasing production volume and quality by a large margin .According to incomplete statistics , **China**[h3] currently has over **3,000**[c1] down and down product enterprises , among which , those above scale have reached more than **600**[c2] , with employed staff of about **300,000**[c3] .It has an annual production capacity of **50 million**[c4] down products with a total annual industrial output value reaching **8 billion yuan**[c5] .Through more than **ten years '**[d2] market development , **China**[h4] has now become the world 's main down manufacturing country and down products export country , annually exporting nearly **30,000 tons**[c6] of down and over **20 million**[c7] down products , with earned foreign exchange reaching

820 million US dollars[c8] , including down clothing export values accounting for more than **50 %**[c9] of total industry export values .

例 9 中 ACE Type 为 GPE 的实体词有（h1~h4）,依据对齐标注原则，该实体词可标注成指代链。其中 ACE Type 为 DATE 的实体词有 d1 和 d2,因其仅有一个实体词，不单独标注成链。ACE Type 为 CARDINAT 的实体词(c1~c4 和 c7) 、ACE Type 为 MONEY 的实体词 c5 和 c8、ACE Type 为 QUANTITY 的实体词 c6 以及 ACE Type 为 PERCENT 的实体词 c9 不在要求标注的实体词范围内，同样不单独标注成链。

6.2 特殊语境指代词标注难点

例 10a: 近年来，(中)e1(韩)q1 两国之间的经贸往来发展迅速。截止去年九月，(韩国)q2 在(华)e2 投资企业总数为五千八百八十三家，(中国)e3 已成为(韩国)q3 最大的投资对象国。据(中国)e4 海关统计，一九九五年两国贸易额已达一百六十九点八亿美元，比前年增长百分之四十四点八。经济专家预计，今年(中)e5(韩)q4 两国贸易额将增至二百五十亿美元。

例 10b:In recent years , the economy and trade contacts between the countries of **China**[e1] and **South Korea**[q1] have been developing rapidly .By September of last year , the total number of **Korean**[q2] enterprises investing in **China**[e2] totaled 5,883 .**China**[e3] has become **Korea 's**[q3] largest target country for investment .According to **Chinese**[e4] Customs statistics , in 1995 , trade between the two countries reached 16.98 billion US dollars , increasing 44.8 % compared with that of the previous year .Economic experts estimate that this year trade between the two countries of **China**[e5] and **South Korea**[q4] would increase to 25 billion US dollars .

例 10 中的 “中”（e1）、“华”（e2）和“中”（e5）若单独出现时，并不能准确判断其具体含义。在本篇文章中，根据其在文章的语境，以及上下文信息,很容易判断其与“中国”（e3 和 e4）形成指代衔接，将其（e1~e5）标注在同一指代链，对应的英文中正确翻译出“china”。同样“韩”（q1）和“韩”（q4）与“韩国”（q2 和 q3）形成指代衔接，应将其（q1~q4）标注在同一指代链，对应英文翻译“South Korea”。

7 结语

本文进行了汉英篇章衔接语料库的标注工作，主要实现了子句、连接词、指代和省略的对齐标注。汉英篇章衔接对齐语料库的对齐标注总原则是“单位对齐，词对齐”，标注语料的整体策略是以汉语为主，英语为辅，省略添加的原则是汉语或英语有对应显式词出现。子句以汉语为主，将英语相应的组块（英语从句或短语）和汉语子句对应。连接词对齐标注连接词及其语义关系，根据其体现为管辖范围和逻辑功能的对齐。单语中的指代信息构成指代链，汉英指代链中的项目两两相互对应，汉英都省略的不做处理，主要处理汉语或者英语省略。在本文汉英衔接对齐标注策略基础上，选择汉英平行文本进行了汉英篇章衔接资源的构建，目前完成了 200 篇平行文档的标注工作。标注中采用辅助平台，对子句、连接词、指代的标注质量分别进行评估，评估结果说明本文方法切实可行，简单实验结果表明本语料子句切分、连接词识别具有较强的可计算性。下一步工作将不断完善本标注策略，扩大标注语料，进行指代和省略的计算分析工作。

致谢

本文得到国家自然科学基金项目（61502149）、河南科技学院高层次人才科研项目（2017039）、华东师范大学统计与数据科学前沿理论及应用教育部重点实验室开放课题资助。实验室小组吕天赐、李书磊、李强、胡大帅、侯昆昊和王基翱等同学认真负责的参与了标注工作。感谢公开学习资源的专家学者，感谢提出建设性指导意见的 CCL 2020 的专家学者。

参考文献

Cook G. 1989. *Discourse*. Oxford Univ Pr (Sd).

Halliday，M．A．K．＆R．Hasan．1976. *Cohesion in English*．Edward Arnold , London.

Halliday，M．A．K．1994. *An Introduction to Functional Grammar*．Edward Arnold , London.

Huang H H, Chen H H．2011.*Chinese discourse relation recognition Proceedings of the 5th International Joint Conference on Natural Language Processing*． Asian Federation of Natural Language Processing． Chiang Mai. 1442-1446.

Kong F. and Zhou G. D. 2010.*A Tree Kernel-based Unified Framework for Chinese Zero Anaphora Resolution.* In Proceedings of EMNLP 2010.

Li Y C，Feng W H，Kong F，et a1．2014.*Building Chinese discourse corpus with connective-driven dependency tree structure.* In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing(EMNLP)．Doha．2105-2114.

Li Y，Feng W，Sun J，et a1．2014.*Building Chinese dis— course corpus with connective-driven dependency tree structure.* In Proceedings of EMNLP 2014. 2105-2114.

Louis A，Joshi A，Nenkova A．2010.*Discourse indicators for content selection in summarization Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*．Portland：Association for Computational Linguistics．147-156.

McCallum A K．2002.*Mallet：a machine learning for language toolkit*．(2002)[2012-02-28]．http：// mallet．CS．umass．edu.

Pradhan S., Hovy E. and Marcus M. et al. 2007. *OntoNotes: A Unified Relational Semantic Representation.* Iternational Journal of Semantic Couputing, 1(4):405-419.

Pradhan S., Ramshaw L., and Weischedel R. 2007. *Unrestricted Coreference: Identifying Entities Events in OntoNotes*. In Proceedings of ICSC'2007.

Tu M., Zhou Y. and Zong C. Q. 2014. *Enhancing Grammatical Cohesion: Generating Transitional Expressions for SMT.* In Proceedings of ACL'2014.

Werth P. 1984.*Focus, Coherence and Emphasis*. Routledge Kegan & Paul.

曹继阳.2019.*汉语口语语篇衔接手段与衔接成分——基于经典情景喜剧《我爱我家》的研究*.语言文字应用,(2):142.

常阳.2015.*英语倒装结构在汉英篇章翻译中的衔接功能及应用*.北方文学（中旬刊）,(12):70-71.

冯洪玉,李艳翠,冯文贺.2019.*基于汉英平行语料库的英文显式篇章关系识别*.河南科技学院学报（自然科学版）47(5):55-62.

冯文贺,李艳翠,任函,等.2017.*汉英篇章结构平行语料库的对齐标注评估*.中文信息学报,31(3):86-93.

胡壮麟.1994.*语篇的衔接与连贯*.上海外语教育出版社,上海.

孔芳,王红玲,周国栋.2019.*汉语篇章理解研究综述*.软件学报,30(7):2052-2072.

李艳翠. 2015.*汉语篇章结构表示体系及资源构建研究*.苏州大学,江苏.

李艳翠,冯文贺,周国栋,等.2013.*基于逗号的汉语子句识别研究*.北京大学学报（自然科学版）, 49(1):7-14.

李艳翠,孙静,冯文贺,等. 2015.*基于连接依存树的汉语篇章结构分析平台*.中国中文信息学会.中国中文信息学会 2015 学术年会(CIPS2015)暨第十四届全国计算语言学学术会议(CCL2015)、第三届基于自然标注大数据的自然语言处理国际学术研讨会(NLP-NABD2015)论文集.1-10.

李艳翠,孙静,周国栋. 2015.*汉语篇章连接词识别与分类*.*北京大学学报（自然科学版）*, 51(2):307-314.

王菲.2018.*从衔接论看汉英语篇翻译中的衔接与连贯——以《落花生》为例*.青春岁月,(11):136-137.

奚雪峰,孙庆英,周国栋.2019.*面向意图性的篇章话题结构分析研究与展望*.计算机学报, 42(12):2769-2794. DOI:10.11897/SP.J.1016.2019.02769.

徐凡,朱巧明,周国栋,等.2014.*衔接性驱动的篇章一致性建模研究*.中文信息学报,28(3):11-21,27.

杨传鸣.2008.*《红楼梦》及其英译本语篇衔接对比*.黑龙江大学,哈尔滨.

杨凤丽.2012.*论汉英否定篇章衔接功能的对比*.齐齐哈尔大学学报：哲学社会科学版,(2):155-156.

张献丽.2017.*略论汉英翻译中的衔接性*.牡丹江大学学报,26(10):146-147,150.

张易男,李燕鸿.2019.*汉英"照应"衔接对比与翻译研究——以《2018 年政府工作报告》及其英译版为例*.英语教师,19(9):134-138,141.

钟书能.2016.*话题链在汉英篇章翻译中的统摄作用*.外语教学理论与实践,(1):85-91,58.

周利芳.2018.*汉语"提及"类衔接成分的用法及其辨析*.华文教学与研究,(3):61-69.

朱永生,郑立信,苗兴伟.2001.*英汉语篇衔接手段对比研究*.上海外语教育出版社,上海.

Cross-Lingual Dependency Parsing via Self-Training

Meishan Zhang[1] and **Yue Zhang**[2*]
1. School of New Media and Communication, Tianjin University, China
2. Institute of Advanced Technology, Westlake Institute for Advanced Study
mason.zms@gmail.com,
zhangyue@westlake.edu.cn,

Abstract

Recent advances of multilingual word representations weaken the input divergences across languages, making cross-lingual transfer similar to the monolingual cross-domain and semi-supervised settings. Thus self-training, which is effective for these settings, could be possibly beneficial to cross-lingual as well. This paper presents the first comprehensive study for self-training in cross-lingual dependency parsing. Three instance selection strategies are investigated, where two of which are based on the baseline dependency parsing model, and the third one adopts an auxiliary cross-lingual POS tagging model as evidence. We conduct experiments on the universal dependencies for eleven languages. Results show that self-training can boost the dependency parsing performances on the target languages. In addition, the POS tagger assistant instance selection can achieve further improvements consistently. Detailed analysis is conducted to examine the potentiality of self-training in-depth.

1 Introduction

Cross-lingual dependency parsing has received increasing attention in recent years (Hwa et al., 2005; McDonald et al., 2011; Tiedemann et al., 2014; Guo et al., 2016a; Agić et al., 2016; Schlichtkrull and Søgaard, 2017; Rasooli and Collins, 2017; Rasooli and Collins, 2019; Zhang et al., 2019), which aims to parse target low-resource language with the supervision of resource-rich language. In this paper, we focus on the unsupervised setting (Ma and Xia, 2014; Guo et al., 2015; Rasooli and Collins, 2015; Tiedemann and Agić, 2016; Agić et al., 2016; Schlichtkrull and Søgaard, 2017; Ahmad et al., 2019), where no targeted dependency treebank is given.

Recent advances of multilingual word representations (Smith et al., 2017; Chen and Cardie, 2018; Mulcaire et al., 2019; Pires et al., 2019; Lample and Conneau, 2019; Wang et al., 2019; Wu and Dredze, 2019) has substantially promoted cross-lingual dependency parsing, especially serving as the basic input features for model transfer methods (Guo et al., 2016a; Schuster et al., 2019; Wang et al., 2019). They reduce the input divergences between languages significantly. As a result, the cross-lingual transfer learning setting can be considered highly similar to the monolingual semi-supervised and cross-domain settings. In light of this, the self-training strategy, which is widely adopted for cross-domain parsing (Reichart and Rappoport, 2007; Rush et al., 2012; Yu et al., 2015; Saito et al., 2017; More et al., 2019), can be potentially applicable for cross-lingual dependency parsing as well. However, relatively little work has demonstrated the effects of this potential method.

Instance selection for the next-round training is the key to self-training (Mihalcea, 2004; McClosky et al., 2006a; McClosky et al., 2006b; He and Zhou, 2011; Artetxe et al., 2018), which requires a certain criterion to rank the automatic outputs from the baseline model (Goldwasser et al., 2011; Yu et al., 2015; Zou et al., 2019). Such criteria are typically derived from the baseline model directly, for example, the prediction probability (Zou et al., 2018), and the delta probability between the final output and the second-best candidate output (Yu et al., 2015). Here we hypothesize that we can improve the performance of self-training by an auxiliary task which is highly corrective with the target task. A natural auxiliary task for cross-lingual dependency parsing is universal Part-of-speech (POS) tagging.

*Corresponding author.

Proceedings of the 19th China National Conference on Computational Linguistics, pages 807-819, Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

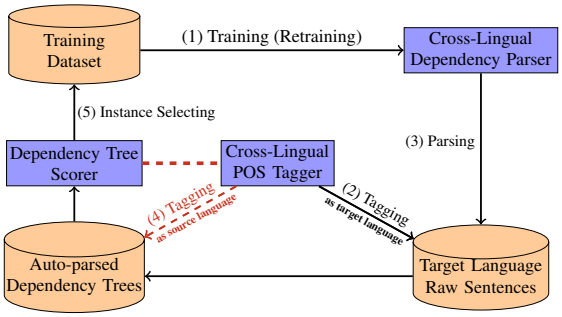


Figure 1: The overall architecture of self-train, where cross-lingual POS tagging is used to assist the instance selection in this work.

POS tags have served as one basic feature for dependency parsing (Zhang and Nivre, 2011; Kiperwasser and Goldberg, 2016; Dozat and Manning, 2016), and universal POS tags have been one important feature source for cross-lingual dependency parsing (McDonald et al., 2011; Petrov et al., 2012). The construction of a POS tagging corpus for a target language has a much lower cost than that of a dependency treebank, leading to the majority work of cross-lingual dependency parsing assuming gold-standard POS tags as inputs (Guo et al., 2016a; Rasooli and Collins, 2015; Tiedemann and Agić, 2016; Rasooli and Collins, 2017). We assume that a POS tag training corpus for the target language is available.

Based on the above settings, we investigate the capacity of self-training for cross-lingual dependency parsing empirically. Taking the BiAffine parser (Dozat and Manning, 2016) as the major architecture and enriching the model with multilingual BERT word representations (Devlin et al., 2019), we evaluate two widely-adopted instance selection strategies of self-training, and further propose a POS tagging guided criterion, which is illustrated in Figure 1. In particular, a supervised cross-lingual POS tagging model is trained to guide the instance selection in self-training, which uses a language-aware parameter generation network (PGN) (Platanios et al., 2018; Jia et al., 2019) for language switching. Our goal is to choose the target language sentences for which the POS tag outputs change relatively little when they are intentionally marked as source language sentences.

We conduct experiments on the Universal Dependencies (McDonald et al., 2013; Nivre et al., 2016) to study the effectiveness of self-training. English is selected as the source language, and eleven target languages belonging to four different families are investigated. Results show that self-training is an effective way for cross-lingual dependency parsing, boosting the dependency parsing performances of all selected target languages. In addition, POS-guided instance selection achieves further improvements. Finally, we conduct detailed analysis to understand the effectiveness of our self-training methods on four representative languages, one for each language family. All codes and datasets will be released publicly available on `https://github.com/zhangmeishan/selftraining` for research purpose under Apache License 2.0.

2 Related Work

Existing work on cross-lingual dependency parsing can be classified into two categories, namely model transferring and annotation projection, respectively. The first aims to train a dependency parsing model on the source-language treebank (McDonald et al., 2011; Guo et al., 2016a; Guo et al., 2016b), and then use it for target languages directly. Language independent features are exploited in order to minimize the gapping between the source and target languages, including multilingual word clusters (Täckström et al., 2012), word embeddings (Guo et al., 2015; Duong et al., 2015b; Duong et al., 2015a; Zhang and Barzilay, 2015; Guo et al., 2016b; Ammar et al., 2016; Wick et al., 2016; de Lhoneux et al., 2018), universal POS tags (McDonald et al., 2011; McDonald et al., 2013) and multilingual contextualized word representations (Wang et al., 2019; Wu and Dredze, 2019). In this work, we build our baselines with multilingual BERT, which has demonstrated state-of-the-art effort for cross-lingual model transferring (Wang et al., 2019).

Annotation projection aims to construct an automatic target-language dependency treebank by projecting

source language dependencies into target language sentences (Hwa et al., 2005; Ganchev et al., 2009). It relies on a parallel corpus, where source dependencies can be obtained by a source parser (Ma and Xia, 2014; Rasooli and Collins, 2015; Xiao and Guo, 2015; Agić et al., 2016; Schlichtkrull and Søgaard, 2017) or manually annotated (Tiedemann et al., 2014; Tiedemann, 2015; Tiedemann and Agić, 2016). Word-level alignment between sentence pairs has been used to project source dependencies into the target sentences. Our self-training strategy is similar in constructing automatic training datasets for target languages, while the key idea is significantly different.

Self-training has been shown effective for a number of NLP tasks (Mihalcea, 2004; McClosky et al., 2006a; Sagae, 2010; Goldwasser et al., 2011; He and Zhou, 2011; Artetxe et al., 2018). For dependency parsing, Rush et al. (2012) show that it fails to improve the performance under a supervised setting. Several studies have demonstrated its effectiveness on neural dependency parsing under the fully supervised multilingual setting (Rybak and Wróblewska, 2018). Lightly supervised learning and cross-domain adaption are more successful settings for self-training (McClosky et al., 2006b; Reichart and Rappoport, 2007; Rush et al., 2012; Yu et al., 2015; More et al., 2019). Our work applies self-training in the unsupervised cross-lingual setting. There is only one work of a similar setting. Rasooli and Collins (2017) add a number of auto-parsed outputs to enlarge the training dataset as an auxiliary technique. Their auto labeling is limited to the small-scale raw corpora with gold-standard POS tags, obtaining much smaller improvements than our work. To our knowledge, we are the first work to study self-training systematically.

3 Models

In this section, we describe the dependency parsing and POS tagging models, and the key details which would be used in the self-training.

3.1 Dependency Parsing

We use the BiAffine dependency parsing model (Dozat and Manning, 2016) as the baseline parser, adapting it for cross-lingual parsing with multilingual BERT inputs (Devlin et al., 2019).

Input. An input sentence $w_1 \cdots w_n$ is fed directly into a pretrained multilingual BERT module. BERT would split each word into pieces. We adopt averaged pooling to obtain word-level representations from the piece-level outputs. The top-k layer outputs of the BERT are used, which are combined by a parameterized scalar vector into a single representation layer.[1] Finally, we obtain word-level representations $\boldsymbol{x}_1 \cdots \boldsymbol{x}_n$ by this process.

Encoder. The BiAffine dependency parsing simply adopts a three-layer BiLSTM as encoder, which can be formalized as:

$$\boldsymbol{h}_1^l \cdots \boldsymbol{h}_n^l = \text{BiLSTM}(\boldsymbol{h}_1^{l-1} \cdots \boldsymbol{h}_n^{l-1}), \tag{1}$$

where $l = \{1, 2, 3\}$, $\boldsymbol{h}_1^0 \cdots \boldsymbol{h}_n^0 = \boldsymbol{x}_1 \cdots \boldsymbol{x}_n$, and $\boldsymbol{h}_1^3 \cdots \boldsymbol{h}_n^3$ is our desired outputs.

Decoder. The BiAffine operation is used to calculate head and dependency label scores for each sentential word. Take head prediction as an example. First, two MLP layers are used to obtain the features for a word as head ($\boldsymbol{h}_1^{\text{head}} \cdots \boldsymbol{h}_n^{\text{head}}$) and child ($\boldsymbol{h}_1^{\text{child}} \cdots \boldsymbol{h}_n^{\text{child}}$), respectively. Then for each word w_i, we find its head word by calculating:

$$\boldsymbol{s}(w_i^{\curvearrowleft} w_j) = \text{BiAffine}(\boldsymbol{h}_i^{\text{child}}, \boldsymbol{h}_j^{\text{head}}), \tag{2}$$

where $j \in [1, n]/\{i\}$, and the highest-scored j is selected as the head for word w_i. For dependency relation prediction, we simply extend the scale $\boldsymbol{s}(w_i^{\curvearrowleft} w_j)$ into a vector $\boldsymbol{s}^{\text{rel}}(w_i^{\curvearrowleft} w_j)$, whose dim size equals the relation size. After the head word j is specified, we obtain the dependency relation label by the highest-scored index.

Dependency Probability. The probability for each dependency arc will be used as the confidence score in self-training. For each sentential word w_i, the probability of a given head j is calculated by:

$$p(w_i^{\curvearrowleft} w_j) = \frac{\exp(\boldsymbol{s}(w_i^{\curvearrowleft} w_j))}{\sum_{k \in [1,n]/\{i\}} \exp(\boldsymbol{s}(w_i^{\curvearrowleft} w_k))}. \tag{3}$$

[1] In this work, we set $k = 6$ and freeze BERT parameters according to the preliminary experiments.

The probability is computed in terms of words since the BiAffine decoder classifies heads at the word level. The conditional dependency relation probability $p(r_i|w_i, h_i)$ is computed similarly by softmax over $\boldsymbol{s}^{\text{rel}}(w_i \curvearrowright w_j)$. The reader is referred to as Dozat and Manning (2016) for more details.

3.2 POS Tagging

POS Tagging is exploited for two purposes related to self-training. On the one hand, we produce automatic POS tag inputs for automatic dependency parsing, as it is impractical to assume a very large corpus with gold-standard POS tags. On the other hand, we use the tagging model to rank auto-parsed dependency trees for instance selection. Here, we introduce the POS tagging model in detail, which is adapted from a typical BiLSTM POS tagger (Huang et al., 2015; Plank et al., 2016).

Input. Given a sentence $w_1 \cdots w_n$, we obtain $\boldsymbol{x}_1 \cdots \boldsymbol{x}_n$ by going through a multilingual BERT module, which is exactly the same as that of the dependency parsing model. The details can be found in the input part of Section 3.1 directly.

Encoder. For the encoder, we exploit PGN-BiLSTM (Jia et al., 2019) instead of a standard BiLSTM, taking the language ID as input to choose parameters for the BiLSTM module, which enables the model better capture the language differences.

For convenience, we formalize the standard BiLSTM by:

$$\boldsymbol{h}_1 \cdots \boldsymbol{h}_n = \text{BiLSTM}(\boldsymbol{x}_1 \cdots \boldsymbol{x}_n, \boldsymbol{V}), \tag{4}$$

where $\boldsymbol{V}$ denotes the flattened equivalent of all the BiLSTM parameters $\{\boldsymbol{W}_1 \cdots \boldsymbol{W}_K\}$. $\boldsymbol{V}$ can be implemented by $\boldsymbol{V} = \text{Vec}(\boldsymbol{W}_1) \oplus \cdots \oplus \text{Vec}(\boldsymbol{W}_K)$, where $\text{Vec}(\cdot)$ indicates vectorizing to reshape tensors into vectors, and $\oplus$ denotes concatenation.

In PGN-BiLSTM, we produce $\boldsymbol{V}$ dynamically according to the input language ID. Formally, the PGN-BiLSTM can be formalized as:

$$\begin{aligned} \boldsymbol{h}_1 \cdots \boldsymbol{h}_n &= \text{PGN-BiLSTM}(\boldsymbol{x}_1 \cdots \boldsymbol{x}_n, \boldsymbol{e}_{\text{lg}}) \\ &= \text{BiLSTM}(\boldsymbol{x}_1 \cdots \boldsymbol{x}_n, \boldsymbol{V}_{\text{lg}}), \\ &= \text{BiLSTM}(\boldsymbol{x}_1 \cdots \boldsymbol{x}_n, \boldsymbol{W}_{\text{pgn}}\boldsymbol{e}_{\text{lg}}), \end{aligned} \tag{5}$$

where $\boldsymbol{e}_{\text{lg}}$ is the embedding of the input language ID, and $\boldsymbol{W}_{\text{pgn}}$ is a meta model parameter of PGN-BiLSTM. In this way, we obtain different encoder parameters when the input language ID changes.

Decoder. Finally, the decoder consists of a single MLP layer:

$$\boldsymbol{o}_1 \cdots \boldsymbol{o}_n = \text{MLP}(\boldsymbol{h}_1 \cdots \boldsymbol{h}_n), \tag{6}$$

which is used to score all POS candidates directly for each word. The highest-scored tag index of each $\boldsymbol{o}_i$ is the final POS predictions.[2]

POS Probability. We also need to calculate POS probabilities for self-training. This is conducted straightforwardly by softmax since word-level prediction is used in our POS tagging model:

$$p(t|w_i, \text{lg}) = \frac{\exp(\boldsymbol{o}_{i,t})}{\sum \exp(\boldsymbol{o}_{i,*})}, \tag{7}$$

where t is the desired tag for word w_i.

4 Self-Training

The self-training framework for cross-lingual dependency parsing is as follows. First, a cross-lingual dependency parser (Section 3.1) trained on a source language corpus is used to parse the raw corpus of a target language. In particular, POS tags of the raw corpus are produced by a supervised cross-lingual POS tagger (Section 3.2). Next, we select a number of auto-parsed dependency trees from the outputs, and use them as the extra corpus to enhance the dependency parser. Instance selection is a key factor to the performance of self-training. We investigate two instance selection strategies based on the baseline dependency parser, and further suggest another alternative by using the cross-lingual POS tagger.

[2]We do not exploit CRF as its final impact on self-training is marginal while introduces addition calculation cost.

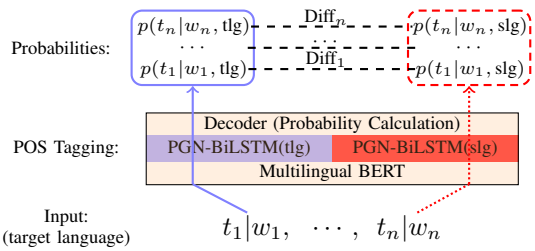


Figure 2: Illustration of the POS tagging guided instance selection, where the inner structures of the POS tagging model is described in Section 3.2, tlg and slg denote the target and source languages, respectively.

4.1 Strategies based on Dependency Parsing

Prediction Probability. The prediction probability is a widely-adopted strategy for instance selection in self-training (Yu et al., 2015; Zou et al., 2019), where auto-parsed dependency trees are ranked according to their tree probabilities, and the top probability trees are used for next-round training. Given a sentence $w_1 \cdots w_n$, assuming the output heads by our dependency parsing model are $h_1 \cdots h_n$, we calculate the score of the output dependency tree by the following formula:

$$s_{\text{prob}} = \prod_{i=1}^{n} p(w_i \curvearrowright w_{h_i}), \tag{8}$$

where $p(w_i \curvearrowright w_{h_i})$ is defined by Formula 3, which can be regarded as the confidence value of the current dependency arc.[3] We refer to this strategy as `prob` for simplicity.

Delta Probability. The second strategy is to use the delta value of the probabilities between the output head and the second-best head for each sentential word (Mejer and Crammer, 2012; Yu et al., 2015), where auto-parsed trees with larger delta values are selected for self-training.[4] For the sentence $w_1 \cdots w_n$, where the output heads and the second-best heads are $h_1 \cdots h_n$ and $h'_1 \cdots h'_n$, respectively, the selection score is defined by:

$$s_{\text{delta}} = \prod_{i=1}^{n} \left(p(w_i \curvearrowright w_{h_i}) - p(w_i \curvearrowright w_{h'_i})\right). \tag{9}$$

Note that there are cases where the final output head is not the highest-probability head because of the tree constraints, which are excluded directly. We use `delta` to denote this method for short.

4.2 POS Tagging Enhanced Criterion

Ranking the output sentences from the cross-lingual dependency parsing model itself may be biased, as it captures little knowledge on the differences between the source and target languages. Instead, the cross-lingual POS tagging model can offer such information, since it learns a universal model from the gold-standard training corpora of both the source and target languages. In addition, POS tagging is closely related to dependency parsing because they are both syntax-oriented, but POS tagging is much more light-weighted than dependency parsing, which makes our method more feasible in practice. Our goal is to select a target sentence which behaves highly similar across languages. We use these sentences to bridge the syntactic knowledge from the source into the target. Figure 2 illustrates the idea of the confidence computation strategy in detail.

Formally, given a target language sentence $w_1 \cdots w_n$, we first go through POS tagging as introduced in Section 3.2, feeding the target language ID into the PGN-BiLSTM encoder and computing the POS tagging probabilities of the best predictions $t_1 \cdots t_n$ at the word level by Equation 7. Then we compute another set of POS tagging probabilities by using the source language ID instead, feeding it into the PGN-BiLSTM encoder and computing the POS tagging probabilities of $t_1 \cdots t_n$. The process can be

[3] We do not use the relation probability for simplicity and meanwhile more importantly because it brings little influence.

[4] This is a simplified version of Yu et al. (2015).

regarded as by intentionally treating the target language sentence as a source language sentence. Finally, we obtain the confidence value for each sentence by:

$$\begin{aligned} \text{Diff}_i &= \|p(t_i|w_i, \text{tlg}) - p(t_i|w_i, \text{slg})\|, \\ s_{\text{pos}} &= \prod_{i=1}^{n}(1 - \text{Diff}_i), \end{aligned} \tag{10}$$

where the first equation indicates the language gaps, and the sentences with smaller gaps are chosen for self-training. We use `pos` to denote it for short.

4.3 Confidence-Aware Training of Dependency Parsing

Although with relatively high quality, the selected auto-parsed trees can nevertheless include noise. In order to address the influence of the noise, we introduce the confidence-aware training for the cross-lingual dependency parsing. The idea is inspired by Li et al. (2014), who solve parse ambiguities for monolingual self-training.

The standard training objective of the dependency parsing model mentioned in Section 3.1 is a cross-entropy loss over the dependency trees in the training corpus. Given a sentence $w_1 \cdots w_n$ and the corresponding dependency structure $(h_1, r_1) \cdots (h_n, r_n)$, where h and r indicate the head and dependency relation, respectively, the loss function is defined as follows:

$$\mathcal{L} = -\frac{\sum \log p(h_i, r_i|w_i)}{n}, \tag{11}$$

where $p(h_i, r_i|w_i) = p(w_i^{\frown} w_{h_i})p(r_i|w_i, h_i)$.

We use the word-level confidence values to regularize the loss function, which is defined by:

$$\mathcal{L}_{\text{conf}} = -\frac{\sum \tilde{p}(w_i^{\frown} w_{h_i}) \log p(h_i, r_i|w_i)}{n}, \tag{12}$$

where $\tilde{p}(w_i^{\frown} w_{h_i})$ is the confidence, defined by the dependency probability obtained from the original baseline dependency parsing model.

In particular, when the training corpus of the source and target languages is mixed to train a target language parser, we adopt a hyper-parameter α as the word-level confidence to rescale all the source language dependencies.

5 Experiments

5.1 Data and Settings

We conduct experiments on the Google Universal Dependency Treebanks (v2.2) (McDonald et al., 2013; Nivre et al., 2016) to verify the effectiveness of our models.[5] We adopt English as the source language. and choose eleven target languages, including German (de), Dutch (nl) and Swedish (sv) of the IE.Germanic family,[6] Spanish (es), French (fr) and Portuguese (pt) of the IE.Romance family, Polish (pl), Slovak (sk) and Slovenian (sl) of the IE.Romance family, and Estonian (et) and Finnish (fi) of the Uralic family. For each language, we use the same treebank type as Wang et al. (2019).[7]

We collect 500,000 raw sentences for each target language, respectively. The raw sentences are all selected from the Europarl v8 parallel corpus, which are download from the OPUS website directly. These sentences are already tokenized by the OPUS. We exclude the sentences shorter than 5 words or longer than 100 words, and then randomly sample 500,000 from the remaining.

For dependency parsing, we train models on the source English dataset and the auto-parsed dependency trees produced by self-training. During evaluation, gold POS tags are used as inputs on the test datsets for

[5] http://hdl.handle.net/11234/1-2837

[6] English also belongs to this family.

[7] The data statistics are omitted due to the space limitation.

Model.	IE.Germanic			IE.Romance			IE.Slavic			Uralic		AVG
	de	nl	sv	es	fr	pt	pl	sk	sl	et	fi	
Source Only												
baseline	75.31	75.22	81.35	78.35	81.51	78.84	79.80	72.08	72.22	69.30	72.15	76.01
Target Only												
prob	76.44‡	76.55‡	82.40‡	77.20↓	81.86	78.45↓	80.13	72.76	73.34‡	71.05	72.59	76.62
delta	76.85‡	76.29‡	82.67‡	77.31↓	82.11	78.07↓	80.22	72.60	73.48‡	71.24‡	72.51	76.67
pos	**77.52**‡	**76.74**‡	**83.20**‡	**78.54**	**82.35**‡	**79.17**	**80.85**‡	**73.32**‡	**73.71**‡	**71.64**‡	**73.51**‡	**77.32**
Δ(pos)	+2.21	+1.52	+1.85	+0.19	+0.84	+0.33	+1.05	+1.24	+1.49	+2.34	+1.36	+1.31
Standard Self-Training (Source + Target)												
prob	78.00‡	76.68‡	83.06‡	78.41	82.37‡	79.05	80.38	73.13‡	74.04‡	71.39‡	73.13‡	77.24
delta	77.85‡	76.54‡	83.23‡	78.53	82.14	79.52‡	80.17	73.37‡	74.09‡	71.50‡	73.22‡	77.29
pos	**78.45**‡	**77.22**‡	**83.58**‡	**79.42**‡	**82.80**‡	**80.01**‡	**80.70**‡	**73.74**‡	**74.21**‡	**72.04**‡	**73.94**‡	**77.83**
Δ(pos)	+3.14	+2.00	+2.23	+1.07	+1.29	+1.17	+0.90	+1.66	+1.99	+2.74	+1.79	+1.82

Table 1: Final UAS results, where the $\Delta(\cdot)$ rows show the improvements over the corresponding baseline without self-training, the negative results are marked with ↓, the results marked with ‡ denote that the p-value is less than 0.001 compared with the baseline by using the pairwise t-test.

all target languages, following the majority of the previous studies. We adopt the unlabeled attachment score (UAS) as the major evaluation metric (excluding the punctuations).[8]

For POS tagging, we train models on the combined dataset of the source English training corpus and the test corpus of each target language. Since gold-standard POS tags are already given as inputs for dependency parsing, it is fair and reasonable to adopt this setting. The POS tagging model is also used to tag raw corpus of the self-training for each language, which is a pre-requisite step for dependency parsing since no POS tag exists in the collected large-scale raw corpus.

There are several hyper-parameters in the neural dependency parsing and POS tagging models. We set them empirically according to previous work. For the input multilingual BERT, we exploit the BERT-Base Multilingual Cased version, where the output dimension size is 768.[9] The POS tag embedding size of the dependency parsing model is 100. The language embedding size of the POS tagging model is 4. The hidden sizes of various BiLSTMs for both parsing and tagging are all 400, and the hidden sizes of the two MLP layers in the dependency parsing model are both 600.

For training, we exploit batch learning with a batch size of 200 and Adam with a learning ratio of 0.002 to optimize the model parameters. Dropout is adopted by a rate of 0.33 for all neural modules except BERT. Since we assume only a test (no development) dataset for the target language, we stop the training after 8,000 iterations. We train each model five times and report the averaged results.

5.2 Results

First, our baseline dependency parsing model achieves a UAS of 96.75 and an LAS of 95.14 on the benchmark English Penn Treebank dataset (Stanford Dependencies v3.5.0) by using the base version of the English BERT, and a UAS of 93.38 and an LAS of 91.34 on the UDT dataset,[10] achieving state-of-the-art dependency parsing performance (Kondratyuk and Straka, 2019). However, when multilingual BERT is exploited, the performance shows a significant decrease, resulting in a UAS of 91.54 and an LAS of 89.30 on the UDT dataset. The observation indicates that monolingual training with language-specific BERT might be better than multilingual BERT.

The final result on the test datasets with self-training is shown in Table 1. 50,000 target language dependency trees are selected for training.[11] First, we focus on the models trained on the selected

[8] LAS is not given for the target languages to save space.

[9] https://github.com/google-research/bert

[10] The scores change very little by fine tuning the BERT.

[11] 50,000 is the closest setting to the best-performance models considering all settings and languages.

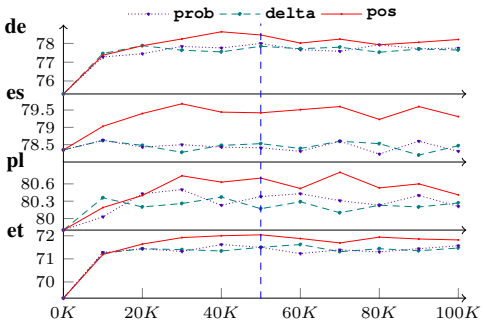


Figure 3: Impact of the selected sentence number.

automatic target dependency trees only, which indicates the effectiveness of the transferred knowledge by the target raw corpus. We list the performances in four groups according to the language family. In this setting, the strategy `prob` and `delta` can bring better performances on the majority languages, except on the language Spanish (es) and Portuguese (pt), which may be due to their differences with the English language making the transferring difficult.

Our final POS guided strategy `pos` can give consistently improved performances on all languages compared with the baseline, demonstrating that it is more effective than the `prob` and `delta` strategies. Although the improvements on all languages are better, the `pos` strategy also shows large variances among the eleven languages, which is similar to that of the `prob` and `delta` strategies. For the language Spanish (es) and Portuguese (pt), the improvements by using `pos` are also much smaller than the other languages. The observation indicates that the individual difference between the source and the target languages is a key factor for the effectiveness of knowledge transferring.

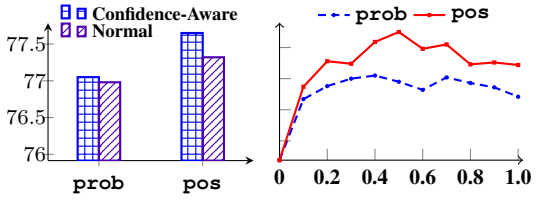
Further, we examine the standard setting of the self-training, merging the selected auto-parsed target dependency trees into the source English trees, and training target language dependency parsing models on both the source and target corpora. We set $\alpha = 0.4$ to reweigh the source English corpus. As shown in Table 1, there are great improvements compared with those of using only the target trees in the majority of cases. After the combination, all three instance selection strategies can obtain large gains. For the strategy `prob` and `delta`, marginal improvements can be obtained for the language Spanish (es) and Portuguese (pt) as well. Thus, self-training can bring improved performances for all the selected languages by using any of the three instance selection strategies, demonstrating the effectiveness of self-training. Overall, we obtain an averaged UAS improvement of $\frac{1.23+1.28+1.82}{3} = 1.44$ considering all selected eleven languages and all instance selection strategies.

We new look at the performances of self-training with the `pos` instance selection strategy in detail, which is used as our final model. As shown in Table 1, this model achieves the best performances on all languages. The final model can obtain an averaged increase of 1.82 UAS points over all the eleven languages, better than the other two strategies which are 1.23 and 1.28, respectively. In particular, the languages of the IE.Germanic family benefit the most from self-training, leading to an averaged improvement of $\frac{3.14+2.00+2.33}{3} = 2.46$ UAS points, which may be due to the same language family as the source English language. Similarly, the large variations (i.e., the best is 3.14, while the worst is 0.90) of the gains by our final model further demonstrate that the individual difference between the source and the target languages has a strong influence on the effectiveness of self-training.

5.3 Analysis

We choose four languages German (de), Spanish (es), Polish (pl) and Estonian (et) for further analysis, where one language is selected for each family.

Influence of the selected number. First, we examine the performance variations by the selected target dependency tree numbers. Figure 3 shows the tendency, where the start position with zero target tree is our baseline. When the number is surrounding 50,000, the UAS scores remain stable for all languages and instance selection strategies. The `pos` strategy gives more sustainable growth compared with the



(a) target reweighing (b) source reweighing

Figure 4: Impact of confidence-aware training.

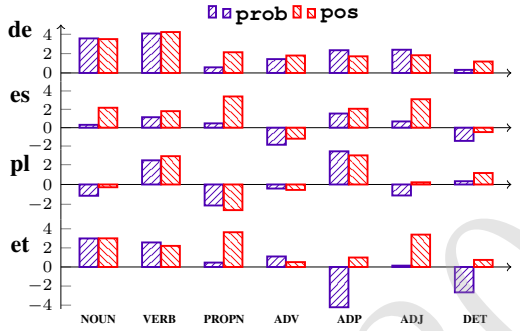


Figure 5: UAS variations with respect to POS tags.

`prob` and `delta` strategies, where the latter two show decreases when the number reaches 20,000. The observation again indicates that `pos` is more effective for instance selection. In addition, we find that `prob` and `delta` are highly similar. Averaged 90% of the selected sentences are identical by the two strategies, while the percentages are lower than 30% when compared to the `pos` strategy, respectively. Thus we exclude the `delta` strategy for the remaining analysis.

Impact of Confidence-Aware Training. Next, we test the effectiveness of confidence-Aware training. Our preliminary experimental results show that their influences are similar across all the four languages. Thus we average their performance to offer overall tendencies of the `prob` and `pos` instance selection strategies. Figure 4 shows the comparison results. For reweighing via the target dependency confidences, the `prob` strategy gains relatively little improvements compared with `pos`, which may be due to repeated information exploited. For source dependency reweighing, the performances remain stable in $[0.4, 0.7]$ for both strategies, resulting in increased UAS values by approximatively 0.3 compared with $\alpha = 1.0$. The observation demonstrates that confidence-aware training can give better performances for self-training.

Performances by POS tags. Further, we analyze the profit distributions of self-training with respect to different POS tags. The delta UAS values by different POS tags (only list seven popular tags) are shown in Figure 5. We see that self-training can not consistently improve the performances over all POS tags, especially for the languages which belong to a different family. By the fine-grained investigation, we can see further that the syntax characteristic of the target language is critical for self-training. The results further indicate that the individual difference between the source and the target languages is important, as mentioned in Section 5.2, as it may determine which kinds of syntax can be accurately captured by self-training. Given a target language, the highly-different syntax attributes might be difficult to learn, as self-training transfers syntax knowledge in a purely unsupervised way. For the language German (de), self-training can obtain better performance on all the seven popular POS tags, while for the other distant language to the English, there exist no consistent findings in more details despite the fact that we can obtain the overall improvements.

Performances by sentence lengths. Finally, we compare the performances in terms of sentence length. Figure 6 shows the results, where the sentence length is categorized into six bins. Overall, self-training

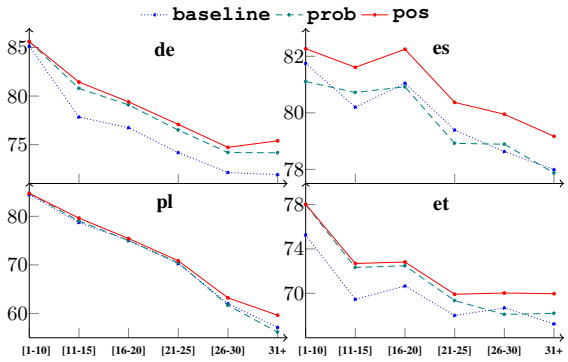


Figure 6: Performances by the sentence length.

brings consistently better performances over all sentence lengths on the four languages, which demonstrates the effectiveness further. We can see that the UAS decreases as a whole as the sentence length grows, which is reasonable since long sentences are difficult to parse (e.g., the head selection range is much larger). By examining the performance differences of the `prob` and `pos` in-depth, we find that `pos` gives larger improvements on longer sentences, which is possibly due to that `prob` tends to select shorter sentences (i.e., averaged 11.4 words compared with 15.2 words by `pos` when 50,000 sentences are selected).

6 Conclusions

We investigated self-training for unsupervised cross-lingual dependency parsing. A baseline dependency parser with multilingual BERT representations is trained and used to parse sentences of a target language and a set of the resulting dependency trees are selected to help training a target language dependency parser. We studied three different instance selection strategies, including two criteria by using the baseline dependency parser, and one criterion guided by a multilingual POS tagger. Results showed that self-training is effective in general for cross-lingual parsing. With the POS-assistant strategy, our final model brings the largest improvements, demonstrating the effectiveness of the method.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (NSFC No. 61976180 and 61602160), the Westlake University and Bright Dream Joint Institute for Intelligent Robotics.

References

Željko Agić, Anders Johannsen, Barbara Plank, Héctor Martínez Alonso, Natalie Schluter, and Anders Søgaard. 2016. Multilingual projection for parsing truly low-resource languages. *TACL*, 4:301–312.

Wasi Ahmad, Zhisong Zhang, Xuezhe Ma, Eduard Hovy, Kai-Wei Chang, and Nanyun Peng. 2019. On difficulties of cross-lingual transfer with order differences: A case study on dependency parsing. In *Proceedings of the NAACL*, pages 2440–2452.

Waleed Ammar, George Mulcaire, Miguel Ballesteros, Chris Dyer, and Noah A Smith. 2016. Many languages, one parser. *TACL*, 4:431–444.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th ACL*, pages 789–798.

Xilun Chen and Claire Cardie. 2018. Unsupervised multilingual word embeddings. In *Proceedings of the EMNLP*, pages 261–270.

Miryam de Lhoneux, Johannes Bjerva, Isabelle Augenstein, and Anders Søgaard. 2018. Parameter sharing between dependency parsers for related languages. In *Proceedings of the EMNLP*, pages 4992–4997.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL*, pages 4171–4186.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

Long Duong, Trevor Cohn, Steven Bird, and Paul Cook. 2015a. Cross-lingual transfer for unsupervised dependency parsing without parallel data. In *Proceedings of the CONLL*, pages 113–122.

Long Duong, Trevor Cohn, Steven Bird, and Paul Cook. 2015b. Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser. In *Proceedings of the 53rd ACL*, pages 845–850.

Kuzman Ganchev, Jennifer Gillenwater, and Ben Taskar. 2009. Dependency grammar induction via bitext projection constraints. In *Proceedings of the Joint Conference of ACL-IJCNLP*, pages 369–377.

Dan Goldwasser, Roi Reichart, James Clarke, and Dan Roth. 2011. Confidence driven unsupervised semantic parsing. In *Proceedings of the 49th ACL*, pages 1486–1495.

Jiang Guo, Wanxiang Che, David Yarowsky, Haifeng Wang, and Ting Liu. 2015. Cross-lingual dependency parsing based on distributed representations. In *Proceedings of ACL-IJCNLP*, pages 1234–1244.

Jiang Guo, Wanxiang Che, David Yarowsky, Haifeng Wang, and Ting Liu. 2016a. A distributed representation-based framework for cross-lingual transfer parsing. *JAIR*, 55:995–1023.

Jiang Guo, Wanxiang Che, David Yarowsky, Haifeng Wang, and Ting Liu. 2016b. A representation learning framework for multi-source transfer parsing. In *AAAI*.

Yulan He and Deyu Zhou. 2011. Self-training from labeled features for sentiment analysis. *Information Processing & Management*, 47(4):606–616.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Rebecca Hwa, Philip Resnik, Amy Weinberg, Clara Cabezas, and Okan Kolak. 2005. Bootstrapping parsers via syntactic projection across parallel texts. *Natural language engineering*, 11(3):311–325.

Chen Jia, Xiaobo Liang, and Yue Zhang. 2019. Cross-domain NER using cross-domain language modeling. In *Proceedings of the 57th ACL*, pages 2464–2474.

Eliyahu Kiperwasser and Yoav Goldberg. 2016. Simple and accurate dependency parsing using bidirectional lstm feature representations. *TACL*, 4:313–327.

Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing universal dependencies universally. In *Proceedings of the EMNLP-IJCNLP*, pages 2779–2795.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Zhenghua Li, Min Zhang, and Wenliang Chen. 2014. Ambiguity-aware ensemble training for semi-supervised dependency parsing. In *Proceedings of the 52nd ACL*, pages 457–467.

Xuezhe Ma and Fei Xia. 2014. Unsupervised dependency parsing with transferring distribution via parallel guidance and entropy regularization. In *Proceedings of ACL*, volume 1, pages 1337–1348.

David McClosky, Eugene Charniak, and Mark Johnson. 2006a. Effective self-training for parsing. In *Proceedings of the NAACL*, pages 152–159.

David McClosky, Eugene Charniak, and Mark Johnson. 2006b. Reranking and self-training for parser adaptation. In *Proceedings of the ACL*, pages 337–344.

Ryan McDonald, Slav Petrov, and Keith Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In *Proceedings of EMNLP*, pages 62–72.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, et al. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st ACL*, volume 2, pages 92–97.

Avihai Mejer and Koby Crammer. 2012. Are you sure? confidence in prediction of dependency tree edges. In *Proceedings of the NAACL*, pages 573–576.

Rada Mihalcea. 2004. Co-training and self-training for word sense disambiguation. In *Proceedings of the CoNLL*, pages 33–40.

Amir More, Amit Seker, Victoria Basmova, and Reut Tsarfaty. 2019. Joint transition-based models for morpho-syntactic parsing: Parsing strategies for MRLs and a case study from modern Hebrew. *TACL*.

Phoebe Mulcaire, Jungo Kasai, and Noah A. Smith. 2019. Polyglot contextual representations improve crosslingual transfer. In *Proceedings of the NAACL*, pages 3912–3918.

Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan T McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. 2016. Universal dependencies v1: A multilingual treebank collection. In *LREC*.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the LREC*.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th ACL*, pages 4996–5001.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. In *Proceedings of the 54th ACL*, volume 2, pages 412–418.

Emmanouil Antonios Platanios, Mrinmaya Sachan, Graham Neubig, and Tom Mitchell. 2018. Contextual parameter generation for universal neural machine translation. In *Proceedings of the EMNLP*, pages 425–435.

Mohammad Sadegh Rasooli and Michael Collins. 2015. Density-driven cross-lingual transfer of dependency parsers. In *Proceedings of EMNLP*, pages 328–338.

Mohammad Sadegh Rasooli and Michael Collins. 2017. Cross-lingual syntactic transfer with limited resources. *TACL*, 5:279–293.

Mohammad Sadegh Rasooli and Michael Collins. 2019. Low-resource syntactic transfer with unsupervised source reordering. *arXiv preprint arXiv:1903.05683*.

Roi Reichart and Ari Rappoport. 2007. Self-training for enhancement and domain adaptation of statistical parsers trained on small datasets. In *Proceedings of the 45th ACL*, pages 616–623.

Alexander M Rush, Roi Reichart, Michael Collins, and Amir Globerson. 2012. Improved parsing and pos tagging using inter-sentence consistency constraints. In *Proceedings of the EMNLP*, pages 1434–1444.

Piotr Rybak and Alina Wróblewska. 2018. Semi-supervised neural system for tagging, parsing and lematization. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual parsing from raw text to universal dependencies*, pages 45–54.

Kenji Sagae. 2010. Self-training without reranking for parser domain adaptation and its impact on semantic role labeling. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pages 37–44.

Kuniaki Saito, Yoshitaka Ushiku, and Tatsuya Harada. 2017. Asymmetric tri-training for unsupervised domain adaptation. In *Proceedings of the 34th ICML*, pages 2988–2997.

Michael Schlichtkrull and Anders Søgaard. 2017. Cross-lingual dependency parsing with late decoding for truly low-resource languages. In *Proceedings of the EACL*, volume 1, pages 220–229.

Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. 2019. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. In *Proceedings of the NAACL*, pages 1599–1613.

Samuel L Smith, David HP Turban, Steven Hamblin, and Nils Y Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *arXiv preprint arXiv:1702.03859*.

Oscar Täckström, Ryan McDonald, and Jakob Uszkoreit. 2012. Cross-lingual word clusters for direct transfer of linguistic structure. In *Proceedings of the NAACL*, pages 477–487.

Jörg Tiedemann and Zeljko Agić. 2016. Synthetic treebanking for cross-lingual dependency parsing. *JAIR*, 55:209–248.

Jörg Tiedemann, Željko Agić, and Joakim Nivre. 2014. Treebank translation for cross-lingual parser induction. In *Proceedings of the EMNLP*, pages 130–140.

Jörg Tiedemann. 2015. Improving the cross-lingual projection of syntactic dependencies. In *Proceedings of the 20th NODALIDA*, number 109, pages 191–199.

Yuxuan Wang, Wanxiang Che, Jiang Guo, Yijia Liu, and Ting Liu. 2019. Cross-lingual BERT transformation for zero-shot dependency parsing. In *Proceedings of EMNLP*, pages 5725–5731.

Michael Wick, Pallika Kanani, and Adam Pocock. 2016. Minimally-constrained multilingual embeddings via artificial code-switching. In *AAAI*.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of EMNLP-IJCNLP*, pages 833–844.

Min Xiao and Yuhong Guo. 2015. Annotation projection-based representation learning for cross-lingual dependency parsing. In *Proceedings of the CONLL*, pages 73–82.

Juntao Yu, Mohab Elkaref, and Bernd Bohnet. 2015. Domain adaptation for dependency parsing via self-training. In *Proceedings of the IWPT*, pages 1–10.

Yuan Zhang and Regina Barzilay. 2015. Hierarchical low-rank tensors for multilingual transfer parsing. In *Proceedings of EMNLP*, pages 1857–1867.

Yue Zhang and Joakim Nivre. 2011. Transition-based dependency parsing with rich non-local features. In *Proceedings of the 49th ACL*, pages 188–193.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2019. Cross-lingual dependency parsing using code-mixed TreeBank. In *Proceedings of the EMNLP-IJCNLP*, pages 997–1006.

Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. 2018. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of ECCV*, pages 289–305.

Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. 2019. Confidence regularized self-training. In *Proceedings of CVPR*, pages 5982–5991.

A Joint Model for Graph-based Chinese Dependency Parsing

Xingchen Li, Mingtong Liu, Yujie Zhang†, Jinan Xu, Yufeng Chen
School of Computer and Information Technology, Beijing Jiaotong University,
Beijing 100044, China
†yjzhang@bjtu.edu.cn

Abstract

In Chinese dependency parsing, the joint model of word segmentation, POS tagging and dependency parsing has become the mainstream framework because it can eliminate error propagation and share knowledge, where the transition-based model with feature templates maintains the best performance. Recently, the graph-based joint model (Yan et al., 2019) on word segmentation and dependency parsing has achieved better performance, demonstrating the advantages of the graph-based models. However, this work can not provide POS information for downstream tasks, and the POS tagging task was proved to be helpful to the dependency parsing according to the research of the transition-based model. Therefore, we propose a graph-based joint model for Chinese word segmentation, POS tagging and dependency parsing. We designed a charater-level POS tagging task, and then train it jointly with the model of Yan et al. (2019). We adopt two methods of joint POS tagging task, one is by sharing parameters, the other is by using tag attention mechanism, which enables the three tasks to better share intermediate information and improve each other's performance. The experimental results on the Penn Chinese treebank (CTB5) show that our proposed joint model improved by 0.38% on dependency parsing than the model of Yan et al. (2019). Compared with the best transition-based joint model, our model improved by 0.18%, 0.35% and 5.99% respectively in terms of word segmentation, POS tagging and dependency parsing.

1 Introduction

Chinese word segmentation, part-of-speech (POS) tagging and dependency parsing are three fundamental tasks for Chinese natural language processing, whose accuracy obviously affects downstream tasks such as semantic comprehension, machine translation and question-answering. The traditional method is usually following pipline way: word segmentation, POS tagging and dependency parsing. However, there are two problems of the pipline way, one is error propagation:incorrect word segmentation directly affects POS tagging and dependency parsing, another is information sharing: the tree tasks are strongly related, the label information of one task can help others, but the pipline way cannot exploit the correlations among the three tasks.

Using joint model for Chinese word segmentation, POS tagging and dependency parsing is a solution to these two problems. The previous joint models (Hatori et al., 2012; Zhang et al., 2014; Kurita et al., 2017) mainly adopted a transition-based framework to integrate the three tasks. Based on the standard sequential shift-reduce transitions, they design some extra actions for word segmentation and POS tagging. Although these transition-based models maintained the best performance of word segmentation, POS tagging and dependency parsing, its local decision problem led to the low precision of long-distance dependency parsing, which limited the precision of dependency parsing.

Different from the transition-based framework, the graph-based framework has the ability to make global decisions. Before the advent of neural network, the graph-based framework was rarely applied to the joint model due to its large decoding space to calculate.With the development of neural network

Proceedings of the 19th China National Conference on Computational Linguistics, pages 820-830,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

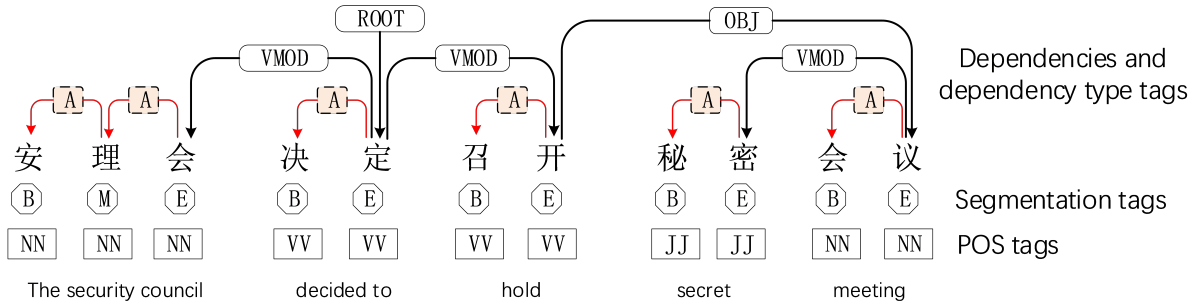


Figure 1: An example of a character-level dependency tree

technology, the graph-based method for dependency parsing improves rapidly and comes back into researchers' vision. Yan et al. (2019) firstly proposed a graph-based unified model for joint Chinese word segmentation and dependency parsing with neural network and attention mechanism, which is superior to the best transition-based joint model in terms of word segmentation and dependency parsing. This work without POS tagging task shows that dependency parsing task is beneficial to Chinese word segmentation.

Chinese word segmentation, POS tagging and dependency parsing are three highly correlated tasks and can improve each other's performance. Dependency parsing is beneficial to word segmentation and POS tagging, while word segmentation and POS tagging are also helpful to dependency parsing, which has been demonstrated by considerable work on the existing transition-based joint model of three tasks. We consider that joint POS tagging task can further improve the performance of dependency parsing. In addition, it makes sense of the model to provide POS information for downstream tasks. For these reasons, this paper proposes a graph-based joint model for word segmentation, POS tagging and dependency parsing. First, we design a character-level POS tagging task, and then combine it with a graph-based joint model for word segmentation and dependency parsing(Yan et al., 2019). As for the joint approach, this paper proposes two ways, one is to combine the two tasks by hard sharing parameters(Baxter, 1997) and the other is combine the two tasks by introducing tag attention mechanism in the shared parameter layer. Finally, we analyze our proposed models on the Chinese treebank (CTB5) dataset.

2 The Proposed Model

In this section, we introduce our proposed graph-based joint model for Chinese word segmentation, POS tagging and dependency parsing. Through the joint POS tagging task, we explore the joint learning method among multiple tasks and seek for a better joint model to improve the performance of Chinese dependency parsing further.

2.1 Character-level Chinese Word Segmentation and Denpendency Parsing

This paper refers to Yan et al. (2019)'s approach of combining word segmentation and dependency parsing into a character-level dependency parsing task. Firstly, we transform the word segmentation task to a special arc prediction problem between characters. Specifically, we treat each word as a dependency subtree, and the last character of the word is the root node, and for other characters, the next character is its head node. For example, the root node of the dependency subtree of the word "秘密" is "密", and the head node of the character "秘" is "密", which constitutes an intra-word dependency arc of "秘←密". To distinguish it from the dependencies between words, a special dependency label "Append(A)" was added to represent the dependencies between characters within a word. We use the last character in each word (the root node of the dependency subtree) as a representation of this word, and the dependency between words can be replaced by the dependency between last characters of each word. For example, the dependency relationship "安理会←决定" is transformed into "会←定". Figure 1 shows an example of CTB5 dataset being converted to a character-level dependency tree.

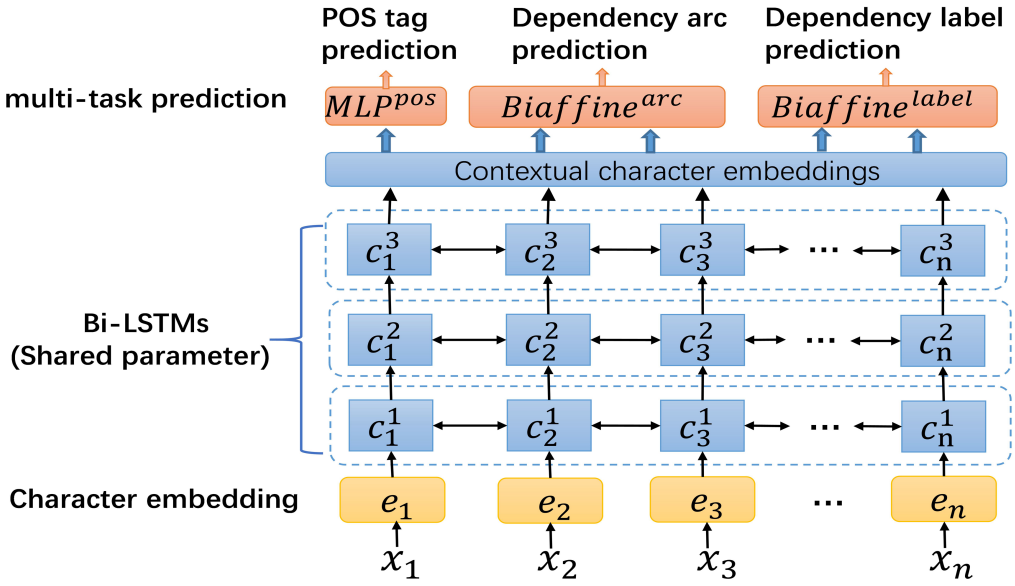


Figure 2: A joint model of segmentation, POS tagging and dependency parsing with parameter sharing

2.2 Character-level POS Tagging

In order to transform the POS tagging into a character-level task, this paper adopts the following rules to convert the POS tag of words into POS tag of each character: the POS tag of each character is the POS tag of the word it is in. In predicting word's POS tag, it is represented by the POS tag of last character of the word. For example, if the predicted POS tag sequence of the word "安理会" is "NN, VV, NN", then the POS tag "NN" of the last character "会" is taken as the POS tag prediction result of the whole word. It is important to note that a word's POS tag is predicted correctly only if the word segmentation is predicted correctly and the last charater's POS tag is also predicted correctly.

2.3 Graph-based Joint Model for Word Segmentation, POS Tagging and Dependency Parsing

According to sections 2.1 and 2.2, after converting three tasks into two character-level tasks, we designed a shared deep Bi-LSTM network to encode the input characters and obtain contextual character vectors. As shown in figure 2, given the input sentence (character sequence) $X = \{x_1, ..., x_n\}$. Firstly, vectorize each character x_i to get vector e_i, which consists of two parts, one is pre-trained vector p_i which is fixed during training, and the other is randomly initializing embeddings s_i which can be adjusted in training. Element-wise adds the pre-trained and random embeddings as the final input characters' embedding e_i, that is $e_i = p_i + s_i$. Then we feed the characters' embedding into multi-layer Bi-LSTM network, and get each character's contextual representation $C = \{c_1, ..., c_n\}$.

$$\overrightarrow{c_i} = \overrightarrow{\text{LSTM}}(e_i, \overrightarrow{c}_{i-1}, \overrightarrow{\theta}); \ \overleftarrow{c_i} = \overleftarrow{\text{LSTM}}(e_i, \overleftarrow{c}_{i+1}, \overleftarrow{\theta}); \ c_i = \overrightarrow{c_i} \oplus \overleftarrow{c_i} \tag{1}$$

After the contextual character vectors are obtained, the character-level POS tagging and dependency parsing are carried out respectively. We adopted the graph-based framework to analyze the character-level dependency parsing task. By taking each character as a node on the graph, and taking the possibility of forming a dependency relationship between characters as a probability directed edge between nodes (from the head node points to the dependency node), we can define dependency parsing as finding a dependency tree with the highest probability that conforms to the dependency grammar on a directed complete graph. The process of dependency parsing contains two subtasks: prediction of dependency relationship and prediction of dependency relationship type.

Prediction of dependency relationship: We use $x_i \leftarrow x_j$ to represent the dependency relation between x_i as the dependency node and x_j as the head node. After context encoding, each character obtains a vector representation c_i. Considering that each character has the possibility of being a dependency node

and a head node, we use two vectors d_i^{arc} and h_i^{arc} to represent them respectively, and get them from c_i through two different MLP, as shown in formula(2).

$$d_i^{arc} = \text{MLP}_d^{arc}(c_i);\ h_i^{arc} = \text{MLP}_h^{arc}(c_i) \tag{2}$$

To calculate the probability s_{ij}^{arc} of $x_i \leftarrow x_j$, we use biaffine attention mechanism proposed by Dozat and Manning (2016).

$$s_{ij}^{arc} = \text{Biaffine}^{arc}(h_j^{arc}, d_i^{arc}) = h_j^{arc} U^{arc} d_i^{arc} + h_j^{arc} u^{arc} \tag{3}$$

where U^{arc} is a matrix whose dimension is (d_c, d_c), and the d_c is the dimension of vector c_i, u^{arc} is a bias vector. After we get the scores of all head nodes of the i-th character, we select the max score node as its head.

$$s_i^{arc} = [s_{i1}^{arc}, ..., s_{in}^{arc}];\ y_i^{arc} = \arg\max(s_i^{arc}) \tag{4}$$

Prediction of dependency relatoinship type: After obtaining the best predicted unlabeled dependency tree, we calculate the label scores s_{ij}^{label} for each dependency relationship $x_i \leftarrow x_j$. In our joint model, the arc labels set consists of the standard word-level dependency labels and a special label "A" indicating the intra-dependency within a word. We also use two vectors d_i^{label} and h_i^{label} to represent them respectively, and get them from c_i through two different MLP, and we use another biaffine attention network to calculate the label scores s_{ij}^{label}.

$$d_i^{label} = \text{MLP}_d^{label}(c_i);\ h_i^{label} = \text{MLP}_h^{label}(c_i) \tag{5}$$

$$s_{ij}^{label} = \text{Biaffine}^{label}(h_j^{label}, d_i^{label}) = h_j^{label} U^{label} d_i^{label} + (h_j^{label} \oplus d_i^{label}) V^{label} + b \tag{6}$$

where U^{arc} is a tensor whose dimension is (k, d_c, d_c), k is the number of dependency relationship labels, and V^{label}'s dimension is $(k, 2d_c)$, and b is a bias vector. The best label of the dependency relationship $x_i \leftarrow x_j$ is:

$$y_{ij}^{label} = \arg\max(s_{ij}^{label}) \tag{7}$$

Prediction of POS tagging: We use multi-layer perceptron (MLP) to calculate the probability distribution of the POS tag for each character.

$$s_i^{POS} = \text{MLP}^{POS}(c_i) \tag{8}$$

The best POS tag of the character x_i is

$$y_i^{POS} = \arg\max(s_i^{POS}) \tag{9}$$

Loss function for joint model: For the three tasks described above, we adopt cross-entropy loss for all of them, and the results are denoted as $Loss_{arc}, Loss_{dep}, Loss_{pos}$ respectively. The common way to deal with the loss of multiple tasks is to add them together, but this way does not balance the loss of each task. Therefore, we adopt the method proposed by Kendall et al. (2018), that is using uncertainty to weigh losses for three tasks.

$$\mathcal{L}(\theta) = \frac{1}{\delta_{arc}^2} Loss_{arc} + \frac{1}{\delta_{dep}^2} Loss_{dep} + \frac{1}{\delta_{pos}^2} Loss_{pos} + log\delta_{arc}^2 + log\delta_{dep}^2 + log\delta_{pos}^2 \tag{10}$$

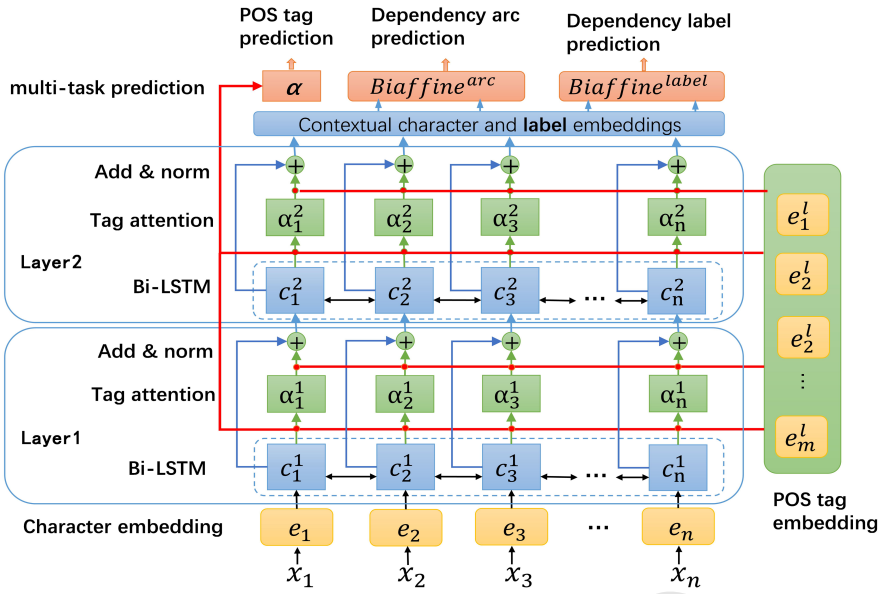


Figure 3: A joint model of segmentation, POS tagging and dependency parsing with tag attention mechanism

2.4 Introduction of Tag Attention Mechanism

The above model joint the three tasks through sharing Bi-LSTM layers to encode the contextual character's information. However, there is no explicit representation of the POS information in the shared encoding layers, the POS tagging task cannot provide the predicted information for word segmentation and dependency parsing. Therefore, we introduce the vector representation of the POS tag and propose the tag attention mechanism (TAM) to integrate the POS information of contextual characters into the vector representation of each character, so that the POS information of the contextual character can also be used in the word segmentation and dependency parsing. This structure is similar to the hierarchically-refined label attention network (LAN) proposed by Cui and Zhang (2019), but we use it to obtain POS information of each layer for subsequent character-level dependency parsing tasks. LAN differs from TAM in that LAN only predicts at the last layer while TAM predicts at each layer. We have tried to predict only at the last layer, but the result of segmentation and dependency parsing is slightly lower than predicting at each layer. The model is shown in figure 3.

Firstly, we vectorize the POS tags. Each POS tag is represented by a vector e_i^t, and the represents of the set of POS tags denoted as $E^t = \{e_1^t, ..., e_m^t\}$, which is randomly initialized before model training, and then is adjusted during the model training. Then, we calculate the attention weight between the contextual character vectors and POS tag vectors:

$$\alpha = \text{softmax}(\frac{QK^T}{\sqrt{d_c}}) \tag{11}$$

$$E^+ = \text{Attention}(Q, K, V) = \alpha V \tag{12}$$

$$C^+ = \text{LayerNorm}(C + E^+) \tag{13}$$

where Q, K, V are matrices composed of a set of queries, keys and values. We set $Q = C, K = V = E^t$. The i-th line of α represents the POS tag probability distribution of the i-th character of the sentence. According to this probability distribution α, we calculate the representation of predicted POS tag of each character of the sentence, and it is denoted as E^+. The E^+ is added to the contextual vectors C as the POS tag information. After layer normalization((Ba et al., 2016), we can obtain the character vectors (C^+) containing the POS information, and then take it as the input of the next Bi-LSTM layer. After

the second layer of Bi-LSTM encoding, each character vector we get will contain every characters' POS information, which can be used by word segmentation and dependency parsing.

When the tag attention mechanism is applied, the i-th line of the calculated attention weight for each layer is the POS tag distribution of the i-th character. Different from the prediction method of POS tagging in previous model, we added the attention weights of all layers as the final POS tag distribution:

$$s_i^{POS} = \sum_j^m \alpha_i^j \tag{14}$$

where, m is the number of layers. The prediction of POS tag is:

$$y_i^{POS} = \arg\max(s_i^{POS}) \tag{15}$$

For word segmentation and dependency parsing, we use the same approach as the previous model. For the losses of three tasks, we also use the same way to calculate it as the previous model.

3 Experiment

3.1 Dataset and Evaluation Metrics

We conducted experiments on the Penn Chinese Treebank5 (CTB-5). We adopt the data splitting method as same as previous works (Hatori et al., 2012; Kurita et al., 2017; Yan et al., 2019). The training set is from section 1∼270, 400∼931 and 1001∼1151, the development set is from section 301∼325, and the test set is from section 271∼300. The statistical information of the data is shown in Table 1.

Dataset	Sentence	word	character
Training	16k	494k	687k
Develop	352	6.8K	31k
Test	348	8.0k	81k

Table 1: The statistics of the dataset.

Following previous works (Jiang et al., 2008; Kurita et al., 2017; Yan et al., 2019), we use standard measures of word-level F1 score to evaluate word segmentation, POS tagging and dependency parsing. F1 score is calculated according to the precision P and the recall R as $F = 2PR/(P + R)$ (Jiang et al., 2008). Dependency parsing task is evaluated with the unlabeled attachment scores excluding punctuations. The output of POS tags and dependency arcs cannot be correct unless the corresponding words are correctly segmented.

3.2 Model Configuration

We use the same Tencent's pre-trained embeddings (Song et al., 2018) and configuration as Yan et al. (2019), and the dimension of character vectors is 200. The dimension of POS tag vectors is also 200. We use with 400 units for each Bi-LSTM layer and the layer numbers is 3. Dependency arc MLP output size is 500 and the label MLP output size is 100.The dropout rates are all 0.33.

The models are trained with Adam algorithm (Kingma and Ba, 2014) to minimize the total loss of the cross-entropy of arc predictions, label predictions and POS tag predictions, which using uncertainty weights to combine losses. The initial learning rate is 0.002 annealed by multiplying a fix decay rate 0.75 when parsing performance stops increasing on development sets. To reduce the effects of "gradient exploding", we use gradient clip of 5.0 (Pascanu et al., 2013). All models are trained for 100 epochs.

3.3 Results

We conduct comparison of our models with other joint parsing models. The model shown in figure 2 is denoted as Ours and the model shown in figure 3 as Ours-TAM (with tag attention mechanism). The comparison models include three types: one is the transition-based joint models with feature templates

Model	Framework	SEG	POS	DEP
Hatori et al. (2012)	Transition	97.75	94.33	81.56
Zhang et al. (2014)	Transition	97.67	94.28	**81.63**
Kurita et al. (2017)	Transition	**98.24**	**94.49**	80.15
Kurita et al. (2017)(4-gram)	Transition	97.72	93.12	79.03
Kurita et al. (2017)(8-gram)	Transition	97.70	**93.37**	79.38
Yan et al. (2019)[0]	Graph	**98.47**	—	**87.24**
Ours	Graph	98.34	94.60	**87.91**
Ours-TAM	Graph	**98.42**	**94.84**	87.62

Table 2: Performance comparison of Chinese dependency parsing joint models.

(Hatori et al., 2012; Zhang et al., 2014; Kurita et al., 2017), the other is the transition-based joint models with neural network (Kurita et al., 2017)(4-gram, 8-gram), and the third is the graph-based model with neural network without POS tagging task (Yan et al., 2019). The results are shown in table 2 [0].

From the table, we see that transition-based joint models using feature templates maintain the best performance in word segmentation, POS tagging and dependency parsing for a long time. Although Kurita et al. (2017)(4-gram, 8-gram) adopted the neural network approach, it still didn't surpass the joint model with feature templates. While, the graph-based joint model (Yan et al., 2019) obtained the better performance in word segmentation and dependency parsing than all transition-based model.

Our models Ours and Ours-TAM exceeded Yan et al. (2019) 0.67 and 0.38 percentage points respectively in dependency parsing, indicating that the POS tag information contributes to dependency parsing. Although they are 0.13 and 0.05 percentage points lower than Yan et al. (2019) on word segmentation task respectively, they still exceed the best transtion-based joint model with feature templates (Kurita et al., 2017). Yan et al. (2019) does not have POS tagging task, but our models have, and its performance exceeded that of the previous best joint model (Kurita et al., 2017) by 0.11 and 0.35 percentage points respectively, indicating that after the introduction of POS tagging, other tasks such as dependency parsing are also helpful for POS tagging task itself. Compared to the best transition-based joint model, our model improves on all three tasks, indicating that the graph-based model using neural network is superior to the transition-based model in word segmentation, POS tagging and dependency parsing.

3.4 Detailed Analysis

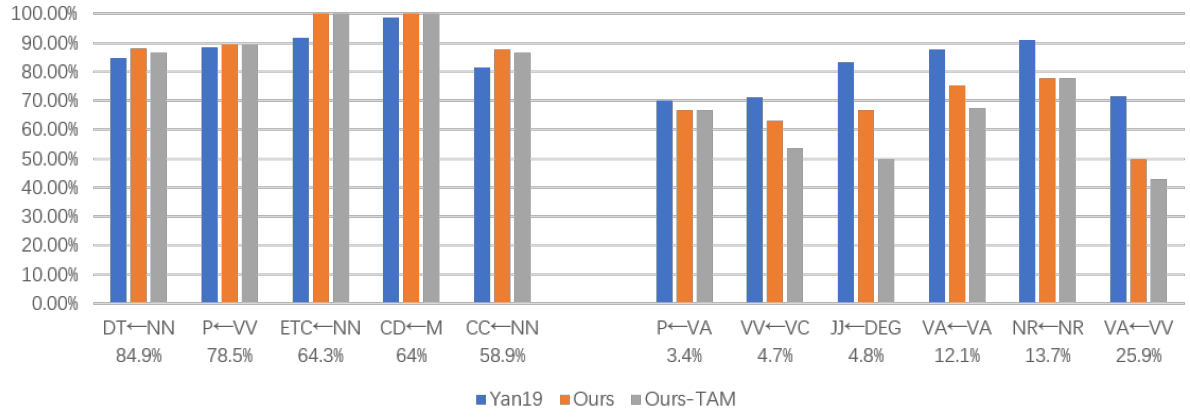


Figure 4: Comparison of precision on different POS tag patterns before and after joint POS tagging task

[0]Yan et al. later submitted an improved version (Yan et al., 2020), and the results of word segmentation and dependency parsing reached 98.48 and 87.86, respectively.

We will further investigate the reasons for the improvement of dependency parsing after the combination of POS tagging task. For a dependency relationship $x_i \leftarrow x_j$, we use $X \leftarrow Y$ to represent its POS dependency pattern, the X is the POS tag of x_i, and the Y is the POS tag of x_j. We calculated the distribution of Y for each X in training set and found that the probability between some X and Y was very high. For example, when X was P(preposition), the distribution of Y was {VV(78.5%), DEG(5.1%), ..., NN(3.1%), ... }. In order to verify whether our models can use these POS informations in training dataset, we calculated the accuracy of each POS dependency patterns in test dataset on our models and the re-implemented model of Yan et al. (2019). The patterns on which the accuracy of our models are better than Yan et al. (2019) are shown in left part of figure 4.

Table 3: Head POS distribution

Node POS	Head POS distribution					
DT	NN 84.9%	VV 7.5%	DEG 1.8%	P 1.3%	M 1%	NR 0.8%
P	VV 78.5%	DEG 5.1%	VA 3.4%	VE 3.3%	VC 3.1%	NN 3.1%
ETC	NN 64.3%	NR 22.5%	VV 10.4%	VA 1.6%	VE 0.2%	VC 0.2%
CD	M 64%	NN 20.6%	VV 6.7%	CD 2.7%	DT 1.6%	DEG 1.2%
CC	NN 58.9%	VV 20.5%	NR 7.9%	NT 2.3%	VA 2.1%	M 1.9%

The X of these 5 patterns are {DT, P, ETC, CD, CC}, and the Y's distributions of each X are shown in the table 3. It is found that all 5 patterns select Y with the highest probability, indicating that our model can fully utilize the POS informations to improve the accuracy of dependencies with these POS dependency patterns.

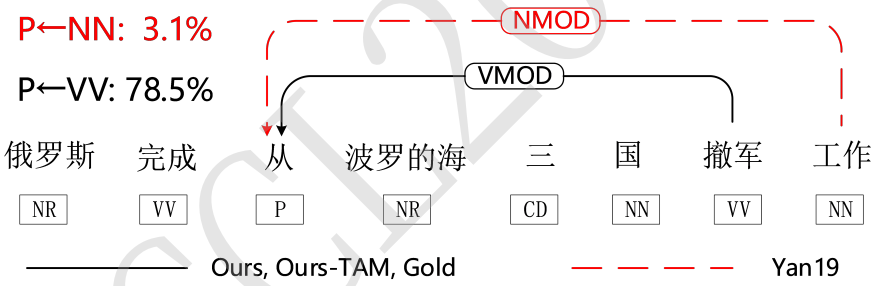
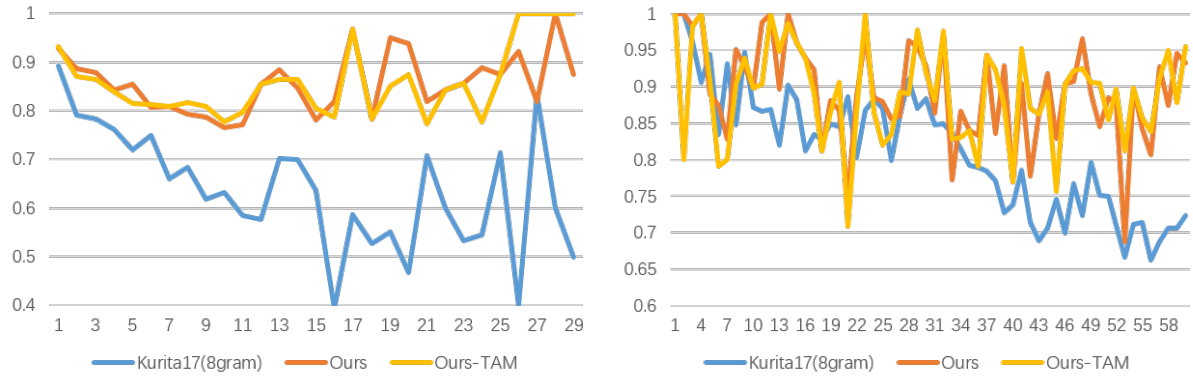


Figure 5: An example of POS information contributes to dependency parsing

As the example shown in the figure 5, when predicting the head node of "从", Yan et al. (2019) predicted wrong node "工作", while our models both predicted right node "撤退". The POS tag of "从" is P and the POS tag of correct head node "撤退" is VV whose probability is 78.5%, while the wrong head node 工作' s POS tag is NN whose probability is only 3.1%. Because our models can use these POS informations to exclude the candidate head nodes of low probability POS, thus improving the performance of dependency parsing.

Although Ours-TAM achieved better results in segmentation and POS tagging, the dependency parsing was reduced compared with Ours. The right part of the figure 4 shows the patterns on which the accuracy of our models are worse than Yan et al. (2019). It can be found that the dependency probability of these patterns is small, and the addition of POS information actually reduces the accuracy. Therefore, Ours-TAM has better POS information, so the accuracy of these patterns is lower than Ours, thus the overall precision of dependency parsing of Ours-TAM decreases compared with that of Ours.

Next, we will investigate the difference between the graph-based joint model and the transition-based joint model in dependency parsing. We compare our graph-based joint models to the transition-based joint model (Kurita et al., 2017) according to dependency length and sentence length respectively. The results are shown in figure 6. From the figure, we can see that our proposed joint models on long-distance dependencies have obvious advantages, and the accuracy of the dependency parsing is relatively stable



(a) Dependency length

(b) Sentence length

Figure 6: The influence of Depedency length and Sentence length on dependency parsing

with the increase of sentence length, while the transiton-base joint model has an obvious downward trend, which indicates that our graph-based joint model can predict the long-distance dependencies more effectively than transition-based joint model.

4 Related work

Hatori et al. (2012) proposed a character-level dependency parsing for the first time, which combines word segmentation, POS tagging and dependency parsing, They combined the key feature templates on the basis of the previous feature engineering research on the three tasks, and realized the synchronous processing of the three tasks. Zhang et al. (2014) annotated the internal structure of words, and regarded the word segmentation task as dependency parsing within characters to jointly process with three tasks. Kurita et al. (2017) firstly applied neural network to the charater-level dependency parsing. Although these transition-based joint models achieved best accuracy in dependency parsing, they still suffer from the limitation of local decision.

With the development of neural network, the graph-based dependency parsing models (Kiperwasser and Goldberg, 2016; Dozat and Manning, 2016) using neural networks have developed rapidly. those model fully exploit the ability of the bidirectional long short-term memory network (Bi-LSTM) (Hochreiter and Schmidhuber, 1997) and attention mechanism (Bahdanau et al., 2014; Vaswani et al., 2017) to capture the interactions of words in a sentence. Different from transition-based models, the graph-based model can make global decision when predicting dependency arcs, but few joint model adopted this framework. Yan et al. (2019) firstly proposed a joint model adopting graph-based framework with neural network for Chinese word segmentation and dependency parsing, but they does not use POS tag.

According to the research of existing transition-based joint model, the word segmentation, POS tagging and dependency parsing are three highly correlated tasks that influence each other. Dependency parsing is beneficial to word segmentation and POS tagging, while word segmentation and POS tagging are also helpful to dependency parsing. Therefore, we consider that integrating POS tagging task into graph-based joint model (Yan et al., 2019) to further improve the performance of joint model and to provide POS information for downstream tasks. We transform the POS tagging task into a character-level sequence labeling task and then we joint the word segmentation and dependency parsing into a graph-based framework, and then combine the two character-level tasks into a multi-task models. There are many multi-task learning approaches such as Baxter (1997), Misra et al. (2016), Long and Wang (2015) and Hashimoto et al. (2016), we use parameter sharing (Baxter, 1997) to realize the joint model, and then improve it with tag attention mechanism. Finally, we analyze the models on the CTB5 dataset.

5 Conclusion

This paper proposed the graph-based joint model for Chinese word segmentation, POS tagging and dependency parsing. The word segmentation and dependency parsing are transformed into a character-level dependency parsing task, and the POS tagging task is transformed into a character-level sequence labeling task, and we use two ways to joint them into a multi-task model. Experiments on CTB5 dataset show that the combination of POS tagging task is beneficial to dependency parsing, and using the POS tag attention mechanism can exploit more POS information of contextual characters, which is beneficial to POS tagging and dependency parsing, and our graph-based joint model outperforms the existing best transition-based joint model in all of these three tasks. In the future, we will explore other joint approaches to make three tasks more mutually reinforcing and further improve the performance of three tasks.

Acknowledgements

We are grateful for helpful comments and suggestions from the anonymous reviewers. This work is supported by the National Nature Science Foundation of China (Contract 61876198, 61976015, 61976016).

References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Jonathan Baxter. 1997. A bayesian/information theoretic model of learning to learn via multiple task sampling. *Machine learning*, 28(1):7–39.

Leyang Cui and Yue Zhang. 2019. Hierarchically-refined label attention network for sequence labeling. *arXiv preprint arXiv:1908.08676*.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. 2016. A joint many-task model: Growing a neural network for multiple nlp tasks. *arXiv preprint arXiv:1611.01587*.

Jun Hatori, Takuya Matsuzaki, Yusuke Miyao, and Jun'ichi Tsujii. 2012. Incremental joint approach to word segmentation, pos tagging, and dependency parsing in chinese. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 1045–1053. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Wenbin Jiang, Liang Huang, Qun Liu, and Yajuan Lü. 2008. A cascaded linear model for joint chinese word segmentation and part-of-speech tagging. In *Proceedings of ACL-08: HLT*, pages 897–904.

Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Eliyahu Kiperwasser and Yoav Goldberg. 2016. Simple and accurate dependency parsing using bidirectional lstm feature representations. *Transactions of the Association for Computational Linguistics*, 4:313–327.

Shuhei Kurita, Daisuke Kawahara, and Sadao Kurohashi. 2017. Neural joint model for transition-based chinese syntactic analysis. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1204–1214.

Mingsheng Long and Jianmin Wang. 2015. Learning multiple tasks with deep relationship networks. *arXiv preprint arXiv:1506.02117*, 2:1.

Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. 2016. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3994–4003.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318.

Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. 2018. Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 175–180.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Hang Yan, Xipeng Qiu, and Xuanjing Huang. 2019. A unified model for joint chinese word segmentation and dependency parsing. *arXiv preprint arXiv:1904.04697*.

Hang Yan, Xipeng Qiu, and Xuanjing Huang. 2020. A graph-based model for joint chinese word segmentation and dependency parsing. *Transactions of the Association for Computational Linguistics*, 8:78–92.

Meishan Zhang, Yue Zhang, Wanxiang Che, and Ting Liu. 2014. Character-level chinese dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1326–1336.

Semantic-aware Chinese Zero Pronoun Resolution with Pre-trained Semantic Dependency Parser

Lanqiu Zhang
Beijing Language and Culture University
zhang_lanqiu@163.com

Zizhuo Shen
Beijing Language and Culture University
blcushzz@gmail.com

Yanqiu Shao✉
Beijing Language and Culture University
yqshao163@163.com

Abstract

Deep learning-based Chinese zero pronoun resolution model has achieved better performance than traditional machine learning-based model. However, the existing work related to Chinese zero pronoun resolution has not yet well integrated linguistic information into the deep learning-based Chinese zero pronoun resolution model.This paper adopts the idea based on the pre-trained model, and integrates the semantic representations in the pre-trained Chinese semantic dependency graph parser into the Chinese zero pronoun resolution model. The experimental results on OntoNotes-5.0 dataset show that our proposed Chinese zero pronoun resolution model with pre-trained Chinese semantic dependency parser improves the F-score by 0.4% compared with our baseline model, and obtains better results than other deep learning-based Chinese zero pronoun resolution models. In addition, we integrate the BERT representations into our model so that the performance of our model was improved by 0.7% compared with our baseline model.

1 Introduction

Chinese zero pronoun resolution is a special task of coreference resolution (Zhao and Ng, 2007). Its purpose is to find the real referent of the omitted parts with syntactic functions in the text. These omitted parts are usually called zero pronouns, and their real referents are called antecedents. Below is a sentence with zero pronouns:

[我]之前没有听说过[她]，[$*pro_1*$]听说[$*pro*_2$]是个有才华的美女。(I have not heard of [her] before, [$*pro_1*$] heard that [$*pro_2*$] is a talented beauty.)

In this example, the referent of zero pronoun $*pro_1*$ is “我/I”, and the referent of zero pronoun $*pro_2*$ is “她/her”. Since the zero pronoun is not a real word in the text, its resolution is much more difficult than that of the overt pronoun. The existence of zero pronouns poses challenges for machines to automatically understand text.

The existing Chinese zero pronoun resolution models with better performance usually adopt the method of deep learning (Chen and Ng, 2016);(Liu et al., 2017);(Yin et al., 2017);(Yin et al., 2018a);(Yin et al., 2018b). The deep learning-based methods can make the model automatically extract the task-related distributed representations through end-to-end training, thereby avoiding the problem that traditional machine learning-based methods rely heavily on artificially designed feature templates (Chen and Ng, 2016). However, it is difficult for deep learning-based models to encode effective syntactic, semantic and other linguistic information only through end-to-end training. Many deep learning-based Chinese zero pronoun resolution models still use syntactic features extracted from the syntactic parsing tree as a supplement to distributed representations.

Intuitively, semantic information as a higher level linguistic information is also very important to the Chinese zero pronoun resolution task, however few studies have attempted to integrate semantic information into the Chinese zero pronoun resolution model. Therefore, how to effectively integrate semantic information into the Chinese zero pronoun resolution model is a challenging problem. With the development of semantic parsing, the performance of some sentence-level semantic parsers have made

Proceedings of the 19th China National Conference on Computational Linguistics, pages 831-841, Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

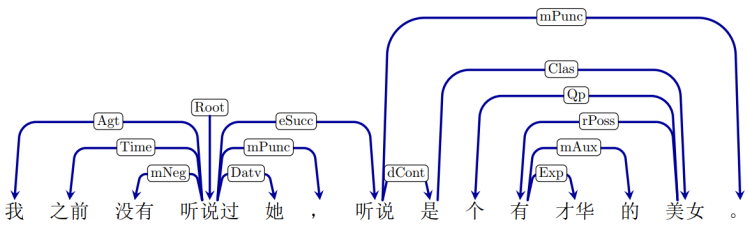


Figure 1: An example of a Chinese semantic dependency graph

remarkable progress, which provides opportunities for the application of sentence-level semantic parsing in other natural language processing tasks.

In this paper, we proposed a semantic-aware Chinese zero pronoun resolution model that integrates the semantic information from pre-trained Chinese semantic dependency graph parser. Chinese semantic dependency graph parsing (Che et al., 2016) is a semantic-level dependency parsing task, which is an extension of syntactic dependency parsing. Each node in the semantic dependency graph represents a word in the sentence, and the nodes are connected by directed edges with semantic relationship labels. Figure 1 is an example of a Chinese semantic dependency graph.

The realization of our model requires two stages. In the first stage, we use the Chinese semantic dependency graph parsing as a pre-training task to obtain a pre-trained semantic dependency graph parser. In the second stage, we feed the sentence which will be processed into the pre-trained semantic dependency graph parser to obtain the semantic-aware representations, and integrate these implicit semantic information into the Chinese zero pronoun resolution model.

We implement a attention-based Chinese zero pronoun resolution model as our baseline model. The experiments on OntoNotes-5.0 dataset show that our proposed Chinese zero pronoun resolution model with pre-trained Chinese semantic dependency parser improves the F-score by 0.4% compared with our baseline model, and obtains better results than other deep learning-based Chinese zero pronoun resolution models. In addition, we integrate the BERT representations into our model so that the performance of our model was improved by 0.7% compared with our baseline model.

2 Related Work

2.1 Zero Pronoun Resolution

Methods for solving Chinese zero pronoun resolution include rule-based methods, traditional machine learning-based methods, deep learning-based methods, etc. Converse (P and S, 2006) used Hobbs algorithm to traverse the syntactic tree of sentences to find the referent of zero pronoun. Zhao et al. (Zhao and Ng, 2007) designed more effective manual features for Chinese zero pronoun resolution task, and adopted a decision tree-based method to train supervised model. Kong et al. (Kong and Zhou, 2010) adopted a tree kernel-based method to model the syntax tree, so that the Chinese zero pronoun resolution model can make full use of the characteristics of the syntax tree. Chen et al. (Chen and Ng, 2016) designed a Chinese zero pronoun resolution model based on feed-forward neural network, and represented the zero-pronoun and candidate antecedent by combining manual feature vectors and word vectors, and obtained better performance than traditional machine learning-based methods. Yin et al. (Yin et al., 2017);(Yin et al., 2018a);(Yin et al., 2018b) designed a series of deep learning-based Chinese zero pronoun resolution model, which promoted the application of deep learning to Chinese zero pronoun resolution. Liu et al. (Liu et al., 2017) transformed the Chinese zero pronoun resolution task into the cloze-style reading comprehension task, and automatically constructed large-scale pseudo-data for the pre-training of their model.

2.2 Pre-training of Syntactic Dependency Parsing

Our method is similar to the method of pre-training of syntactic dependency parser, which has been successfully applied to some natural language processing tasks. Zhang et al. (Zhang et al., 2017) first

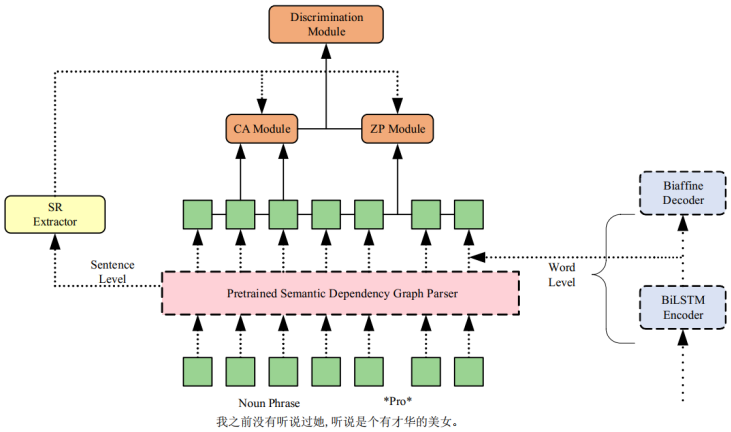


Figure 2: Chinese zero pronoun resolution model with pre-trained semantic dependency graph parser

proposed this method in the task of relation extraction. First, they trained the LSTM-based Biaffine syntactic dependency parser. Then, they extracted implicit syntactic representations from the LSTM layer of the well-trained syntactic dependency parser and integrated these representations into the relation extraction model. Guo et al. (Gao et al., 2017) and Yu et al. (Yu et al., 2018) used this method to integrate syntactic representations in the task of target-dependent sentiment analysis and discourse parsing respectively, and verified the effectiveness of this method in these tasks. Zhang et al. (Zhang et al., 2019) systematically studied the application of this method in the task of machine translation. Their experimental results show that this method obtains a more significant improvement than other methods such as Tree-Linearization and Tree-RNN in the task of machine translation. Jiang et al. (Jiang et al., 2020) applied this method to the task of Universal Conceptual Cognitive Annotation(UCCA) (Abend and Rappoport, 2013). Inspired by the method of integrating pre-trained information in ELMo (Peters et al., 2018), They made a weighted sum for the output of different LSTM layers of syntactic dependency parser. Their experimental results show that the method of fine-tuning pre-trained syntactic dependency parser improves the performance of UCCA model significantly.

3 Method

Given the success of the method of pre-training of syntactic dependency parser in some natural language processing tasks, we adopt a similar method to take the Chinese semantic graph dependency parsing as a pre-training task, and apply this method to Chinese zero pronoun resolution task.

Our proposed Chinese zero pronoun resolution model with pre-trained Chinese semantic dependency parser is composed of two parts, one is the pre-trained Chinese semantic dependency graph parser and the other is the Chinese zero pronoun resolution model. Specifically, The Chinese semantic dependency graph parser consists of two parts: BiLSTM-based encoder and Biaffine-based decoder. The Chinese zero pronoun resolution model consists of three parts: the zero pronoun module(ZP Module), the candidate antecedents module (CA Module) and the discrimination module. In addition, in order to obtain sentence-level semantic representations, we also used a CNN-based sentence representation extractor(SR Extractor).

For a sentence to be processed, the representations of each word will be feed into the pre-trained Chinese semantic dependency graph parser, so that each word can obtain the semantic-aware representations containing the information of semantic dependency graph. Then, the semantic-aware representations will be integrated into the Chinese zero pronoun resolution model to perform the subsequent processing. The overall architecture of our proposed model is shown in Figure 2:

3.1 Semantic Dependency Graph Parser

For the semantic dependency graph parser, we adopt 3-layer BiLSTM network and Biaffine network as encoder and decoder. The Biaffine-based parser has achieved the state of the art performance in some

tasks related to semantic dependency graph parsing. (Dozat and Manning, 2018);(Shen et al., 2019)

In the process of pre-training, we first use the concatenation of word vector, part of speech vector and character-level vector to represent a word. Then, we feed the word representations into the encoder to obtain the context-aware representations. Finally, we feed the context-aware representations of the word into the decoder to calculate the score of the dependency arc in the semantic dependency graph. The complete calculation process of the semantic dependency graph parser is shown in the following formulas:

$$w_t = [e_t^{(word)}; e_t^{(pos)}; e_t^{(char)}] \tag{1}$$

$$h_t = BiLSTM(w_t, h_{t-1}) \tag{2}$$

$$s_t^{(H,D)} = Biaffine(h_t^H, h_t^D) \tag{3}$$

where w_t means the word representations, h_t means the context-aware representations , $s_t^{(H,D)}$ means the score of the dependency arc, h_t^H and h_t^D mean context-aware representations of the head word and the dependent word respectively.

3.2 Zero Pronoun Module

According to the work of Yin et al (Yin et al., 2018b), we use BiLSTM network and self-attention mechanism to encode the preceding and following text of the zero pronoun. The purpose of using the self-attention mechanism is to obtain the attention weight distribution of the preceding and following word sequence. In this way, we can get the more powerful zero pronoun representations.

For a given anaphoric zero pronoun w_{zp}, we use $Context^{(pre)} = (w_1, w_2, \ldots, w_{zp-1})$ to denote the preceding word sequence of the zero pronoun, and use $Context^{(fol)} = (w_{zp+1}, w_{zp+2}, \ldots w_n)$ to denote the following word sequence of the zero pronoun. Each word w_t in the sentence is represented by the pre-trained word embedding.

In order to encode the contextual information of the word sequence, we first use two different 1-layer BiLSTM networks to separately process the the preceding word sequence and the following word sequence:

$$h_t^{(pre)} = BiLSTM^{(pre)}(w_t, h_{t-1}^{(pre)}) \tag{4}$$

$$h_t^{(fol)} = BiLSTM^{(fol)}(w_t, h_{t-1}^{(fol)}) \tag{5}$$

After that, we can obtain the preceding and following hidden vectors of the zero pronoun $h_t^{(pre)}$and $h_t^{(fol)}$ from the LSTM networks. we use $H^{(pre)}$ to denote the matrix which is concatenated by all preceding hidden vectors, and use $H^{(fol)}$ to denote the matrix which is concatenated by all following hidden vectors. where $H^{(pre)} \in \mathbb{R}^{n^{(pre)} \times d}$, $H^{(fol)} \in \mathbb{R}^{n^{(fol)} \times d}$, $n^{(pre)}$ and $n^{(fol)}$ means the number of words in the preceding and following word sequence respectively. d means the dimension of the hidden vectors.

The matrix $H^{(pre)}$ and $H^{(fol)}$ will be feed into the affine-based attention layers $Affine^{(pre)}$ and $Affine^{(fol)}$ to calculate the attention weight distribution of their associated sequences:

$$Affine(H) = Softmax(W_2 tanh(W_1 H^T)) \tag{6}$$

$$A^{(pre)} = Affine^{(pre)}(H^{(pre)}) \tag{7}$$

$$A^{(fol)} = Affine^{(fol)}(H^{(fol)}) \tag{8}$$

where $W_1 \in R^{h \times d}$, $W_2 \in R^{a \times h}$, $A^{(pre)} \in \mathbb{R}^{a \times n^{(pre)}}$, $A^{(fol)} \in \mathbb{R}^{a \times n^{(fol)}}$. It is worth explaining that a denotes the number of attention weight distributions. According to the work of Yin et al. (Yin et al., 2018b), we set the value of a to 2. Different attention weight distributions can capture different information, which further enhances the ability of the zero pronoun module.

Then, we can calculate the weighted sum of each row vector in the matrix by the following formula:

$$h_{zp}^{(pre)} = A^{(pre)} H^{(pre)} \tag{9}$$

$$h_{zp}^{(fol)} = A^{(fol)} H^{(fol)} \tag{10}$$

where $h_{zp}^{(pre)} \in R^{a \times d}$, $h_{zp}^{(fol)} \in R^{a \times d}$, If a is not equal to 1, We need to calculate the average of its row vectors.

At last, We take the concatenation of these two vectors as the final zero pronoun representations:

$$h_{zp} = [h_{zp}^{(pre)}; h_{zp}^{(fol)}] \tag{11}$$

3.3 Candidate Antecedents Module

When building the candidate antecedents module, we need to consider two types of the key information for the candidate antecedents. The first type of information is the context information of the candidate antecedents, and the second type of information is the interactive information between the zero pronoun and the candidate antecedents. Inspired by previous work (Lee et al., 2017), we use the context-aware boundary representations to capture context information and use attention mechanism to capture interactive information.

The candidate antecedent is usually a noun phrase composed of several words. So, we use $NP = (np_1, np_2, \ldots, np_n)$ to denote the set of all candidate antecedents for a given zero pronoun w_{zp} , and use $np_t = (w_i, w_2, \ldots, w_j)$ to denote a candidate antecedent within the set. First, we feed the pre-trained word vectors into the 1-layer BiLSTM network to obtain the context-aware representations of each word:

$$h_t = BiLSTM(w_t, h_{t-1}) \tag{12}$$

Apparently, we can get the sequence of the context-aware representations $np_t = (h_i, h_2, \ldots, h_j)$ from the outputs of the BiLSTM, where h_i means the start of the candidate antecedent, and h_j means the end of the candidate antecedent. we use h_i and h_j as the context-aware boundary representations of the candidate antecedents.

Then, we use a simple and effective scaled dot-product-based attention layer to calculate the weight distribution of the words in the candidate antecedent. We regard the zero pronoun representations h_{zp} as the query term, and regard the context-aware representations of all words in the candidate antecedent as the key term and value term. For simplicity in formula expression, we use the matrix H_{np} to denote he key term and value term:

$$h_{np}^{(attn)} = Softmax(\frac{h_{zp} H_{np}^T}{\sqrt{d_{np}}}) H_{np} \tag{13}$$

where $h_{zp} \in \mathbb{R}^{d_{zp}}$, $H_{np} \in \mathbb{R}^{n \times d_{np}}$, $d_{zp} = d_{np}$, n denotes the number of words in the candidate antecedent. d_{np} denotes the dimension of context-aware representations of all words in the candidate antecedent. d_{zp} denotes the dimension of zero pronoun representations. $h_{np}^{(attn)}$ is the weighted sum the context-aware representations of all words in the candidate antecedent, where $h_{np}^{(attn)} \in \mathbb{R}^{d_{np}}$.

Finally, we take the concatenate of h_i, h_j, and $h_{np}^{(attn)}$ as the the final representations of each candidate antecedent.

$$h_{np} = [h_i; h_j; h_{np}^{(attn)}] \tag{14}$$

3.4 Discrimination Module

After obtaining the representations of the zero pronoun and all candidate antecedents of this zero pronoun, we can feed these representations into the discrimination module to predict the real referent of the current zero pronoun.

For the discrimination module, this paper uses a bilinear function to calculate the probability distribution of all candidate antecedents of the current zero pronoun.

$$P(np_t|w_{zp}) = Softmax(h_{zp}UM_{np}^T + b) \tag{15}$$

$$\sum_{t=1}^{m} P(np_t|w_{zp}) = 1 \tag{16}$$

The parameters of the bilinear function are U and b, where $U \in \mathbb{R}^{k\times k}$, $b \in \mathbb{R}^k$, k denotes the dimension of the input vector of the bilinear function. h_{zp} denotes the zero pronoun representations. M_{np} denotes the matrix of all candidate antecedents of the current zero pronoun, where $h_{zp} \in \mathbb{R}^{1\times k}$, $M_{np} \in \mathbb{R}^{m\times k}$. m denotes the number of all candidate antecedents of the current zero pronoun.

Given the probability distribution of of all candidate antecedents of the current zero pronoun, we select the candidate antecedent with the highest probability as the real referent of the current zero pronoun.

3.5 The Integration of Semantic Representations

To make better use of the semantic representations from the pre-trained semantic dependency graph parser, We integrate the semantic representations of word-level and sentence-level into the Chinese zero pronoun resolution model.

Inspired by the work of Jiang et al. (Jiang et al., 2020), we first extract all output vectors from the BiLSTM-based encoder of the pre-trained semantic dependency graph parser and then use a set of trainable parameters to weighted sum these vectors to obtain the final semantic representations. we use h_t^{sem} to denote the semantic representations of a word. This process is formally denoted by the following formula:

$$h_t^l = BiLSTM^{(l)}(w_t, h_{t-1}) \tag{17}$$

$$h_t^{(sem)} = \sum_{l=1}^{L} \alpha_l h_t^l \tag{18}$$

where w_t is the original word representations，L is the layer number of the Bi-LSTM-based encoder, and α_l is the normalized weight of each layer.

For the integration of the word-level semantic representations, we simply concatenate the semantic representations of each word with its original word representations:

$$w_t^{(sem)} = [w_t; h_t^{(sem)}] \tag{19}$$

For the integration of the sentence-level semantic representations, We use the CNN-based sentence-level semantic representations extractor to perform 2-dimensional convolution and hierarchical pooling operations on the sentence sequence. Hierarchical pooling (Shen et al., 2018) is a combination of average pooling and max-pooling, which has better ability to capture word-order information. we use S_1^n to denote a sentence sequence with n words. This process is shown in the following formulas:

$$s^{(sem)} = Pooling(Convolution(S_1^n)) \tag{20}$$

After we obtain the sentence-level semantic representations, we integrate it into the zero pronoun module and the candidate antecedent module. We use two different multi-layer perceptrons to transform sentence-level semantic representations into zero pronoun-related and candidate antecedent-related representations. In this way, even if the zero pronoun and candidate antecedent are in the same sentence, these sentence-level semantic representations are different. This process is shown in the following formulas:

$$h_{zp}^{(sem)} = MLP^{(zp)}(s^{(sem)}) \tag{21}$$

$$h_{np}^{(sem)} = MLP^{(np)}(s^{(sem)}) \tag{22}$$

Finally, the zero pronoun representations and candidate antecedent representations that are integrated into the semantic representations can be formalized as:

$$h_{zp} = [h^{(pre)}; h^{(fol)}; h^{(sem)}] \tag{23}$$

$$h_{np} = [h_i; h_j; h_{np}^{(attn)}; h^{(sem)}] \tag{24}$$

3.6 Training Objective

The training objective is defined as:

$$Loss = -\Sigma_{zp} log P(np_t|w_{zp}) \tag{25}$$

where zp means the number of all anaphoric zero pronouns in the training set.

4 Experiment

4.1 Dataset and Resource

We conduct our experiments on the OntoNotes-5.0 dataset[0] which consists of document-level text selected from 6 domains : Broadcast News(BN), Newswire(NW), Broadcast Conversation(BC), Web Blog (WB), Telephone Conversation (TC) and Magazine(MZ). The training set has 1391 documents, a total of 36487 sentences and 12111 zero pronouns; The development set has 172 documents with a total of 6083 sentences and 1713 zero pronouns. The pre-trained word embedding used in Chinese zero pronoun resolution are trained by Word2Vec algorithm on Chinese Gigawords [1]. For Pre-training the Chinese semantic dependency graph parser, we use the SemEval-2016 Task 9 dataset[2]. For BERT related experiments, We use the Chinese Bert-base model, which has been pre-trained by the Google [3].

4.2 Evaluation Measures

We adopt the Recall, Precision and F-score (denoted as F) as the evaluation metrics of our Chinese zero pronoun resolution model. More specifically, recall, precision and F are defined as:

$$P = \frac{the\ number\ of\ zero\ pronouns\ predicted\ correctly}{the\ number\ of\ all\ predicted\ zero\ pronouns} \tag{26}$$

$$R = \frac{the\ number\ of\ zero\ pronouns\ predicted\ correctly}{the\ number\ of\ zero\ pronouns\ labeled\ in\ all\ datasets} \tag{27}$$

$$F = \frac{2PR}{P+R} \tag{28}$$

4.3 Hyperparameters

For Zero Pronoun Module, the hidden dimension of the LSTM is 128 and the output dimension of the affine-based attention layer is 128. For Candidate Antecedents Module, the hidden dimension of the LSTM is 128 and the output dimension of the scaled dot-product-based attention layer is 128. For all pre-trained representations, we convert the final input dimension to 256. For all LSTM, dropout rates are set to 0.33. For other neural network, dropout rates are set to 0.5. For training, the model is optimized by the Adam algorithm with the initial learning rate 0.003.

[0]http://catalog.ldc.upenn.edu/LDC2013T19
[1]https://catalog.ldc.upenn.edu/LDC2003T09
[2]https://github.com/HIT-SCIR/SemEval-2016
[3]https://github.com/google-research/bert

4.4 Main experiments

We chose three deep learning-based Chinese zero pronoun resolution model implemented by Yin et al as reference: Deep Memory Network-based Chinese zero pronoun resolution model (Yin et al., 2017)(DMN-ZP Model), Self-attention-based Chinese zero pronoun resolution model (Yin et al., 2018b)(SA-ZP Model) and Deep Reinforcement Learning-based Chinese zero pronoun resolution model (Yin et al., 2018a)(DRL-ZP Model).

We evaluate the performance of our Chinese zero pronoun resolution model on OntoNotes-5.0 development dataset with two different model settings: Chinese zero pronoun resolution model without pre-trained Chinese semantic dependency graph parser(Our Baseline Model), Chinese zero pronoun resolution model with pre-trained Chinese semantic dependency graph parser(Our Semantic-aware Model). The specific experimental results are shown in Table 1:

Model	NW(84)	MZ(162)	WB(284)	BN(390)	BC(510)	TC(283)	Overall
DMN-ZP Model	48.8	46.3	59.8	58.4	53.2	**54.8**	54.9
DRL-ZP Model	63.1	50.2	63.1	56.7	57.5	54	57.2
SA-ZP Model	**64.3**	52.5	62	**58.5**	57.6	53.2	57.3
Our Baseline Model	63.3	51.5	61.8	58.2	57.5	53.1	57.2
Our Semantic-aware Model	64.3	**52.7**	**63.3**	58.3	**58.8**	53.1	**57.6**

Table 1: Comparison of Different Chinese Zero pronoun Resolution Models

Compared with the baseline model, our semantic-aware model has achieved a 0.4 % improvement in F-score. Compared with previous deep learning-based models, the performance of our semantic-aware model is the best. According to the experimental results in various fields, we found that our semantic-aware model obtains the highest F-score in the MZ, BC and WB fields. Among them, the improvement of our semantic-aware model in the BC field is the most obvious. However, in the field of NW, BN and TC, the performance of our semantic-aware model has no advantage. One possible reason for this phenomenon is that the performance of the semantic dependency graph parser in these three fields is relatively poor, and it cannot provide valuable semantic information to the task of Chinese zero pronoun resolution.

4.5 Ablation Experiment

In order to further verify the effectiveness of our model, we tested the performance of models using the word-level and sentence-level integration method through ablation experiments. According to the experimental results in Table 2, we found that both integration methods can improve the performance of our model, and when both integration methods are used simultaneously, the performance of our model is optimal. The word-level integration method can only focus on the semantic information within the same sentence, while the sentence-level integration method has the ability to focus on the difference in sentence-level semantic information between different sentences. Therefore, the word-level integration method may be more suitable for the case where the zero pronoun and the candidate antecedent are in the same sentence, and the sentence-level integration method is more suitable for the case where the zero pronoun and the candidate antecedent are in different sentences. It is the complementarity of these two methods that makes the performance of our model continuously improved.

Model	Overall
Baseline Model	57.2
Sematic-aware Model(Sentence-Level)	57.3
Sematic-aware Model(Word-Level)	57.5
Sematic-aware Model	**57.6**

Table 2: Ablation experiment results

4.6 Integration with BERT

BERT (Devlin et al., 2018) is a pre-trained language model with strong capabilities and wide application. Many BERT-based natural language processing models have achieved the state of the art performance. In order to verify the effectiveness of our model after integrating the BERT representations, we compared and analyzed the following four sets of experiments: Baseline Model without BERT, Baseline model with BERT, Semantic-aware Model without BERT, Semantic-aware Model with BERT. It is worth noting that the method of integrating BERT information is the same as the method of integrating semantic dependency graph information. The specific experimental results are shown in Table 3:

Model	Overall
Baseline Model without BERT	57.2
Baseline Model with BERT	57.7
Semantic-aware Model without BERT	57.6
Semantic-aware Model with BERT	**57.9**

Table 3: Integration with BERT

According to the experimental results in the Table 3, we can see that the performance of the Semantic-aware Model with BERT is the best. This shows that BERT information and semantic dependency graph information have certain complementarity in the Chinese zero pronoun resolution task. But by comparing the performance of the Semantic-aware Model without BERT and Baseline model with BERT, We can see that the BERT information contributes more to the Chinese zero pronoun resolution task than the semantic dependency graph information. In addition, we can also see that BERT information improves the Baseline Model more than the Semantic-aware Model. This shows that the BERT model may encode part of the semantic information of the semantic dependency graph. Based on the above analysis, we hope that in the future research, we can further integrate the semantic dependency graph and even the information of semantic role labeling on the basis of the BERT model, so as to further enhance the ability of the BERT model in the Chinese zero pronoun resolution task.

4.7 Conclusion

This paper proposes a semantic-aware Chinese zero pronoun resolution model with pre-trained semantic Dependency Parser. In order to effectively integrate semantic information from the pre-trained semantic dependency graph parser, We integrate semantic representations into the Chinese zero pronoun resolution model at two levels: word level and sentence level. The experimental results show that our proposed model achieves better performance than other deep learning-based models. In addition, we find

that BERT information and semantic dependency graph information have certain complementarity in the Chinese zero pronoun resolution task. After our model is enhanced with the BERT representations, its performance has been further improved. In future research, we will explore the integration of BERT information and semantic dependency graph information to provide richer information for Chinese zero-finger resolution tasks.

Acknowledgements

This research project is supported by the National Natural Science Foundation of China (61872402), the Humanities and Social Science Project of the Ministry of Education (17YJAZH068)，Science Foundation of Beijing Language and Culture University (supported by the Fundamental Research Funds for the Central Universities) (18ZDJ03)，the Open Project Program of the National Laboratory of Pattern Recognition (NLPR).

References

Omri Abend and Ari Rappoport. 2013. Universal conceptual cognitive annotation (UCCA). page 11.

Wanxiang Che, Yanqiu Shao, Ting Liu, and Yu Ding. 2016. SemEval-2016 task 9: Chinese semantic dependency parsing. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1074–1080. Association for Computational Linguistics.

Chen Chen and Vincent Ng. 2016. Chinese zero pronoun resolution with deep neural networks. 1:778–788.

Jacob Devlin, Mingwei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv: Computation and Language*.

Timothy Dozat and Christopher D. Manning. 2018. Simpler but more accurate semantic dependency parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 484–490, Melbourne, Australia, July. Association for Computational Linguistics.

Yuze Gao, Yue Zhang, and Tong Xiao. 2017. Implicit syntactic features for targeted sentiment analysis. page 9.

Wei Jiang, Zhenghua Li, and Min Zhang. 2020. Syntax-enhanced ucca semantic parsing. *Beijing Da Xue Xue Bao*, 56(1):89–96.

Fang Kong and Guodong Zhou. 2010. A tree kernel-based unified framework for chinese zero anaphora resolution. pages 882–891.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution.

Ting Liu, Yiming Cui, Qingyu Yin, Weinan Zhang, Shijin Wang, and Guoping Hu. 2017. Generating and exploiting large-scale pseudo training data for zero pronoun resolution. 1:102–111.

Converse S P and Palmer M S. 2006. *Pronominal anaphora resolution in Chinese*. University of Pennsylvania.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations.

Dinghan Shen, Guoyin Wang, Wenlin Wang, Martin Renqiang Min, Qinliang Su, Yizhe Zhang, Chunyuan Li, Ricardo Henao, and Lawrence Carin. 2018. Baseline needs more love: On simple word-embedding-based models and associated pooling mechanisms.

Zizhuo Shen, Huayong Li, Dianqing Liu, and Yanqiu Shao. 2019. Dependency-gated cascade biaffine network for chinese semantic dependency graph parsing. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 840–851. Springer.

Qingyu Yin, Yu Zhang, Weinan Zhang, and Ting Liu. 2017. Chinese zero pronoun resolution with deep memory network. pages 1309–1318.

Qingyu Yin, Yu Zhang, Weinan Zhang, Ting Liu, and William Yang Wang. 2018a. Deep reinforcement learning for chinese zero pronoun resolution. 1:569–578.

Qingyu Yin, Yu Zhang, Weinan Zhang, Ting Liu, and William Yang Wang. 2018b. Zero pronoun resolution with attention-based neural network. pages 13–23.

Nan Yu, Meishan Zhang, and Guohong Fu. 2018. Transition-based neural rst parsing with implicit syntax features. pages 559–570.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2017. End-to-End Neural Relation Extraction with Global Optimization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1730–1740, Copenhagen, Denmark. Association for Computational Linguistics.

Meishan Zhang, Zhenghua Li, Guohong Fu, and Min Zhang. 2019. Syntax-enhanced neural machine translation with syntax-aware word representations. pages 1151–1161.

Shanheng Zhao and Hwee Tou Ng. 2007. Identification and resolution of chinese zero pronouns: A machine learning approach. pages 541–550.

Improving Sentence Classification by Multilingual Data Augmentation and Consensus Learning

Yanfei Wang[†], **Yangdong Chen**[†], **Yuejie Zhang**[*]
School of Computer Science, Shanghai Key Laboratory of Intelligent Information Processing,
Fudan University, Shanghai 200433, China
{17210240046, 19110240010, yjzhang}@fudan.edu.cn

Abstract

Neural network based models have achieved impressive results on the sentence classification task. However, most of previous work focuses on designing more sophisticated network or effective learning paradigms on monolingual data, which often suffers from insufficient discriminative knowledge for classification. In this paper, we investigate to improve sentence classification by multilingual data augmentation and consensus learning. Comparing to previous methods, our model can make use of multilingual data generated by machine translation and mine their language-share and language-specific knowledge for better representation and classification. We evaluate our model using English (i.e., source language) and Chinese (i.e., target language) data on several sentence classification tasks. Very positive classification performance can be achieved by our proposed model.

1 Introduction

Sentence classification is a task of assigning sentences to predefined categories, which has been widely explored in past decades. It requires modeling, representing and mining a degree of semantic comprehension, which are mainly based on the structure or sentiment of sentences. This task is important for many practical applications, such as product recommendation (Dong et al., 2013), public opinion detection (Pang et al., 2008), and human-machine interaction (Clavel and Callejas, 2015), etc.

Recently, deep learning has achieved state-of-the-art results across a range of Computer Vision (CV) (Krizhevsky et al., 2012), Speech Recognition (Graves et al., 2013), and Natural Language Processing tasks (NLP) (Kalchbrenner et al., 2014a). Especially, Convolutional Neural Network (CNN) has gained great success in sentence modelling. However, training deep models requires a great diversity of data so that more discriminative patterns can be mined for better prediction. Most existing work on sentence classification focuses on learning better representation for a sentence given limited training data (i.e., *source language*), which resorts to design a sophisticated network architecture or learning paradigm, such as attention model (Yang et al., 2016), multi-task learning (Liu et al., 2016), adversarial training (Liu et al., 2017), etc. Inspired by recent advances in Machine Translation (MT) (Wu et al., 2016), we can perform an input data augmentation by making use of multilingual data (i.e., *target language*) generated by machine translation for sentence classification tasks. Such generated new language data can be used as the auxiliary information, and provide the additional knowledge for learning a robust sentence representation. In order to effectively exploit multilingual data, we further propose a novel deep consensus learning framework to mine the language-share and language-specific knowledge for sentence classification. Since the machine translation model can be pre-trained off-the-shelf with great generalization ability, it is worth noting that we do not directly introduce other language data comparing to existing methods in the training and testing phase.

Our main contributions are of two-folds: 1) We first propose utilizing multilingual data augmentation to assist sentence classification, which can provide more beneficial auxiliary knowledge for sentence

[†]: Equal contribution
[*]: Corresponding author

Proceedings of the 19th China National Conference on Computational Linguistics, pages 842-852,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

modeling; 2) A novel deep consensus learning framework is constructed to fuse multilingual data and learn their language-share and language-specific knowledge for sentence classification. In this work, we use English as our source language and Chinese/Dutch as the target language from an English-Chinese/Dutch translator. The related experimental results s how that our model can achieve very promising performance on several sentence classification tasks.

2 Related Work

2.1 Sentence Classification

Sentence classification is a well-studied research area in NLP. Various approaches have been proposed in last a few decades (Tong and Koller, 2001; Fernández-Delgado et al., 2014). Among them, Deep Neural Network (DNN) based models have shown very good results for several tasks in NLP, and such methods become increasing popular for sentence classification. Various neural networks are proposed to learn better sentence representation for classification. An influential one is the work of Kim (2014), where a simple Convolutional Neural Network (CNN) with a single layer of convolution was used for feature extraction. Following this work, Zhang and LeCun (2015) used CNNs for text classification with character-level features provided by a fully connected DNN. Liu et al. (2016) used a multi-tasking learning framework to learn multiple related tasks together for sentence classification task. Based on Recurrent Neural Network (RNN), they utilized three different mechanisms of sharing information to model text. In practice, they used Long Short-Term Memory Network (LSTM) to address the issue of learning long-term dependencies. Lai et al. (2015) proposed a Recurrent Convolutional Neural Network (RCNN) model for text classification, which applied a recurrent structure to capture contextual information and employed a max-pooling layer to capture the key components in texts. Jiang et al. (2018) proposed a text classification model based on deep belief network and softmax regression. In their model, a deep belief network was introduced to solve the sparse high-dimensional matrix computation problem of text data. They then used softmax regression to classify the text. Yang et al. (2016) used Hierarchical Attention Network (HAN) for document classification in their model, where a hierarchical structure was introduced to mirror the hierarchical structure of documents, and two levels of attention mechanisms were applied both at the word and sentence level.

Another direction of solutions for sentence classification is to use more effective learning paradigms. Yogatama et al. (2017) combined Generative Adversarial Networks (GAN) with RNN for text classification. Billal et al. (2017) solved the problem of multi-label text classification in semi-supervised learning manner. Liu et al. (2017) proposed a multi-task adversarial representation learning method for text classification. Zhang et al. (2018a) attempted to learn structured representation of text via deep reinforcement learning. They tried to learn sentence representation by discovering optimized structures automatically and demonstrated two attempts of Information Distilled LSTM (ID-LSTM) and Hierarchically Structured LSTM (HS-LSTM) to build structured representation.

However, these tasks do not take into account the auxiliary language information corresponding to the source language. This auxiliary language can provide the additional knowledge to learn more accurate sentence representation.

2.2 Deep Consensus Learning

Existing sentence classification works (Kim, 2014; Zhang and LeCun, 2015; Lai et al., 2015; Jiang et al., 2018; Yogatama et al., 2017; Billal et al., 2017; Zhang et al., 2018a) mainly focus on feature representation or learning a structured representation (Zhang et al., 2018a). Deep learning based sentence classification models have obtained impressive performance. Those approaches are largely due to the powerful automatic learning and representation capacities of deep models, which benefit from big labelled training data and the establishment of large-scale sentence/document datasets (Yogatama et al., 2017; Billal et al., 2017; Zhang et al., 2018a). However, all of the existing methods usually consider only one type of language information by a standard single language process. Such methods not only ignore the potentially useful information of other different languages, but also lose the opportunity of mining the correlated complementary advantages across different languages. A similar model is [20], which used

synthetic source sentences to improve the performance of Neural Machine Translation (NMT). While sharing the high-level multilingual feature learning spirit, the proposed consensus learning model significantly has the following three outstanding characteristics. (1) Beyond the language concatenation based on fusion, our model uniquely considers a synergistic cross-language interaction learning and regularization by consensus propagation. This aims to overcome the challenge of learning discrepancy in multilingual feature optimization. (2) Instead of the traditional single loss design, a multi-loss concurrent supervision mechanism is deployed by our model. This enforces and improves the model's individuality learning power of language-specific feature. (3) Through NMT, we can eliminate some of the ambiguous words and highlight some key words.

3 Methodology

We aim to learn a deep feature representation model for sentence classification based on language-specific input, without any specific feature transformation. Figure 1 depicts our proposed framework, which consists of two stages. The first stage performs multilingual data augmentation from an off-the-shelf machine translator; and the second one feeds the source language data and generated target language data to our deep consensus learning model for sentence classification.

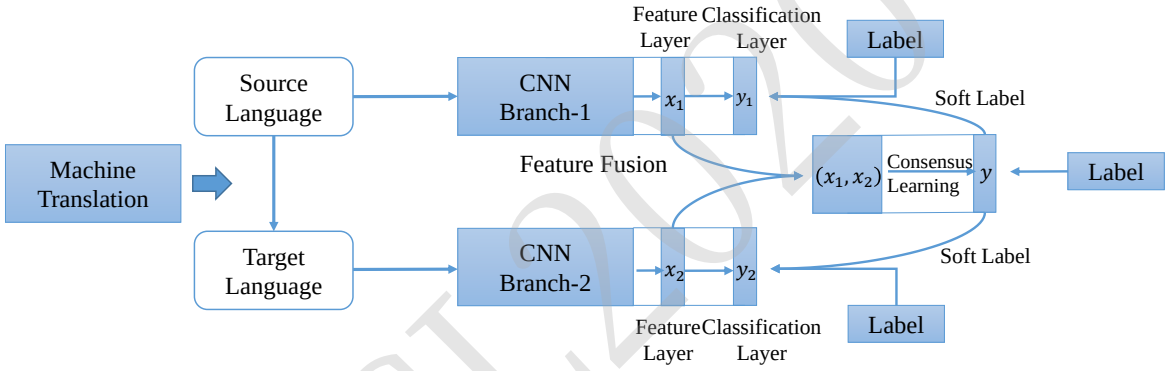


Figure 1. The framework of our proposed model for sentence classification.

3.1 Multilingual Data Augmentation

Data augmentation is a very important technique in machine learning that allows building better models. It has been successfully used for many tasks in areas of CV and NLP, such as image recognition (Krizhevsky et al., 2012) and MT (Zhang et al., 2018a). In MT, Back-translation is a common data argumentation method (Sennrich et al., 2016; Zhang et al., 2018b), which allows us to combine monolingual training data. Especially when the existing data is insufficient to learn a discriminative representation for a specific task, the data augmentation methods can be used.

In sentence classification, given an input sentence in one language, we perform data augmentation by translating the sentence to another language using existing machine translation methods. We name the input language as *source language* and the translated language as *target language*. This motivation comes from the recent great advance in NMT (Wu et al., 2016). Given an input sentence in source language, we simply call the *Google* Translation API [0] to get the translated data in target language. Comparing to other state-of-art NMT models, the *Google* translator has the advantage of both effectiveness and efficiency in real application scenarios. Since target language is used for multilingual data augmentation and the type of it is not important to the proposed model, we random choose Chinese and Dutch respectively as the target language for multilingual data augmentation, and the source language depends on the language of input sentence.

[0] *https://cloud.google.com/translate/*

3.2 Deep Consensus Learning Model

Learning a consensus classification model with the combination of several beneficial information into one final prediction can lead to a more accurate result (Chen et al., 2017). Thus we use two languages of data, $\{S_1, S_2, S_3, \cdots, S_{N-1}, S_N\}$ and $\{T_1, T_2, T_3, \cdots, T_{N-1}, T_N\}$, to perform consensus learning for sentence classification. As shown in Figure 1, our model has three parts: (1) Two branches of language-specific subnetworks for learning the most discriminative features for each language data; (2) One fusion branch responsible for learning the language-share representation with the optimal integration of two kinds of language-specific knowledge; and (3) Consensus propagation for the feature regularization and learning optimization. The design of architecture components will be described in detail as below.

Language-specific Network We utilize the *TextCNN* architecture (Kim, 2014) for each branch of language-specific network, which has been proved to be very effective for sentence classification. *TextCNN* can be divided into two stages, that is, one with convolution layers for feature learning, and another with full connected layers for classification. Given training labels of input sentence, the Softmax classification loss function is used to optimize the category discrimination. Formally, given a corpus of sentences of source language $\{S_1, S_2, S_3, \cdots, S_{N-1}, S_N\}$, the training loss on a batch of n sentences can be computed as:

$$L_{S\_brch} = -\frac{1}{n}\sum_{i=1}^{n}\log\left(\frac{\exp\left(w_{y_i}^T S_i\right)}{\sum_{k=1}^{c}\exp\left(w_k^T S_i\right)}\right) \tag{1}$$

where c is the number of categories of sentences; y_i denotes the category label of the sentence S_i; and w is the prediction function parameter of the training category class k. The training loss for target language branch $L_(T\_brch)$ can be computed in the same manner. Meanwhile, since the source language and target language belong to different language spaces, such two branches of language-specific networks are trained with the uniform architecture but different parameters.

Language-share Network We perform the language-share feature learning from two language-specific branches. For this purpose, we firstly perform the language-share learning by fusing across from these two branches. For design simplicity and cost efficiency, we achieve the feature fusion on the feature vectors from the concatenation layer before dropout in *TextCNN* by an operation of *Concat→FC→Dropout→FC→Softmax*. This produces a category prediction score for input pair (a sentence in source language and its translated one in target language). We similarly utilize the Softmax classification loss L_ST for the language-share classification learning as that in the language-specific branches.

Consensus Propagation Inspired by the teacher-student learning approach, we propose to regularize the language-specific learning by consensus feedback from the language-share network. More specifically, we utilize the consensus probability $P_{ST} = \left[p_{ST}^1, p_{ST}^2, \cdots, p_{ST}^{c-1}, p_{ST}^c\right]$ from the language-share network as the *teacher* signal (called "*soft label*" versus the ground-truth one-hot "*hard label*") to guide the learning process of all language-specific branches (*student*) concurrently by an additional regularization, which can be formulated in a cross-entropy manner as:

$$\mathcal{H}_S = -\frac{1}{c}\sum_{i=1}^{c}\left(p_{ST}^i \ln\left(p_s^i\right) + \left(1 - p_{ST}^i\right)\ln\left(1 - p_s^i\right)\right) \tag{2}$$

where $P_S = [p_S^1, p_S^2, p_S^3, \cdots, p_S^{c-1}, p_S^c]$ defines the probability prediction over all c sentence classes by the source language branch. Thus the final loss function for the language-specific network can be redefined via enforcing an additional regularization in Eq. (1).

$$L_S = L_{S\_brch} + \lambda\mathcal{H}_S \tag{3}$$

where λ controls the importance tradeoff between two terms. The regularization terms $\mathcal{H}_T$ and L_T for target language branch can be computed in the same way.

The training of our proposed model proceeds in two stages. First, we rely on training the language-specific network separately, which is terminated by the early stopping strategy. Afterwards, the language-share network and consensus propagation loss are introduced. We use the whole loss defined in Eq.

(3) and L_{ST} to train the language-specific network and language-share network at the same time. In the testing time, given an input sentence and its translated sentence, the final prediction is obtained by averaging the three prediction scores from the language-specific networks and the language-share network.

4 Experiment and Analysis

In this section, we investigate the empirical performance of our proposed architecture on five benchmark datasets for sentence classification.

4.1 Datasets and Experimental Setup

The sentence classification datasets include:

(1) ***MR***: This dataset includes movie reviews with one sentence per review, in which the classification involves detecting positive/negative reviews (Pang and Lee, 2005).

(2) ***CR***: This dataset contains annotated customer reviews of 5 products, and the target is to predict positive/negative reviews (Hu and Liu, 2004).

(3) ***Subj***: This dataset is a subjectivity dataset, which includes subjective or objective sentiments (Pang and Lee, 2004).

(4) ***TREC***: This dataset focuses on the question classification task that involves 6 question types (Li and Roth, 2002).

(5) ***SST-1***: This dataset is Stanford Sentiment Treebank, an extension of *MR*, which contains training/development/testing splits and fine-grained labels (very positive, positive, neutral, negative, very negative) (Socher et al., 2013).

Similar with (Kim, 2014), the initialized word vectors for source language are obtained from the publicly available *word2vec* vectors that were trained on 100 billion words from *Google News*. For target language of Chinese, we retrain the *word2vec* models on *Chinese Wikipedia Corpus*; and for target language of Dutch, we retrain the *word2vec* models on *Dutch Wikipedia Corpus*. In our experiments, we choose the *CNN-multichannel* model variant of *TextCNN* because of its better performance.

4.2 Ablation Study

We first compare our proposed model with several baseline models for sentence classification. Here, we use ***S+T*** to indicate that the model's input contains the source language and the target language. ***T***(*) indicates the type of target language, i.e., ***T(CH)*** indicates that the target language is Chinese, and ***T(DU)*** indicates that the target language is Dutch. Figure 2 and 3 show the comparison results of classification accuracy rate on five benchmark datasets. *CNN(S)* denotes the *CNN-multichannel* model variant of *TextCNN*, which only uses the source language data of English for training and testing. *CNN(T)* is a retrained *TextCNN* model on the translated target language data of Chinese(CH)/Dutch(DU), and the other settings keep the same as *CNN(S)*. *Ours(S+T(*))* denotes our model by combining multilingual data augmentation with deep consensus learning. We can find that *Ours(S+T(*))* performs much better than those baselines, which proves the effectiveness of our framework. It is obvious that multilingual data augmentation can provide the beneficial additional discrimination for learning a robust sentence representation for classification. It is worth noting that *CNN(T)* is even better than *CNN(S)* on *MR*. This indicates that existing machine translation methods can not only keep the discriminative semantics of source language, but also create useful discrimination in target language space.

Similar to *TextCNN*, we also use several variants of the model to demonstrate the effectiveness of our model. As we know, when lacking a large supervised training set, we usually use word vectors obtained from unsupervised neural language models to initialize word vectors for performance improvement. Thus we use various word vector initialization methods to validate the model.

The different word vector initialization methods include:

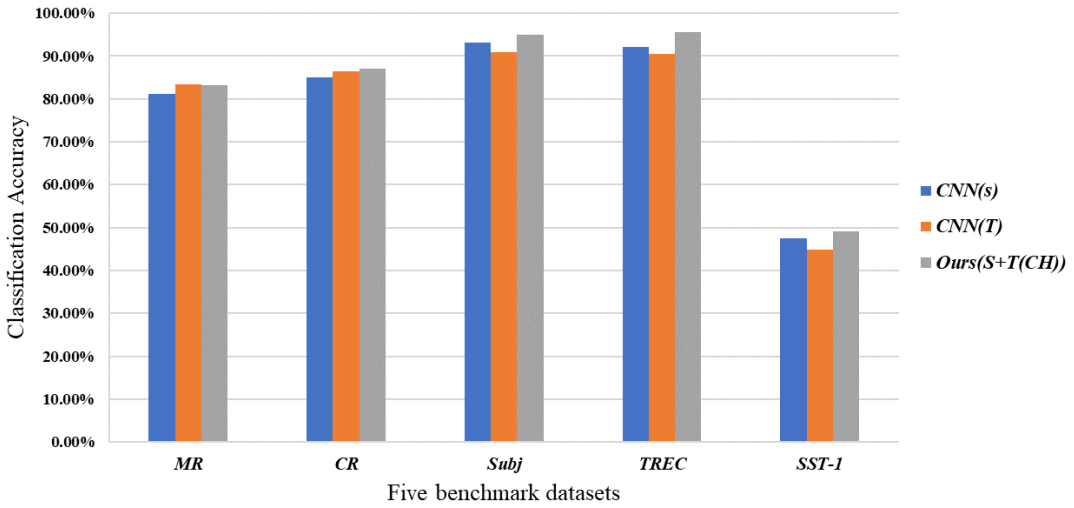


Figure 2. The comparison results with existing baseline models based on English→ Chinese MT.

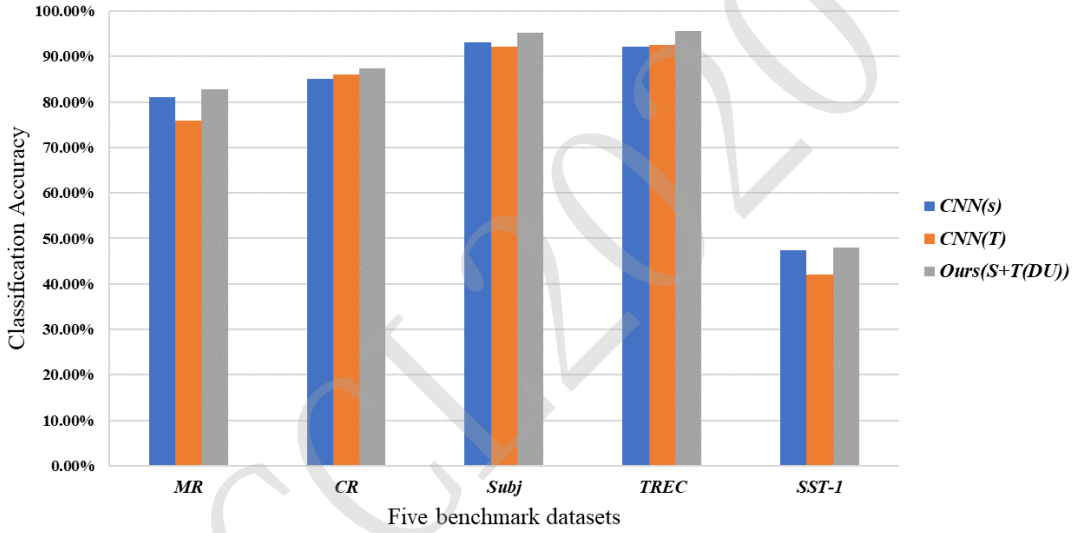


Figure 3. The comparison results with existing baseline models based on English→ Dutch MT.

(1) **Rand**: All words are randomly initialized and can be trained during training.

(2) **Static**: All words of input language are initialized by pre-trained vectors from the corresponding language *word2vec*. Simultaneously, all these words are kept static during training.

(3) **Non-static**: This is an initialization method same to Static, but the pre-trained vectors can be fine-tuned during training.

(4) **Multichannel**: This model contains two types of word vector, which are treated as different channels. One type of word vector can be finetuned during training, while the other keeps static. Two types of word vector are initialized with the same word embedding form *word2vec*.

In Table 1, we show the experimental results of different model variants based on English→ Chinese MT. Compared to the source language *S*, the accuracy rates of the target language *T(CH)* classification are partly improved or decreased, which shows the strong dataset dependency. Considering that the proposed *S+T(CH)* model with Multichannel obtains the current optimal results, we choose the model with Multichannel as our final results. Similar to Table 1, we show the experimental results of different

model variants based on English→ Dutch MT in Table 2. Combining the experimental results in Tables 1 and 2, we have enough reasons to prove the validity of our consensus learning method.

Evaluation Pattern	Model Variant	Benchmark Dataset				
		MR	*CR*	*Subj*	*TREC*	*SST*-1
S	**Rand**	76.1%	79.8%	89.6%	91.2%	45.0%
	Static	81.0%	84.7%	93.0%	92.8%	45.5%
	Non-static	81.5%	84.3%	93.4%	93.6%	48.0%
	Multichannel	81.1%	85.0%	93.2%	92.2%	47.4%
T(CH)	**Rand**	79.5%	79.8%	88.5%	85.4%	42.5%
	Static	83.0%	81.4%	89.8%	89.4%	43.6%
	Non-static	82.5%	86.4%	90.1%	90.4%	42.9%
	Multichannel	83.4%	86.4%	90.9%	90.4%	44.8%
S+T(CH)	**Rand**	79.7%	77.2%	92.0%	92.4%	47.1%
	Static	81.8%	86.4%	93.6%	95.0%	47.6%
	Non-static	81.7%	**87.9%**	94.5%	**95.2%**	48.0%
	Multichannel	**83.2%**	87.1%	**95.0%**	**95.6%**	**49.1%**

Table 1. The experimental results of different model variants based on English→ Chinese MT.

Evaluation Pattern	Model Variant	Benchmark Dataset				
		MR	*CR*	*Subj*	*TREC*	*SST*-1
T(DU)	**Rand**	66.5%	78.5%	85.3%	84.8%	35.3%
	Static	75.0%	82.1%	91.6%	89.0%	40.8%
	Non-static	76.6%	86.6%	92.8%	93.0%	42.9%
	Multichannel	76.0%	86.1%	92.1%	92.6%	42.0%
S+T(DU)	**Rand**	76.1%	87.1%	89.5%	90.8%	42.6%
	Static	81.6%	85.6%	93.4%	94.8%	46.2%
	Non-static	81.8%	84.0%	93.9%	**95.6%**	46.8%
	Multichannel	**82.8%**	**87.3%**	**95.3%**	**95.6%**	**47.9%**

Table 2. The experimental results of different model variants based on English→ Dutch MT.

4.3 Comparison with Existing Approaches

To further exhibit the effectiveness of our model, we compare our approach with several state-of-the-art approaches, including recent LSTM-based models and CNN-based models. As shown in Table 3, it can be concluded that our approach can gain very promising results comparing to these methods. The whole performance is measured by the accuracy rate for sentence classification. We roughly divide the existing approaches into four categories. The first category is the RNN-based model, in which Standard-RNN refers to Standard Recursive Neural Network (Socher et al., 2013), MV-RNN is Matrix-Vector Recursive Neural Network (Socher et al., 2012), RNTN denotes Recursive Neural Tensor Network (Socher et al., 2013), and DRNN represents Deep Recursive Neural Network (Irsoy and Cardie, 2014). The second category is the LSTM-based model, in which bi-LSTM stands for Bidirectional LSTM (Tai et al., 2015), SA-LSTM means Sequence Autoencoder LSTM (Dai and Le, 2015), Tree-LSTM is Tree-Structured LSTM (Tai et al., 2015), and Standard-LSTM represents Standard LSTM Network (Tai et al., 2015). The CNN-based model is the third category, in which DCNN denotes Dynamic Convolutional Neural Network (Kalchbrenner et al., 2014b), CNN-Multichannel is Convolutional Neural Network with Multi-channel (Kim, 2014), MVCNN refers to Multichannel Variable-Size Convolution Neural Network (Yin

and Schütze, 2015), Dep-CNN denotes Dependency-based Convolutional Neural Network (Ma et al., 2015), MGNC-CNN stands for Multi-Group Norm Constraint CNN (Zhang et al., 2016b), and DSCNN represents Dependency Sensitive Convolutional Neural Network (Zhang et al., 2016a). The fourth one is based on other methods, in which Combine-skip refers to skip-thought model with the concatenation of the vectors from uni-skip and bi-skip (Kiros et al., 2015), CFSF indicates initializing Convolutional Filters with Semantic Features (Li et al., 2017), and GWS denotes exploiting domain knowledge via Grouped Weight Sharing (Zhang et al., 2017). Especially on *MR*, our model of ***S+T(CH)*** can achieve the best performance by a margin of nearly 5%. This improvement demonstrates that our multilingual data augmentation and consensus learning can make great contributions to such sentence classification task. Through multilingual data augmentation, important words will be retained. The NMT systems can map those ambiguous words in source language to different word units in target language, which can achieve the result of word disambiguation. Essentially, our method can enable CNNs to obtain better discrimination and generalization abilities. To further demonstrate the superiority of our proposed model, we also use English as the source language and Dutch as the target language to evaluate the model of ***S+T(DU)***. On the four benchmark datasets of *MR*, *CR*, *Subj*, and *TREC*, our models of ***S+T(CH)*** and ***S+T(DU)*** have both achieved the best results at present.

Model	Approach	Benchmark Dataset				
		MR	*CR*	*Subj*	*TREC*	*SST-1*
RNN-based Model	**Standard-RNN** (Socher et al., 2013)	-	-	-	-	43.2%
	MV-RNN (Socher et al., 2012)	-	-	-	-	44.4%
	RNTN (Socher et al., 2013)	-	-	-	-	45.7%
	DRNN (Irsoy and Cardie, 2014)	-	-	-	-	49.8%
LSTM-based Model	**bi-LSTM** (Tai et al., 2015)	-	-	-	-	49.1%
	SA-LSTM (Dai and Le, 2015)	80.7%	-	-	-	-
	Tree-LSTM (Tai et al., 2015)	-	-	-	-	**51.0%**
	Standard-LSTM (Tai et al., 2015)	-	-	-	-	45.8%
CNN-based Model	**DCNN** (Kalchbrenner et al., 2014b)	-	-	-	93.0%	48.5%
	CNN-Multichannel (Kim, 2014)	81.1%	85.0%	93.2%	85.0%	47.4%
	MVCNN (Yin and Schütze, 2015)	-	-	93.9%	-	49.6%
	Dep-CNN (Ma et al., 2015)	-	-	-	95.4%	49.5%
	MGNC-CNN (Zhang et al., 2016b)	-	-	94.1%	95.5%	-
	DSCNN (Zhang et al., 2016a)	82.2%	-	93.9%	**95.6%**	50.6%
Model based on Other Methods	**Combine-skip** (Kiros et al., 2015)	76.5%	80.1%	93.6%	92.2%	-
	CFSF (Li et al., 2017)	82.1%	86.0%	93.7%	93.7%	-
	GWS (Zhang et al., 2017)	81.9%	84.8%	-	-	-
Our Model	***Ours (S+T(CH))***	**87.6%**	**87.1%**	**95.0%**	**95.6%**	49.1%
	Ours (S+T(DU))	**82.8%**	**87.3%**	**95.3%**	**95.6%**	47.9%

Table 3. The comparison results between the state-of-the-art approaches and ours.

5 Conclusion and Future Work

In this paper, multilingual data augmentation is introduced to further improve sentence classification. A novel deep consensus learning model is established to fuse multilingual data and learn the language-share and language-specific knowledge. The related experimental results demonstrate the effectiveness of our proposed framework. In addition, our method requires no external data comparing to existing methods, which makes it very practical with good generalization abilities in real application scenarios. In the future, we will try to explore the performance of the model on larger sentence/document datasets. The linguistic features of different languages will be also considered when selecting the target language.

Acknowledgements

This work was supported by National Natural Science Foundation of China (No. 61976057, No. 61572140). Yanfei Wang and Yangdong Chen contributed equally to this work, and were co-first authors. Yuejie Zhang was the corresponding author.

References

Belainine Billal, Alexsandro Fonseca, Fatiha Sadat, and Hakim Lounis. 2017. Semi-supervised learning and social media text analysis towards multi-labeling categorization. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 1907–1916. IEEE.

Yanbei Chen, Xiatian Zhu, and Shaogang Gong. 2017. Person re-identification by deep learning multi-scale representations. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2590–2600.

Chloe Clavel and Zoraida Callejas. 2015. Sentiment analysis: from opinion mining to human-agent interaction. *IEEE Transactions on affective computing*, 7(1):74–93.

Andrew M Dai and Quoc V Le. 2015. Semi-supervised sequence learning. In *Advances in neural information processing systems*, pages 3079–3087.

Ruihai Dong, Michael P O'Mahony, Markus Schaal, Kevin McCarthy, and Barry Smyth. 2013. Sentimental product recommendation. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 411–414.

Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, and Dinani Amorim. 2014. Do we need hundreds of classifiers to solve real world classification problems? *The journal of machine learning research*, 15(1):3133–3181.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

Ozan Irsoy and Claire Cardie. 2014. Deep recursive neural networks for compositionality in language. In *Advances in neural information processing systems*, pages 2096–2104.

Mingyang Jiang, Yanchun Liang, Xiaoyue Feng, Xiaojing Fan, Zhili Pei, Yu Xue, and Renchu Guan. 2018. Text classification based on deep belief network and softmax regression. *Neural Computing and Applications*, 29(1):61–70.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014a. A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 655–665.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014b. A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 655–665.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Twenty-ninth AAAI conference on artificial intelligence*.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.

Shen Li, Zhe Zhao, Tao Liu, Renfen Hu, and Xiaoyong Du. 2017. Initializing convolutional filters with semantic features for text classification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1884–1889.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Recurrent neural network for text classification with multi-task learning. pages 2873–2879.

Pengfei Liu, Xipeng Qiu, and Xuan-Jing Huang. 2017. Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–10.

Mingbo Ma, Liang Huang, Bowen Zhou, and Bing Xiang. 2015. Dependency-based convolutional neural networks for sentence embedding. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 174–179.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*, page 271. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 115–124. Association for Computational Linguistics.

Bo Pang, Lillian Lee, et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96.

Richard Socher, Brody Huval, Christopher D Manning, and Andrew Y Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 1201–1211. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566.

Simon Tong and Daphne Koller. 2001. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

Wenpeng Yin and Hinrich Schütze. 2015. Multichannel variable-size convolution for sentence classification. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 204–214.

Dani Yogatama, Chris Dyer, Wang Ling, and Phil Blunsom. 2017. Generative and discriminative text classification with recurrent neural networks. *arXiv preprint arXiv:1703.01898*.

Xiang Zhang and Yann LeCun. 2015. Text understanding from scratch. *arXiv preprint arXiv:1502.01710*.

Rui Zhang, Honglak Lee, and Dragomir Radev. 2016a. Dependency sensitive convolutional neural networks for modeling sentences and documents. In *Proceedings of NAACL-HLT*, pages 1512–1521.

Ye Zhang, Stephen Roller, and Byron C Wallace. 2016b. Mgnc-cnn: A simple approach to exploiting multiple word embeddings for sentence classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1522–1527.

Ye Zhang, Matthew Lease, and Byron C Wallace. 2017. Exploiting domain knowledge via grouped weight sharing with application to text categorization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 155–160.

Tianyang Zhang, Minlie Huang, and Li Zhao. 2018a. Learning structured representation for text classification via reinforcement learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Zhirui Zhang, Shujie Liu, Mu Li, Ming Zhou, and Enhong Chen. 2018b. Joint training for neural machine translation models with monolingual data. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Attention-Based Graph Neural Network with Global Context Awareness for Document Understanding

Yuan Hua[1], Zheng Huang[1,2], Jie Guo[1], Weidong Qiu[1]
[1]Shanghai Jiao Tong University, Shanghai, China
[2]Westone Cryptologic Research Center, Beijing, China
{isyuan.hua, huang-zheng, guojie, qiuwd}@sjtu.edu.cn

Abstract

Information extraction from documents such as receipts or invoices is a fundamental and crucial step for office automation. Many approaches focus on extracting entities and relationships from plain texts, however, when it comes to document images, such demand becomes quite challenging since visual and layout information are also of great significance to help tackle this problem. In this work, we propose the attention-based graph neural network to combine textual and visual information from document images. Moreover, the global node is introduced in our graph construction algorithm which is used as a virtual hub to collect the information from all the nodes and edges to help improve the performance. Extensive experiments on real-world datasets show that our method outperforms baseline methods by significant margins.

1 Introduction

Information Extraction (Akbik et al., 2019; Lample et al., 2016; Zheng et al., 2017) is a widely studied task of retrieving structured information from texts and many inspiring achievements have been made in this field. However, most of these works are generally focusing on extracting entities and relationships from plain texts which are not appropriate to apply directly on document understanding.

Document understanding is the process of automatically recognizing and extracting key texts from scanned unstructured documents and saving them as structured data. Document understanding was already introduced in a competition of ICDAR 2019, where the goal was to detect texts in documents and extract key texts from receipts and invoices. In this work, we focus on document understanding which is mainly about key information extraction from scanned unstructured documents. The following paragraphs summarize the challenges of the task and the contributions of our work.

1.1 Challenges

Document understanding is a challenging task and there are little research works published in this topic so far. Although it seems that traditional named entity recognition networks or layout analysis networks are related to this topic, none of the existing research can fully address the problems faced by document understanding.

Firstly, context requires balance. The key cue of the entities usually appears in their neighbors and too much context will add noise and increase problem dimensionality making learning slower and more difficult. As shown in Figure 1, in order to identify the label of $11900, the text Total on its left side is good enough for the model to recognize its tag correctly. Instead of increasing the recognition accuracy, too much context like Tax, Subtotal will lead the performance even worse. Appropriate context is very problem specific and we need to get this relationship by training.

Proceedings of the 19th China National Conference on Computational Linguistics, pages 853-862,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

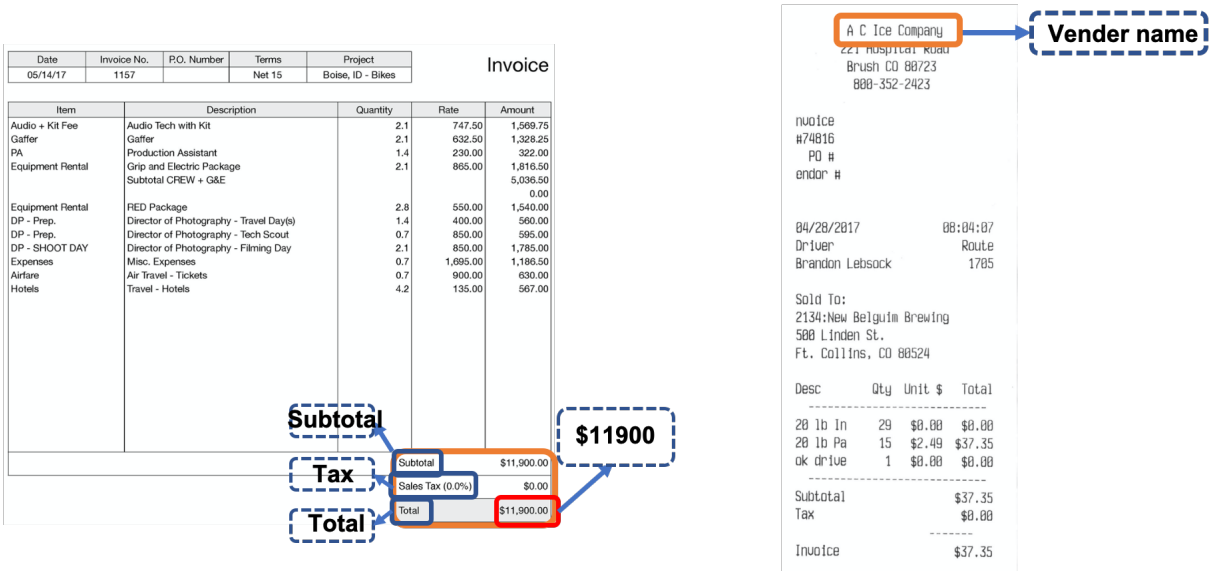


Figure 1: Examples of Documents and example entities to extract.

Secondly, it is not adequate to represent the semantic meaning in documents by using text alone. For example, there can be multiple date related entities in one document such as due date and purchase date. It is difficult for the model to distinguish them only by textual information. Thus, more information like visual information or layout information also needs to be considered at the same time.

Thirdly, the positional cue is critical sometimes. An example is shown in the right side of Figure 1. As for the entity Vender Name, it appears at the top of the document in most cases. The model will benefit from it if it can leverage this information.

1.2 Contributions

In this work, we present a novel method that achieves the document understanding problem as a node classification task. The method first computes a text embedding and an image embedding for each text segment in the document. Then graph construction algorithm will use the coordinates of bounding boxes to generate a unique graph for each document. In order to leverage positional cue effectively, the global node is first proposed in document understanding field which represents the universal context of the current document. Finally, the graph attention network will combine textual information with visual information and the positional cue for information extraction.

The main contributions of this paper can be summarized as follows: 1) we propose a graph construction algorithm to generate a unique graph for each document and achieve the document understanding task as a graph node classification task; 2) the proposed model can capture global context information and local compositions effectively; 3) extensive experiments have been conducted on real-world datasets to show that our method has significant advantages over the baseline methods.

2 Related Works

Several rule-based document understanding systems were proposed in (Chiticariu et al., 2013; Dengel and Klein, 2002; Schuster et al., 2013). Laura et al. (2013) presented a case for the importance of rule-based approaches to industry practitioners. SmartFix by Andreas et al. (2002) employs specific configuration rules designed for each template. The study by Schuster et al. (2013) offers a template matching based algorithm to solve the document understanding problem and plenty of templates have to be constructed and maintained to deal with different situations.

However, rule-based methods rely heavily on the predefined templates or rules and are not scalable and flexible for most document understanding problems since documents in real life have no fixed layout. Furthermore, updating the templates or rules requires a lot of effort.

A recent study by Zhao et al. (2019) proposed Convolutional Universal Text Information Extractor (CUTIE). CUTIE treats the document understanding task as an image semantic segmentation task. It applies convolutional neural networks on gridded texts where texts are semantical embeddings. However, this work only uses text-level features and doesn't involve image-level features.

Inspired by BERT (Devlin et al., 2018), Xu et al. (2019) proposed LayoutLM method. It applies BERT architecture for the pre-training of text and layout. Although LayoutLM uses image features in the pre-training stage and it performs well on several downstream tasks, the potential relationship between two text segments hasn't been taken into consideration. In addition, sufficient data and time are required to pre-train the model inefficiently.

Since graph neural networks (Scarselli et al., 2008; Kipf and Welling, 2016; Veličković et al., 2017) have shown great success in unstructured data tasks, more and more research works are focusing on using GNN to tackle the document understanding problem. Liu et al. (2019) presented a GCN-based method for information extraction from document images. It is a work attempting to extract key information with customized graph convolution model. However, prior knowledge and extensive human efforts are needed to predefine task-specific node and edge representations. One study by Yu et al. (2020) explores the feature fusion of textual and visual embeddings by GNN. This work differs from ours because it still treats the document understanding task as the sequence tagging problem and uses a bi-directional LSTM model to extract entities which has already been proved to have limited ability to learn the relationship among distant words.

3 Proposed Method

This section demonstrates the architecture of our proposed model. To extract textual context, our model first encodes each text segment in the document by pre-trained BERT model as its corresponding text embeddding. Then using multiple layers of CNN to get its image embedding. The combination of these two types of embeddings will generate unique global node representation and various local node representations. These node representations contain both visual context and textual context and will be used as node input to the graph attention network. Our model transforms the document understanding task into a node classification problem by taking both local context and global context into account.

3.1 Feature Extraction

Figure 2 is the overall workflow of feature extraction. As shown in Figure 2, we calculate node representations for both global nodes and local nodes where global nodes capture universal information and local nodes extract internal information. Different from the existing information extraction models that only use plain text features, we also use image features to obtain morphology information to our model.

3.1.1 Text Feature Extraction

We use pre-trained BERT model to generate text embeddings for capturing both global and local textual context. For a set of text segments in the document, we concatenate them by their coordinates from left to right and from top to bottom to generate a sequence. Given a sequence $seq_i = (w_1^{(i)}, w_2^{(i)}, ..., w_n^{(i)})$, text embeddings of a sequence seq_i are defined as follows

$$TE_{0:n}^{(i)} = BERT(w_{0:n}^{(i)}; \Theta_{BERT}) \tag{1}$$

where $w_{0:n}^{(i)} = [w_0^{(i)}, w_1^{(i)}, ..., w_n^{(i)}]$ denotes the input sequence padding with $w_0^{(i)} = [CLS]$. $[CLS]$ is a specific token to capture full sequence context which is introduced in (Devlin et al.,

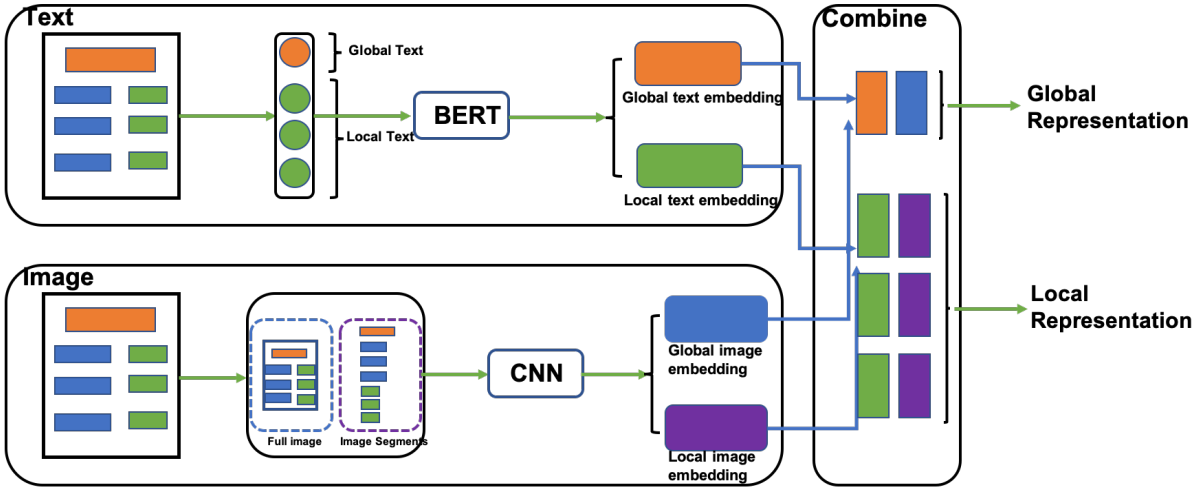


Figure 2: Workflow of feature extraction.

2018). $TE^{(i)}_{0:n} = [TE^{(i)}_0, TE^{(i)}_1, ..., TE^{(i)}_n] \in \mathbf{R}^{n*d_{model}}$ denotes the output sequence embeddings and d_{model} is the dimension of the model. $TE^{(i)}_k$ represents the k-th output of pre-trained BERT model for the i-th document. Θ_{BERT} represents the parameters of pre-trained BERT model. Each text segment of a text sequence is encoded independently and we can get global text embedding and local text embedding simultaneously, defining them as

$$TE^{(i)}_{Global} = TE^{(i)}_0 \tag{2}$$

$$TE^{(i)}_{Local} = [TE^{(i)}_1, TE^{(i)}_2, ..., TE^{(i)}_n] \tag{3}$$

3.1.2 Image Feature Extraction

For image embedding generation, we using CNN for catching both global and local visual information. Given a set of image segments cropped by bounding boxes $seg_i = (p^{(i)}_1, p^{(i)}_2, ..., p^{(i)}_n)$, image embeddings of segments seg_i are defined as follows

$$IE^{(i)}_{0:n} = CNN(p^{(i)}_{0:n}; \Theta_{CNN}) \tag{4}$$

where $p^{(i)}_{0:n} = [p^{(i)}_0, p^{(i)}_1, ..., p^{(i)}_n]$ denotes the input image segments appending with $p^{(i)}_0 = full\_image$. We use $p^{(i)}_0$ to capture global morphology information of the document image. $p^{(i)}_k \in \mathbf{R}^{H*W*3}$ represents k-th image segment of i-th document and H means height of the image, W means width of the image. $IE^{(i)}_{0:n} = [IE^{(i)}_0, IE^{(i)}_1, ..., IE^{(i)}_n] \in \mathbf{R}^{n*d_{model}}$ denotes the output image embeddings and d_{model} is the dimension of the model. In our work, we use classic ResNet model (He et al., 2016) as backbone to extract image features and a full connected layer is used to resize output to d_{model} dimension. $IE^{(i)}_k$ represents the k-th output of CNN model for the i-th document. Θ_{CNN} represents the parameters of CNN model. Each image segment is encoded independently and we can get global image embedding and local image embedding synchronously, defining them as

$$IE^{(i)}_{Global} = IE^{(i)}_0 \tag{5}$$

$$IE^{(i)}_{Local} = [IE^{(i)}_1, IE^{(i)}_2, ..., IE^{(i)}_n] \tag{6}$$

3.1.3 Combination

After text feature extraction and image feature extraction, we can concatenate these features into a new representation RE, which will be used as node input to the graph neural network. $\oplus$ in the formula means concatenation operation.

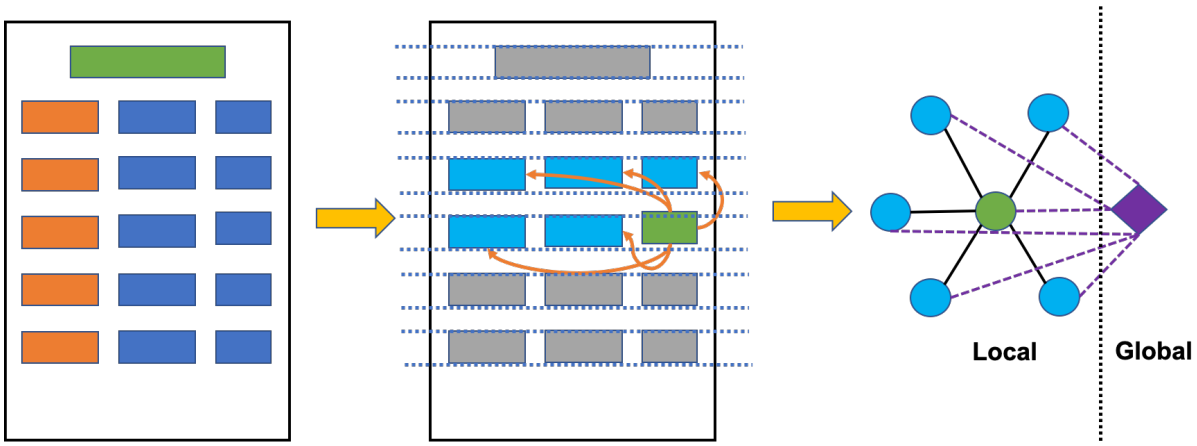


Figure 3: Illustration of graph construction.

$$RE^{(i)}_{Global} = TE^{(i)}_{0} \oplus IE^{(i)}_{0} \tag{7}$$

$$RE^{(i)}_{Local} = TE^{(i)}_{1:n} \oplus IE^{(i)}_{1:n} \tag{8}$$

3.2 Graph Construction

In order to capture relative positional information, we use the coordinates of bounding boxes to connect text segments. Inspired by Gui et al. (2019), we propose the global node mechanism which is used as a virtual hub to capture long-range dependency and high-level features.

The whole document is converted into a directed graph, as shown in Figure 3, where each node represents a text segment and the connection between two nodes can be treated as an edge. Given a set of text segments inside a document, first of all, we need to merge these text segments into different lines based on their bounding boxes' coordinates. To be more specific, if the overlap of the two text segments on the vertical axis exceeds 60%, the two text segments are considered to belong to the same line. In order to capture layout information, we build connection for each text segment in the same line. In addition, an extra connection is built between current text segment and every text segments in its previous line.

To capture global information, we add a global node to connect each local node. The global node is used as a virtual hub to collect universal information from all the nodes inside the graph. Since all internal nodes are connected with global node which means every two non adjacent nodes are two-hop neighbors, universal information can be distributed to these local nodes through such connections.

3.3 Recurrent-based Aggregate and Update

Attention-based graph neural network (Veličković et al., 2017) is applied to fuse multiple information in the graph, as shown in Figure 4. In our model, graph convolution is defined based on the self-attention mechanism and aggregation and update of global node and local node are treated equally.

Given a node v_i and its hidden state h_i which is initialized by RE, the output embedding of node v_i can be calculated by self-attention mechanism as the follows

$$\vec{h}'_i = \sigma(\sum_{j \in N_i} \alpha_{ij} W \vec{h}_j) \tag{9}$$

where $\vec{h}'_i$ is the aggregation and update of $\vec{h}_i$ and $\vec{h}_j$ is the hidden state of node v_i's neighbour v_j. σ is an activation function and α_{ij} is the attention coefficient which indicates the importance

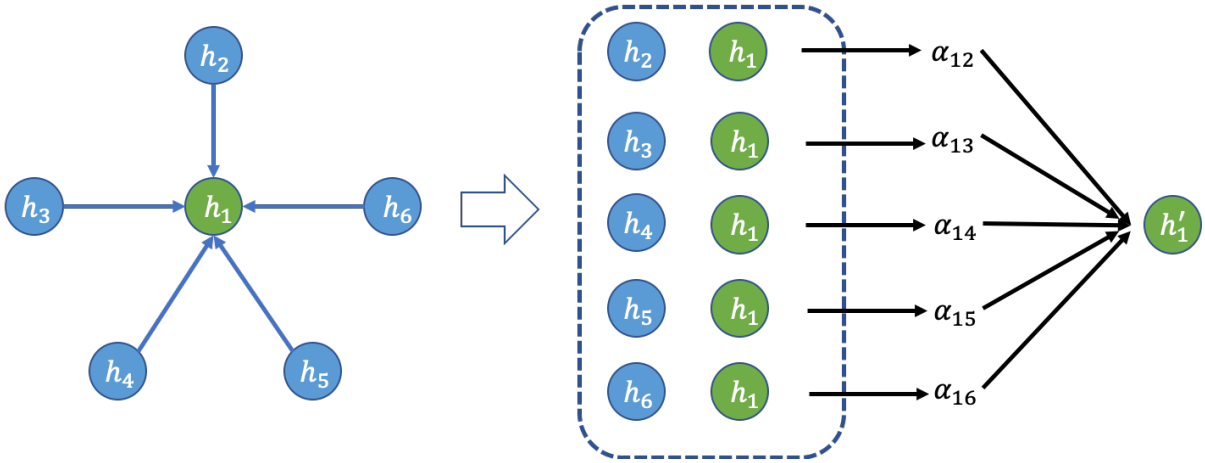


Figure 4: Aggregation in Graph Neural Network.

of node j's features to node i. The coefficients computed by the attention mechanism can be expressed as:

$$\alpha_{ij} = \frac{exp(LeakyReLU(V^T[Wh_i \oplus Wh_j]))}{\sum_{k \in N_i} exp(LeakyReLU(V^T[Wh_i \oplus Wh_k]))} \tag{10}$$

where W and V are trainable parameters. We apply the LeakyReLU nonlinearity (with negative input slope $\alpha = 0.2$) to avoid the "dying ReLU" problem.

Similarly to Vaswani et al. (2017), we also employ multi-head attention to improve the performance of our model. K attention mechanisms execute independently and their features are concatenated in the end. The final representation is as the follows and $\oplus$ in the formula means concatenation operation:

$$\vec{h}'_i = \mathop{\Big\Vert}_{k=1}^{K} \sigma(\sum_{j \in N_i} \alpha_{ij}^k W^k \vec{h}_j) \tag{11}$$

3.4 Decoding and Information Extraction

A conditional random field (CRF) is used to generate a family of conditional probability for the sequence. Given the sequence of final node states $h_{1:n}^{final} = [h_1^{final}, h_2^{final}, ..., h_n^{final}]$, and the probability of a label sequence $\hat{y} = [\hat{l}_1, \hat{l}_2, ..., \hat{l}_n]$ can be defined as the follows

$$p(\hat{y}|s) = \frac{exp(\sum_{i=1}^{n} W_{(l_{i-1},l_i)} h_i^{final} + b_{(l_{i-1},l_i)})}{\sum_{y' \in Y(s)} exp(\sum_{i=1}^{n} W_{(l'_{i-1},l'_i)} h_i^{final} + b_{(l'_{i-1},l'_i)})} \tag{12}$$

where W and b are the weight and bias parameters and $Y(s)$ is the set of all arbitrary label sequences.

Our model parameters of whole networks are jointly trained by minimizing the following loss function as:

$$L = -\sum_{i=1}^{N} log(p(y_i|s_i)) \tag{13}$$

Decoding of CRF layer is to search the output sequence y^* having the highest conditional probability for testing.

$$y^* = \underset{y \in Y(s)}{argmax}\, p(y|s) \tag{14}$$

Table 1: F1-score performance comparisons from contract dataset.

Entities	Bi-LSTM-CRF	BERT-CRF	Our model
Party A	72.2	75.3	79.1
Party B	83.5	84.2	88.4
Project Name	65.6	68.3	74.8
Contract Name	69.2	71.5	80.2
Contract Amount	86.3	89.8	92.3
Consortium Members	45.2	46.1	54.6
Macro Average	70.3	72.5	78.2

Viterbi algorithm is used to calculate the above equations, which can improve algorithm operation efficiency.

4 Experiments

We use Pytorch framework to implement our experiments on a GTX 1080Ti GPU and apply our model for information extraction from two real-world datasets.

4.1 Datasets

We conduct experiments on two document understanding datasets. (1) Contract Dataset: Contract Dataset is a dataset from Alibaba Tianchi Competition. The dataset contains six types of named entities: Party A, Party B, Project Name, Contract Name, Contract Amount and Consortium Members. This dataset has both the original PDF format documents and annotation files of target named entities. The train set consists of 893 contracts and test set consists of 223 contracts. (2) SROIE: SROIE is composed of scanned receipt images and is annotated with 4 types of named entities: Company, Address, Date and Total. The train set consists of 627 receipt images and test set consists of 347 receipt images.

4.2 Implementation Details

We use the Adam (Kingma and Ba, 2014) as the optimizer, with a learning rate of 3e-6 for all datasets. We employ the Dropout (Srivastava et al., 2014) with a rate of 0.5 for node aggregation and update. In the feature extraction part, the text feature extractor is pre-trained BERT model and the hyper-parameter of BERT used in our paper is same as (Devlin et al., 2018). The dimension of text embedding is 512. The image feature extractor is ResNet-50 model and the hyper-parameter of ResNet-50 used in our paper is same as (He et al., 2016). We add a full connected layer after ResNet-50 to resize the output dimension to 512. Then the combination of text embeddings and image embeddings is applied as the input of the graph neural network. We apply 3 graph attention layers with 24 multi-heads and the dimension of hidden state is 1024. The standard F1 score is used as evaluation metrics.

4.3 Evaluation

We compare the performance of our model with Bi-LSTM-CRF (Huang et al., 2015) and BERT-CRF (Devlin et al., 2018). Bi-LSTM-CRF uses Bi-LSTM architecture to extract text information and a CRF layer to get tags. BERT-CRF applies BERT model as backbone to replace Bi-LSTM model and also a CRF layer after to extract entities. The input text sequence is generated by text segments concatenated from left to right and from top to bottom according to (Palm et al., 2017).

4.4 Result

We report our experimental results in this section. Table 1 lists the F1 score of each entity of contract dataset. Macro-averages in the last row of the table are the averages of the corresponding columns, indicating the overall performance of each method on all entity types. In

Table 2: F1-score performance comparisons from SROIE dataset.

Entities	Bi-LSTM-CRF	BERT-CRF	Our model
Company	85.1	86.8	93.5
Address	88.3	89.1	94.6
Date	94.2	96.2	97.3
Total	83.5	84.7	92.1
Macro Average	87.8	89.2	94.4

Table 3: Ablation studies of individual component.

Configurations	contract dataset	SROIE dataset
full model	78.2	94.4
w/o visual feature	75.3	90.1
w/o global node	76.7	92.3

the contract scenario, as can be seen from Table 1, our model outperforms Bi-LSTM-CRF by 12% in F1 score and leads to a 8.00% increment of F1 score over BERT-CRF model. Moreover, our model outperforms the two baseline models in all entities. Further analysis shows that our model makes great improvements in those entities like Contract Name and Project Name. These entities have conspicuous layout features and morphological features which can't be captured by text alone models.

Furthermore, as shown in Table 2, our model shows significant improvement over the baseline methods on SROIE dataset. Compared with the existing Bi-LSTM-CRF model and BERT-CRF model, our model gives the best results by a large margin. These results suggest that, compared to previous text alone methods, our model is able to extract more information from the document to learn a more expressive representation through graph convolutions.

4.5 Ablation Studies

To study the contribution of each component in our model, we conduct ablation experiments on both two datasets and display the results in Table 3. In each study, we exclude visual features and the use of global node respectively, to see their impacts on F1 scores on both two datasets.

As described in Table 3, when we remove visual features, the result drops to the F1 score of 75.3 on contract dataset and 90.1 on SROIE dataset. This indicates that visual features can play an important role in addressing the issue of ambiguously extracting key information. Furthermore, the results show that the model's performance is degraded if the global node is removed, indicating that global connections are useful in the graph structure.

5 Conclusions and Future Works

This paper studies the problem of document understanding. In this work, we present a novel method that takes global context into account to refine the graph architecture on the complex documents. The explanatory experiments suggest that our proposed model is capable of extracting more information from documents to learn a more expressive representation through attention-based graph convolutions. We hope that our research will serve as a base for future studies on document understanding. Furthermore, we intend to extend our model to other document related tasks, such as document classification or document clustering.

Acknowledgements

This work was supported by The National Key Research and Development Program of China under grant 2017YFB0802704 and 2017YFB0802202.

References

Alan Akbik, Tanja Bergmann, and Roland Vollgraf. 2019. Pooled contextualized embeddings for named entity recognition. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 724–728.

Laura Chiticariu, Yunyao Li, and Frederick Reiss. 2013. Rule-based information extraction is dead! long live rule-based information extraction systems! In Proceedings of the 2013 conference on empirical methods in natural language processing, pages 827–832.

Andreas R Dengel and Bertin Klein. 2002. smartfix: A requirements-driven system for document analysis and understanding. In International Workshop on Document Analysis Systems, pages 433–444. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Tao Gui, Yicheng Zou, Qi Zhang, Minlong Peng, Jinlan Fu, Zhongyu Wei, and Xuan-Jing Huang. 2019. A lexicon-based graph neural network for chinese ner. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 1039–1049.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. arXiv preprint arXiv:1508.01991.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. arXiv preprint arXiv:1603.01360.

Xiaojing Liu, Feiyu Gao, Qiong Zhang, and Huasha Zhao. 2019. Graph convolution for multimodal information extraction from visually rich documents. arXiv preprint arXiv:1903.11279.

Rasmus Berg Palm, Ole Winther, and Florian Laws. 2017. Cloudscan-a configuration-free invoice analysis system using recurrent neural networks. In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), volume 1, pages 406–413. IEEE.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2008. The graph neural network model. IEEE Transactions on Neural Networks, 20(1):61–80.

Daniel Schuster, Klemens Muthmann, Daniel Esser, Alexander Schill, Michael Berger, Christoph Weidling, Kamil Aliyev, and Andreas Hofmeier. 2013. Intellix–end-user trained information extraction for document archiving. In 2013 12th International Conference on Document Analysis and Recognition, pages 101–105. IEEE.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1):1929–1958.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Advances in neural information processing systems, pages 5998–6008.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. arXiv preprint arXiv:1710.10903.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2019. Layoutlm: Pre-training of text and layout for document image understanding. arXiv preprint arXiv:1912.13318.

Wenwen Yu, Ning Lu, Xianbiao Qi, Ping Gong, and Rong Xiao. 2020. Pick: Processing key information extraction from documents using improved graph learning-convolutional networks. arXiv preprint arXiv:2004.07464.

Xiaohui Zhao, Endi Niu, Zhuo Wu, and Xiaoguang Wang. 2019. Cutie: Learning to understand documents with convolutional universal text information extractor. arXiv preprint arXiv:1903.12363.

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. 2017. Joint extraction of entities and relations based on a novel tagging scheme. arXiv preprint arXiv:1706.05075.

Combining Impression Feature Representation for Multi-turn Conversational Question Answering

Shaoling Jing[1,2,3,*] and **Shibo Hong**[2] and **Dongyan Zhao**[1] and **Haihua Xie**[2] and **Zhi Tang**[1]
1. Wangxuan Institute of Computer Technology, Peking University, Beijing, China
2. State Key Laboratory of Digital Publishing Technology,
Peking University Founder Group Co. LTD., Beijing, China
3. Postdoctoral Workstation of the Zhongguancun Haidian Science Park, Beijing, China
{jingshaoling, zhaody, tangzhi}@pku.edu.cn
{hongshibo, xiehh}@founder.com

Abstract

Multi-turn conversational Question Answering (ConvQA) is a practical task that requires the understanding of conversation history, such as previous QA pairs, the passage context, and current question. It can be applied to a variety of scenarios with human-machine dialogue. The major challenge of this task is to require the model to consider the relevant conversation history while understanding the passage. Existing methods usually simply prepend the history to the current question, or use the complicated mechanism to model the history. This article proposes an impression feature, which use the word-level inter attention mechanism to learn multi-oriented information from conversation history to the input sequence, including attention from history tokens to each token of the input sequence, and history turn inter attention from different history turns to each token of the input sequence, and self-attention within input sequence, where the input sequence contains a current question and a passage. Then a feature selection method is designed to enhance the useful history turns of conversation and weaken the unnecessary information. Finally, we demonstrate the effectiveness of the proposed method on the QuAC dataset, analyze the impact of different feature selection methods, and verify the validity and reliability of the proposed features through visualization and human evaluation.

1 Introduction

Conversational Question Answering (ConvQA) is a new question answering task that requires a comprehension of the context, which has recently received more and more attention (Zhu et al., 2018; Qu et al., 2019a; Qu et al., 2019b; Meng et al., 2019; Pruthi et al., 2020). Since conversation is one of the most natural ways for humans to seek information, it carries over context through the dialogue flow. Specifically, we ask other people a question, dependending on their answer, we follow up with a new question, and second answer with additional information will be given based on what has been discussed (Reddy et al., 2019). Therefore, multi-turn conversational question answering is formed in this way. It can be used in many fields as a personal assistant systems, such as, customer service, medical, finance, education, etc. Moreover, with the rapid development of artificial intelligence technology in theory and practical applications, many personal assistant products have been launched in the market, such as Alibaba AliMe, Apple Siri, Amazon Alexa, etc. Although these assistants are capable to cover some simple tasks, they cannot handle complicated information-seeking conversations that require multiple turns of interaction (Qu et al., 2019b).

In the tasks of two recent multi-turn ConvQA datasets, CoQA (Reddy et al., 2019) and QuAC (Choi et al., 2018), given a passage, a question, and the conversation context preceding the question, the task is to predict a span of passage as the answer or give an abstrctive answer based on the passage. So the machine has to understand a text passage and converstion history to answer a series of questions. Each conversation in the QuAC dataset is obtained by two annotators playing the roles of teacher (information-provider) and student (information-seeker) respectively. During the conversation, the student only has access to the heading of passage and tries to learn about a hidden Wikipedia passage by asking a sequence

Proceedings of the 19th China National Conference on Computational Linguistics, pages 863-873,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

of freeform questions. The teacher answers the question by providing a span of text in the passage, as in existing reading comprehension tasks SQuAD (Rajpurkar et al., 2016), and gives the dialog acts which indicate the student whether the conversation should follow up. The CoQA has abstractive answers involving adding a pronoun (Coref) or inserting prepositions and changing word forms (Fluency) to existing extractive answers (Yatskar, 2018). Both datasets contain yes/no questions and extractive answers. Compared with the CoQA[1], the QuAC[2] setting is similar to a user query on search engines. The latter is designed to model and understand information-seeking conversation, which is closer to the people's daily question-answering style conversation than other datasets. On the other hand, QAs in QuAC are mostly non-factoid QAs and 86% of the 100 questions are contextual questions which requires reading the history to resolve coreference to dialog and passage. Moreover, the main answer type of QuAC dataset is extractive, resulting experiments are not easily disturbed by other types of answer factors and are suitable for verifying the feasibility of the proposed method. Therefore, this article intends to use the QuAC dataset for ConvQA experiments.

Existing multiple turns of question answering methods (Qu et al., 2019b; Zhu et al., 2018; Yatskar, 2018; Huang et al., 2018) emphasize the influence of historical context on current questions. Some of methods (Zhu et al., 2018; Reddy et al., 2019) prepend history turns to the current question or use a recurrent structure to model the representations of history turns (Huang et al., 2018), which obtain a good performance but a lower training efficiency. Some methods (Choi et al., 2018) adopt a simple heuristic method to select immediate previous turns, but they do not work for complicated conversational behaviors. Some researches attend history embedding (Qu et al., 2019a) or attend history position to the current question (Qu et al., 2019b), but not applicable to several no span-based answers. In addition, according to the literature available, there is a great lack of public studies on selecting or re-weighting of the conversation history turns, and re-representing the current questions and passages. Therefore, this paper proposes an impression feature combined with conversational history. By simulating the process of human question answering, we calculate the correlation from the deep historical context to the current question and the complete semantic unit of the passage to form impression features, and use this feature to replace the position information. This solves the problem that the abstrctive answer is difficult to learn position information, and enhance the knowledge representation ability of the model.

In this paper, we propose a multi-turn conversational question answering model combining with impression features. In order to learn the useful information from the conversation history, we separately calculate the word-level inter attention and turn inter attention from the conversation history to the current question and the passage. Then the learned representation is used as impression feature and fed to BERT (Devlin et al., 2018) with other inputs. The final representation is used to predict the answers.

Therefore, the contributions are as follows:

(1) Design an impression feature representation. This feature helps the model to learn more accurate information from the context of the historical conversation turns and assists the model in understanding passage and conversation, which provides new insights to the ConvQA task.

(2) Adapt different feature selection methods to verify the impact of the proposed impression feature representation on the model.

(3) A multiple turn conversational question answering model combining impression features is proposed.

2 Related Work

ConvQA is closely related to Machine Reading Comprehension (MRC) and conversational system.

The ConvQA task is similar to the machine reading comprehension task (Rajpurkar et al., 2016), but the major difference from MRC is that the questions in ConvQA are organized in conversations (Qu et al., 2019b), such as CoQA (Reddy et al., 2019), QuAC (Choi et al., 2018). Some questions rely on the historical questions or answers through pronouns. For instance, there are two questions Q_1, Q_2 and an answer A_1 to Q_1. Q_1:"who is going to have a birthday?". A_1:"Grandma Li.". Q_2:"where she was

[1]https://stanfordnlp.github.io/coqa/

[2]http://quac.ai/

born?". Here, the pronoun "she" of Q_2 associates Q_2 with Grandma Li of A_1, which indicates that the A_2 depend on A_1. If the QA model does not use Q_1 and A_1, then it does not know who she refers to in Q_2, making it difficult for the model to accurately answer Q_2. However, the questions of traditional MRC datasets (such as SQuAD (Rajpurkar et al., 2016) and SQuAD2.0 (Rajpurkar et al., 2018)) are independent of each other and have no relevance. Compared with the traditional MRC task, multi-turn ConvQA based on MRC adds multiple turns of conversation history to the original MRC task, making the ConvQA task more suitable for human daily conversation habits.

The exsiting methods for ConvQA in (Qu et al., 2019a) and (Qu et al., 2019b) determine whether the token in the question and the passage appear in each round of the historical conversation, and take the distance from the history turn of answers to the current question as the relative position, finally use the embedding of the relative position as an input of BERT encoder (Devlin et al., 2018). These methods are simple and effective, but they are not applicable to some no span-based answers. Because the token in the abstractive answer may be synonymous with a word in the historical answer, not the same word. In this case, the relative position is invalid. Moreover, a large amount of redundant information may also be introduced, and there may be a possibility of over-learning. For example, for a long passage, the author divides the passage into several sub-passages, and learns the relationship between each sub-passage and the answers of the historical rounds. If a question is only related to one of the sub-passages, suppose p_0, and has nothing to do with another sub-passage p_1. The information learned by p_1 and the largely redundant information of history conversation turns might play a negative interference role for the model to find the answer, while answering the current question q_k. Therefore, this paper focuses on how to select historical context and integrate its information into current question and passage.

ConvQA is very similar to the Background Based Conversations (BBCs) which recently proposed in the field of conversational systems. The latter is proposed to generate a more informative response based on unstructured background knowledge. But most of the research is aimed at topic-specific field (Meng et al., 2019), such as the conversation for movies (Moghe et al., 2018; Zhou et al., 2018) and diverse set of topics of Wikipedia (Dinan et al., 2018). Therefore, question answering based on reading comprehension and BBCs, these two tasks have in common that when responsing to each current sentence, not only the passage or background, but also the historical conversational context must be considered. The difference is that the former pays more attention to the ability of the model to understand the passage. When answering questions, the passage is mainly learned, and the historical conversation is supplemented to make the answer more accurate. The latter pays more attention to the ability of the model to understand the conversational context. When making a response, the model mainly learns conversational context, and assists with reference to background knowledge, the purpose is to enable the conversation to continue while making the response more informative.

In terms of model structure, RNN-based structure and BERT-based model (Devlin et al., 2018) have certain effectiveness on ConvQA, MRC and BBCs tasks. The RNN-based model (Zhu et al., 2018) can learn the impact of historical questions and answers on the current question and passage, but it cannot learn the deep bidirectional context representation. The BERT-based model is proved to greatly improve the performance of ConvQA (Qu et al., 2019a; Qu et al., 2019b), but it lacks reasonable integration into the history turns of conversation. Therefore, this paper proposes a method to model the history turns of questions and answers, generate impression features, and integrate them into the current questio and passage to improve model performance.

3 Our Approach

3.1 Task and Notations Definition

The ConvQA task is defined as (Reddy et al., 2019) and (Choi et al., 2018), given a passage x, the k-th question q_k in the conversation and the history conversation H_k preceding q_k, the task is to predict the answer a_k to the question q_k. There are only extractive answers in dataset QuAC (Choi et al., 2018). So the task is to predict the text span a_k within passage x. For the question q_k, there is $k-1$ turns of history conversation, and i-th turn of history conversation H_k^i includes a question q_i and its groudtruth answer a_i, which is $H_k^i = \{q_k^i, a_k^i\}_{i=1}^{k-1}$.

In order to ensure that the latter part of the long passage can be learened by the model, we divide the given passage x into N parts with sliding window following the previous work (Devlin et al., 2018), it is denoted as $x = \{x_n\}_{n=1}^{N}$ and $x_n = \{x_n(t)\}_{t=1}^{T}$, where $x_n(t) \in \mathbb{R}^h$ refers to the representation of the t-th token in x_n, T is the sequence length and h is the hidden size of the token representation. The k-th question is denoted as $q_k = \{q_k(j)\}_{j=1}^{J}$, $q_k \in \mathbb{R}^{J \times h}$, where $q_k(j) \in \mathbb{R}^h$ refers to j-th token in q_k and J is the maximum question length. All $k-1$ turns of history question and answer sequences are represented as $H_k = \{H_k^i\}_{i=1}^{I}$, $H_k \in \mathbb{R}^{I \times M \times h}$, where I is the maximum number of history turns for all conversations. The i-th turn history conversation of the k-th question is denoted as $H_k^i = \{H_k^i(m)\}_{m=1}^{M}$, $H_k^i \in \mathbb{R}^{M \times h}$, where $h_k^i(m) \in \mathbb{R}^h$ is m-th token in H_k^i and M is the maximum length of history questions and answers.

3.2 Impression Feature Representation

Multiple NLP tasks obtained state-of-the-art results by using pre-trained language model BERT, which learned the deep bidirectional representations through transformer (Vaswani et al., 2017). Adaptive to this paper, the encoder of BERT model encodes the question q_k, the passage x and the proposed Impression Feature (ImpFeat) that attend the conversaitonal histories H_k into contextualized representation, which is shown in Figure 1. The input sequences composed of token-level questions q_k and passages x_n are fed into the BERT model. Then the BERT encoder generates the token-level contextualized representation based on the token embedding, segment embedding, position embedding and the proposed impression feature (the different color row in the orange dotted lines of Figure 1). Finally, based on the output representation, the answer span predictor calculate the probability of each token as the beginning and end of the answer. Among them, the proposed impression feature (red-cyan row in the orange dotted frame) generation is detailed in Figure 2.

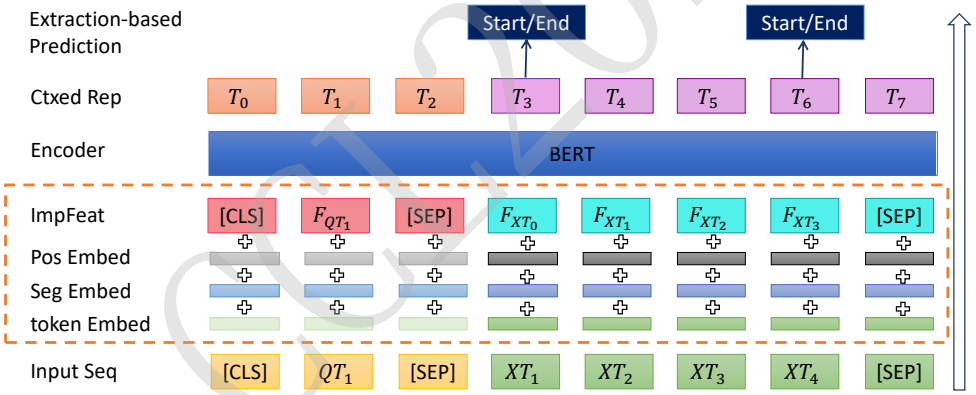


Figure 1: Our model with ImpFeat. It mainly reveals the process from the input of questions and passages (the light yellow-green row) to the contextualized representation (the pink-purple row), and then to the generation of answers (navy blue). This process includes the steps of inputting sequences, making features (marked by orange-dotted lines), BERT encoding, and predicting answers. The method of generating ImpFeat (red-cyan row in the right of Figure 2) from input sequence (the light yellow-green row in the left of Figure 2) is detailed in Figure 2.

As shown in Figure 2, the generation of impression features mainly includes two stages, word-level inter attention and turn inter attention. An input sequence contains a question q_k and a sub-passage x_n. For convenience, q_k is used as the representative of the input sequence in the following formula. The calculation method of the sub-passage x_n is the same as it. So the generation process is as follows.

Step 1: we follow word-level inter attention in the previous work (Zhu et al., 2018) to compute the attended vector from history turns of questions and answers to the input sequence. The relevance score matrix between j-th token of the current question and m-th history questions or answers is defined as Eq. 1:

$$r_j^i(m) = \tanh(U q_k(j)) D \tanh(U H_k^i(m)) \quad (1)$$

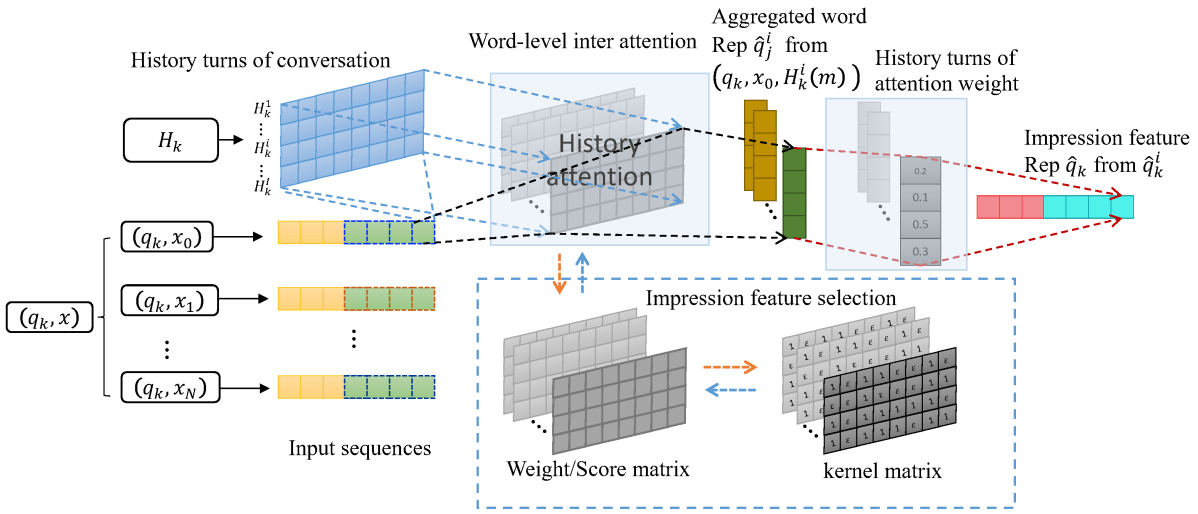


Figure 2: The proposed impression feature generation and selection using history attention. A sliding window approach is used to split a passage into sub-passages $(x_0, x_1, \cdots, x_N)$, which are then packed with the question q_k to form the input sequences $(q_k, x_0), (q_k, x_1), \cdots, (q_k, x_N)$. These input sequences share the same question. Then we generate the conversation history H_k of each input sequence. Take (q_k, x_0) for illustration, we did word-level inter attention and turn inter attention respectively. Word-level inter attention is applied to calculate attention $\hat{q}_k^i$ from each token of the conversational history to each token of the input sequence. Then turn inter attention is calculated from different history turns of conversation to the input sequence. In addition, we also make feature selection (in the blue dotted lines) for the obtained historical memory in word-level inter attention stage to make the memory is selective.

where, $r \in \mathbb{R}^{J \times I \times M}$, $D \in \mathbb{R}^{d \times d}$ is a diagonal matrix, and $U \in \mathbb{R}^{d \times h}$, d is the attention hidden size. The word-level attentive weight of m-th token in i-th history conversation to the j-th token of the current question q_k is represented as $\hat{\alpha}_j^i(m)$:

$$\hat{\alpha}_j^i(m) = \frac{e^{r_j^i(m)}}{\sum_{i'=1}^{I} \sum_{m'=1}^{M} e^{r_j^i(m)}} \tag{2}$$

Therefore, the aggregated word-level representation of all tokens in i-th history turn of conversation to the j-th token of the current question is represented as $\hat{q}_j^i$:

$$\hat{q}_j^i = \sum_{m=1}^{M} \hat{\alpha}_j^i(m) H_k^i(m) \tag{3}$$

Step 2: To learn the attention from different history turns of conversation to the input sequence, i.e. history turn inter attention, we learn an attention vector $D \in \mathbb{R}_I$ to compute attention weight from aggregated representation of i-th history turn of conversation to the current question. Initialize the weight matrix D with random values, then we get:

$$\hat{w}_i = \frac{e^{\hat{q}_j^i \cdot D}}{\sum_{i'=1}^{I} e^{\hat{q}_j^{i'} \cdot D}} \tag{4}$$

Further, the ImpFeat representation of all tokens of all history turns of conversation to the input question is denoted as $\hat{q}_k(j)$:

$$\hat{q}_k(j) = \sum_{i=1}^{I} \hat{w}_i \hat{q}_j^i \tag{5}$$

Step 3: To learn the attention within the tokens of the input question and passage, self-attention in Transformer structure (Vaswani et al., 2017) is applied here. So $\hat{q}_k(j)$ is refered as impression feature representation, and is merged with the token embedding, segment embedding and position embedding as the input of BERT.

The proposed two attention methods, and the self-attention in Transformer (Vaswani et al., 2017) respectively learn the attention from the tokens of history conversation to the input sequence, the attention from history turns to the input sequences, and the attention within the input sequence. So the model learns the historical information from different dimensions. Just like human reading, the model has a deep impression on historical information, which is why we express the learned representation as the impression feature. In addition, we also make feature selection for the obtained historical memory in word-level inter attention stage to make the memory is selective.

3.3 Impression Feature Selection

In order to verify whether the attention learned above is effective, and remove some redundant information. In step1, we use a kernel matrix to disturb the weights learned by the input sequence and history turns of conversation. Make

$$r_j^i = \sum_{m=1}^{M} r_j^i(m) \tag{6}$$

Then we sort r_j^i for each token of input sequence, select the historical turn number corresponding to the top s of r_j^i as the selected useful turn, which is represented as $r_j^{s'}$, $0 \leq s' \leq I$, and generate the corresponding kernel matrix :

$$a = \{a_j^i(m)\}_{1\leq i\leq I, 1\leq m\leq M}, a_j^i(m) = \begin{cases} 1, & \text{if } i = s' \\ \epsilon, & \text{otherwise} \end{cases} \tag{7}$$

where, ϵ is equals to a very small value, it is 0.001 in this paper. s is from 3 to 5 in this paper. $a_j^{s'}(m) = 1$ for all m in the s'-th turn. The new weight matrix after selection is represented as:

$$\alpha_j^i(m) = \hat{\alpha}_j^i(m) \cdot a_j^i(m) \tag{8}$$

where, $\alpha_j^i(m)$ represents that which history turns of conversation are more useful to the input sequence. Then we use the new weight matrix $\alpha_j^i(m)$ to replace $\hat{\alpha}_j^i(m)$ in Eq.(3), the q_k after adding impression feature selection is represented as:

$$q_j^i = \sum_{i=1}^{I} \alpha_j^i(m) H_k^i(m) \tag{9}$$

At last, use Eq.(9) and Eq.(5) to recalculate the ImpFeat representation.

4 Experiments

4.1 Data Description

The QuAC (Choi et al., 2018) dataset mentioned in the introduction is used for our experiment. It is a large-scale dataset contained more than 8,850 conversations and 98,400 questions. Statistics for this dataset is summarizied in Table 1, we can only access the training and validation data.

4.2 Experimental Setup

4.2.1 Competing Methods

The methods with published papers on QuAC leaderboard[3] are considered as baselines. To be specific, the competing methods are:

[3]http://quac.ai/

Table 1: Statistics of QuAC dataset.

Items	Training data	Validation data
Number of passages	6,843	1,000
Number of dialogs	11,567	1,000
Number of questions	83,568	7,354
Average questions per dialogs	7.2	7.4
Average tokens per passage	396.8	440.0
Average tokens per question	6.5	6.5
Average tokens per answer	15.1	12.3
Min/Avg/Med/Max history turns per question	0/3.4/3/11	0/3/5/3/11
% unanswerable	20.2	20.2

BiDAF++ (Choi et al., 2018; Peters et al., 2018): BiDAF++ is a re-implementation of a top-performing SQuAD model (Peters et al., 2018), which augments bidirectional attention flow (BiDAF) (Seo et al., 2016) with self-attention and contextualized embeddings.

BiDAF++ w/2-ctx (Choi et al., 2018): Based on BiDAF++, BiDAF++ w/r-ctx consider the context(ctx) from the previous r QA pairs. When $r = 2$, the model reached the best performance.

FlowQA (Huang et al., 2018): This model incorporate intermediate representations generated during the process of answering previous questions, thus it integrates the latent semantics of the conversation history more deeply than apporaches that just concatenate previous questions/answers as input.

BERT (Qu et al., 2019a): A ConvQA model with BERT is implemented and without any history modeling. We re-implement the model with batch size as 12 and marked with BERT_BZ12.

BERT + PHQA (Qu et al., 2019a): Based on BERT, this model adds conversation history by prepending history turn(s) to the current question. Here, PHQA prepends both history questions and answers. **BERT + PHA** prepends answers only.

BERT + HAE (Qu et al., 2019a): This approach model the conversation history by adding history answer embedding that denote whether a token is part of history answers or not.

BERT + PosHAE (Qu et al., 2019b): Based on BERT + HAE, This model learn position information of history turns by setting the distance from the historical turn to the current turn.

BERT + Att_PHQA : We implement a BERT-based ConvQA model that encode attention of history questions and answers (Att_PHQA), where, attention is computed from the prepended previous r QA pairs $(q_k, q_{k-1}, a_{k-1}, \cdots, q_1, a_1)$ to the input sequence (q_k, x_n). Here $r = 2$, i.e. $(q_k, q_{k-1}, a_{k-1}, q_{k-2}, a_{k-2})$.

BERT + Att_PHA: A BERT-based ConvQA model that encode attention of history answers only, where the prepended previous history is formed by $(q_k, a_{k-1}, a_{k-2}, \cdots, a_1)$. we set max answer length as 35 since it gives the best performance under this setting.

BERT + ImpFeat w/r-ctx: This is the solution we proposed in Section 3. The history turns of conversation H_k from the previous r QA pairs.

4.2.2 Hyper-parameter Settings and Implementation Details

In order to compare with methods similar to this article, such as BERT + HAE (Qu et al., 2019a), BERT + posHAE (Qu et al., 2019b), most of our experimental setting are the same as paper (Qu et al., 2019b), such as Tensorflow[4], v0.2 QuAC data, and BERT-Base Uncased model with the max sequence length of 384. The difference is that the batch size is set to 12, and the max answer length is set to 35 in BERT+ Att_PHA. The total training steps is set to 58000. Experiments are conducted on a single NVIDIA TESLA V100 GPU.

[4]https://www.tensorflow.org/

4.2.3 Evaluation Metrics

The QuAC challenge provides two evaluation metrics, word-level F1 and human equivalence score (HEQ) (Choi et al., 2018). Word-level F1 evaluates the overlap between prediction and references. HEQ is used to check if the system's F1 matches or exceeds human F1. It has two variants: (1) the percentage of questions for which this is true (HEQ-Q), and (2) the percentage of dialogs for which this is true for every question in the dialog (HEQ-D).

4.3 Experimental Results and Analysis

4.3.1 Main Evaluation Results

Table 2: Evaluation results on QuAC. Validation result of BiDAF++, FlowQA are from (Choi et al., 2018) and (Huang et al., 2018) ."-" means a result is not available.

Models	F1	HEQ-Q	HEQ-D
BiDAF++	51.8	45.3	2.0
BiDAF++ w/2-ctx	60.6	55.7	5.3
FlowQA	**64.6**	-	-
BERT	54.4	48.9	2.9
BERT + PHQA	62.0	57.5	5.4
BERT + PHA	61.8	57.5	4.7
BERT + HAE	63.1	58.6	6.0
BERT + PosHAE	**64.7**	**60.7**	**6.0**
BERT_Batchsize12	53.26	46.15	2.6
BERT + Att_PHQA	54.3	47.45	2.2
BERT + Att_PHA	62.48	57.74	5.3
BERT + ImpFeat w/11-ctx	63.02	58.54	**6.2**
BERT + ImpFeat w/4-ctx	**63.67**	**59.17**	5.9

The results on the validation sets are reported in Table 2. To implement the method of this article, we re-implement the BERT-based question answering model on the QuAC dataset, and set the batch size as 12. The result is slightly smaller 1% than the result in paper (Qu et al., 2019a), which is caused by the different hyperparameters setting. Moreover, we summarize our observations of the results as follows: (1) BERT + Att_PHA brings a significant improvement compared with BERT + PHA. This shows the advantage of using attention and suggests that making attention from history answer to the current question and passage plays an important role in coversation history modeling. (2) Computing attention with PHQA and PHA are both effective. BERT + Att_PHA achieves a higher performance compared to BERT + Att_PHQA, which indicates that all history answers contribute more information to the model than just the previous two turns of conversation history. (3) Our model (BERT + ImpFeat) obtains a substantially significant improvements over the BERT + Att_PHA model, but suffer the poor performance than FlowQA and BERT + PosHAE. One possible reason is that the impression feature has learned the token relevance from the context history to the current and passage, but it seems that there is still lack of topic flow and positional information of the conversation history, so that there is not enough improvement. (4) BERT + ImpFeat w/4-ctx outperform BERT + ImpFeat w/11-ctx, which indicates that the number of history pairs still affect the performance of the model, but four turns of context history may not be optimal result since we have not yet do experiments for all different history turns.

4.3.2 Ablation Analysis

In order to verify whether the proposed impression feature selection method is effective, we set different selection methods for comparison. Specifically, we randomly set the element of a in Eq.(7) to 1 or ϵ, then predict the answer. The results in Table 3 shows that after removing or replacing our feature selection method, the model performance drops significantly, indicating the importance of our proposed selection method.

Table 3: Results for ablation analysis. “w/o” means to remove or replace the corresponding component.

Models	F1	HEQ-Q	HEQ-D
BERT + ImpFeat w/4-ctx	**63.67**	**59.17**	**5.9**
w/o ImpFeat Selection	62.06	57.49	5.5
w/o Random Selection	23.75	23.02	0.6

Table 4: Results for human evaluation. Correctness, Completeness, Fluency are abbreviated as Cor, Com and Flu.

Evaluator	Cor	Com	Flu
A	4.07	4.74	4.71
B	4.06	4.79	4.74
C	4.0	4.68	4.54
Average	**4.04**	**4.73**	**4.66**

4.3.3 Impression Feature Analysis

To further analyze the impression feature, we randomly select an example and visualize the relationship between current question, passage, and conversation history, as shown in Figures 3 and 4, respectively. In Figures 3, the passage is from “..., faced ratio for 1963, and subsequent years. On May 11, Koufax no-hit the San Francisco Giants 8-0, besting future Hall of Fame pitcher Juan Marichal–himself a no-hit pitcher a month later, ... ”. The current question is from “Are there any other interesting aspects about this article? ”, and the sixth turn of history answer is parts of the passage. We can see that the tokens that are more relevant to the passage have a higher score and the stronger correlation, their corresponding color are redder, even white. On the contrary, the tokens that are less relevant to the passage have a lower score and the worse correlation, their corresponding color are darker. Furthermore, we can clearly see that there is a diagonal score that is generally large, because its answer exactly corresponds to the original answer. Besides, from Figure 4, we can see that the tokens such as ”powerful”, ”graints” in history answers are more relevant to the tokens ”change”, ”walks”, ”affect” and ”basketball” in the current question, indicating that the impression feature has learned relevant information from conversation history, and it is helpful to predict answers.

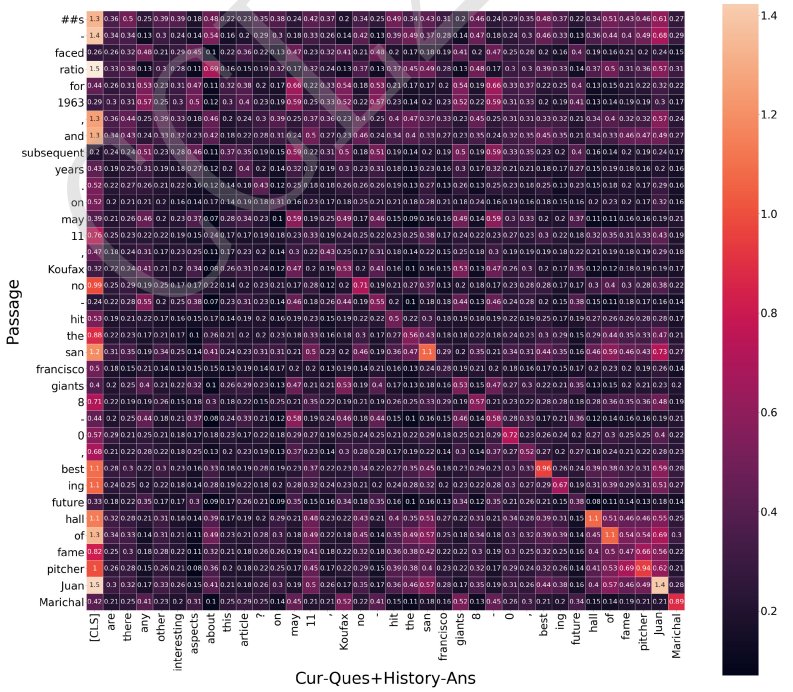


Figure 3: The heatmap of attention score from the current question and conversation history (Cur-Ques + History-Ans) to the passage. The first cloumn is the aggregated scores, the second to ninth tokens on the horizontal axis indicate the ninth current question, and the remaining tokens represent a part of the answer of the sixth turn conversation history. The vertical axis represents parts of passage tokens.

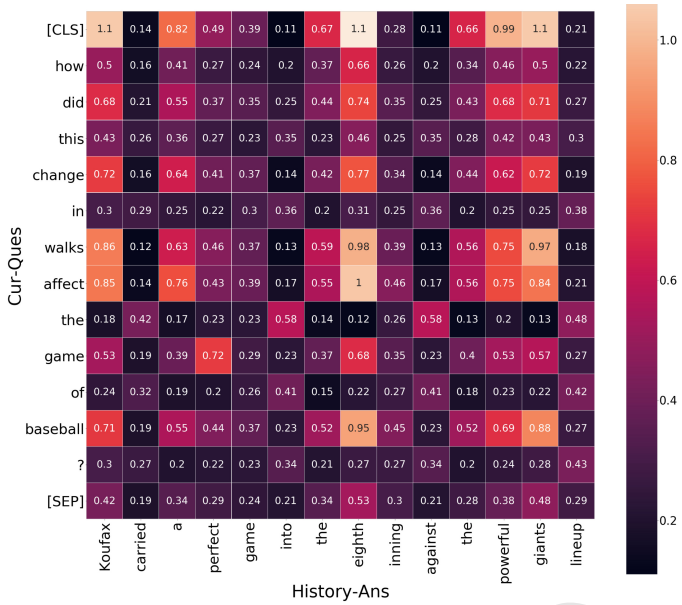


Figure 4: The heatmap of attention score from the conversational history answer (History-Ans) to the current question (Cur-Ques). The first row is the aggregated scores.

4.4 Human Evaluation

In addition, human evaluation is also conducted to verify the reliability of the proposed method. Three graduate students evaluate 100 randomly selected samples from the validation set results. Each sample contains one article and multiple QA pairs.

With reference to the subjective evaluation metrics commonly used in question generation research, we design correctness, completeness, and fluency to evaluate the predicted results. Correctness refers to the correctness of a predicted answer, evaluating whether a predicted answer is the same or related to the original answer, and whether it can be used to answer the question, etc. Completeness refers to the completeness of semantics, evaluating whether a predicted answer has the main components of the sentence, whether it is a complete sentence that is understandable to humans, and whether there are redundant words or missing words, etc. Fluency refers to the fluency of expression, evaluating whether a predicted answer is smooth, and whether the word order is correct, etc.

We divide the score into 1-5 based on three metrics. From 1 to 5, the predicted answer becomes more accurate, complete and fluent. Specifically, 1 means the predicted answer is completely incorrect, incomplete, or not fluent. And 5 means the answer is correct, complete, and fluent. Finally, the average score is calculated and shown in Table 4. The correctness, completeness, and fluency all exceed 4 points, indicating that most predicted answers are reasonable.

5 Conclusion and Future Work

Based on the general framework for ConvQA, we propose a new feature named impression feature, and combine the proposed feature with token embedding, position embedding and segment embedding as the input of BERT encoder. Then we introduce an impression feature selection method to select the important history information. Extensive experiments show the effectiveness of our method. Finally, we perform an in-depth analysis to show the different attention methods under different setting. Future work will consider to integrate multi-oriented information and a free-form answer type for ConvQA.

Acknowledgments

We thank all people who did human evaluation. This work are funded by China Postdoctoral Science Foundation (No.2019M660578), National Key Research and Development Program (No.2019YFB1406302), and Beijing Postdoctoral Research Foundation (No.ZZ2019-93).

References

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. Quac: Question answering in context. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2018. Wizard of wikipedia: Knowledge-powered conversational agents. *arXiv preprint arXiv:1811.01241*.

Hsin-Yuan Huang, Eunsol Choi, and Wen-tau Yih. 2018. Flowqa: Grasping flow in history for conversational machine comprehension. *CoRR*, abs/1810.06683.

Chuan Meng, Pengjie Ren, Zhumin Chen, Christof Monz, Jun Ma, and Maarten de Rijke. 2019. Refnet: A reference-aware network for background based conversation. *arXiv preprint arXiv:1908.06449*.

Nikita Moghe, Siddhartha Arora, Suman Banerjee, and Mitesh M. Khapra. 2018. Towards exploiting background knowledge for building conversation systems. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Danish Pruthi, Mansi Gupta, Bhuwan Dhingra, Graham Neubig, and Zachary C Lipton. 2020. Learning to deceive with attention-based explanations. *The 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, July.

Chen Qu, Liu Yang, Minghui Qiu, W. Bruce Croft, Yongfeng Zhang, and Mohit Iyyer. 2019a. Bert with history answer embedding for conversational question answering. *SIGIR'19: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information*, page 1133–1136.

Chen Qu, Liu Yang, Minghui Qiu, Yongfeng Zhang, Cen Chen, W. Bruce Croft, and Mohit Iyyer. 2019b. Attentive history selection for conversational question answering. *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 1391–1400, Nov.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, Mar.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Mark Yatskar. 2018. A qualitative comparison of coqa, squad 2.0 and quac. *arXiv preprint arXiv:1809.10735*.

Kangyan Zhou, Shrimai Prabhumoye, and Alan W Black. 2018. A dataset for document grounded conversations. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2018. Sdnet: Contextualized attention-based deep network for conversational question answering. *CoRR*, abs/1812.03593.

Chinese Long and Short Form Choice Exploiting Neural Network Language Modeling Approaches

Lin Li
Utrecht University
Utrecht, the Netherlands
Qinghai Normal University
Xining, China
l.li1@uu.nl

Kees van Deemter
Utrecht University
Utrecht, the Netherlands
c.j.vandeemter@uu.nl

Denis Paperno
Utrecht University
Utrecht, the Netherlands
d.paperno@uu.nl

Abstract

Lexicalisation is one of the most challenging tasks of Natural Language Generation (NLG). This paper presents our work in choosing between long and short forms of elastic words in Chinese, which is a key aspect of lexicalisation. Long and short forms is a highly frequent linguistic phenomenon in Chinese such as 老虎-虎 *(laohu-hu, tiger)*. The choice of long and short form task aims to properly choose between long and short form for a given context to producing high-quality Chinese.

We tackle long and short form choice as a word prediction question with neural network language modeling approaches because of their powerful language representation capability. In this work, long and short form choice models based on the-state-of-art Neural Network Language Models (NNLMs) have been built, and a classical n-gram Language Model (LM) is constructed as a baseline system. A well-designed test set is constructed to evaluate our models, and results show that NNLMs-based models achieve significantly improved performance than the baseline system.

1 Introduction

The long and short form of an elastic word refers to words have different word length (i.e. number of syllables) but share at least one identical word meaning such as 丢失-丢 *(diushi-diu, lose)*. Duanmu(Duanmu, 2013) points out that as high as 80% percent of Chinese words has both long and short forms, therefore Chinese speakers need to make the choice between long and short forms during daily communication. Like human speakers and writers, the long and short form choice task also needs to be carefully resolved for various domain including Natural Language Generation(Inkpen and Hirst, 2004), Machine Translation(Nguyen and Chiang, 2017), and Style Transfer(Fu et al., 2018).

In this work, we focus on long and short forms that share at least one same word meaning and one same morpheme, but compose of different number of syllables. The long and short form choice task is formulated as Fill-in-the-blank (FITB) task(Inkpen and Hirst, 2004; Zweig et al., 2012), whose goal is to select a missing word for a sentence from a set of candidates. A FITB example used in this work is shown in Table 1.

Sentence	Long Form	Short Form
她去日本旅游时，必逛各种免税 _____。	*(1)* 商店	*(2)* 店
When travels to Japan, she must go to duty free_____.	*(1) shop*	*(2) shop*

Table 1: A long and short form choice FITB question example.

The lexical choice is difficult in the context of long and short forms for most language processing systems due to the identical word sense leading to their preceding and subsequent contexts are too similar to providing distinguishing information. To address this problem, we investigate in learning language representation by LMs to making elegant choice of long and short

forms. This paper makes the following contributions: (1) propose long and short form choice models by making use of language modeling approaches LSTM-RNN LM and pre-trained LM (BERT(Devlin et al., 2018) and ERNIE(Sun et al., 2019) (2) to compare the performance of different LMs, constructing a well-designed test set for long and short form choice task.

The remainder of this paper is organized as follows. In Section 2, we discuss related work. Section 3 describes the language modeling methods we have used for our research and introduce our models. Section 4 presents our experimental results. We conclude with a discussion in Section 5.

2 Related work

A lot of words can be expressed by either a long form or a short form(Packard, 2000), for instance, elastic word, abbreviation, reduplication. In this work, we focus on the choice of long and short form of elastic words, that is, to choose between the long form (disyllabic) and short form (monosyllabic) of an elastic word that shares one morpheme and at least one same word meaning, and are interchangeable in some contexts(Duanmu and Dong, 2016). Previous work(Guo, 1938; Duanmu, 2013; Duanmu and Dong, 2016; Huang and Duanmu, 2013) show that as high as 90% Chinese word has long and short forms, which is a key issue in Chinese lexical choice. Li et al.(2019) investigated the problem of long and short form choice through human and corpus-based approaches, whose results support the statistical significant correlation between word length and the predictability of its context. Most previous work investigate the distribution and preference of long and short form based on corpus. It is still an open question to automated choose between long and short forms for a given context.

We framed the choosing between long and short forms as a FITB task proposed by Edmonds (1997) in English near-synonyms choice. Unsupervised statistical approaches were applied to accomplish FITB task in near-synonym choice, for instance, Co-occurrence Networks(Edmonds, 1997) and Pointwise Mutual Information (PMI)(Inkpen, 2007) were used to build up near-synonym choice model separately. Wang and Hirst(2010) explore lexical choice problem by capturing high dimensional information of target words ant their contexts thorough Latent Semantic Space.

Language models have obtained excellent performance in many language processing tasks, thus they have been also used to tackle the lexical choice task. A 5-gram language model(Islam and Inkpen, 2010) was trained from a large-scale Web corpus to choosing among English near-synonyms, following which Yu et al.(2011) implemented n-gram language model to Chinese near-synonym choice. N-gram model shows a better accuracy than PMI in near-synonym choice which is similar to our task. Neural Language Models overcome the limitation of n-gram language model by its powerful capability of long-range dependency. Recurrent Neural Networks (RNN)(Mirowski and Vlachos, 2015) and it variation Long-short Term Memory (LSTM)(Tran et al., 2016). Zweig et al.(Zweig et al., 2012) tackled the sentence completion problem with various approaches like language models. NNLMs achieved a better performance in these work, whose improvement can be attributed to its capability of capturing global information.

3 Long and Short Form Choice via Language Models

Language modeling is an effective approach to solve the task by computing occurrence probability of each candidate words. Given a context, the best long and short form can be chosen according to the probability acquired from language models. The state-of-the-art language modeling techniques and apply them to our task is described in this section.

3.1 N-gram Language Model

An input sentence S contains n words, i.e.,

$$S = \{w_1 w_2 w_3 ... w_i ... w_{n-2} w_{n-1} w_n\} \quad (1)$$

where w_i (i^{th} word of the sentence), denotes the lexical gap. The candidate words for the gap is $w_i = \{w_{long}, w_{short}\}$. Our task is to choose the w_i that best matches with the context.

N-gram language model, a classical probability language model, has succeeded in many previous work(Zweig et al., 2012; Yu et al., 2011; Islam and Inkpen, 2010) by capturing contiguous word associations in given contexts. A n-gram smoothed model(Islam and Inkpen, 2010) for long/short word choice is used as our baseline model, whose key idea of acquiring the probability of a string is defined as follow:

$$P(S) = \prod_{i=1}^{p+1} P(w^i|w_{i-n+1}^{i-1}) = \prod_{i=1}^{p+1} \frac{C(w_{i-n+1}^{i}) + M(w_{i-n+1}^{i-1})P(w_i|w_{i-n+2}^{i-1})}{C(w_{i-n+1}^{i-1}) + M(w_{i-n+1}^{i-1})} \tag{2}$$

$$M(w_{i-n+1}^{i-1}) = C(w_{i-n+1}^{i-1}) - \sum_{w_i} C(w_{i-n+1}^{i}) \tag{3}$$

where p is the number of words in the input sentence, i is the word position, $C(w_{i-n+1}^{i})$ and $C(w_{i-n+1}^{i-1})$ denotes the occurrence of the n-gram in the corpus, $P(w_i|w_{i-n+2}^{i-1})$ is the probability of w_i occurs given the words w_{i-n+1}^{i-1}, missing count $M(w_{i-n+1}^{i-1})$ is defined as 2.

The lexical gap of the input sentence S is replaced by long and short form separately, as follow:

$$S_1 = \{w_1 w_2 w_3 ... w_{long} ... w_{n-2} w_{n-1} w_n\}$$
$$S_2 = \{w_1 w_2 w_3 ... w_{short} ... w_{n-2} w_{n-1} w_n\}$$

Equation 1 is used to calculate $P(S_1)$ and $P(S_2)$, and take the target word in the sentence with higher probability as result. A disadvantage of n-gram model is not capable of maintaining long distance dependencies that play important role on long/short word choice. Hence, we proposed a neural language model to accomplish our task.

3.2 Recurrent Neural Networks (RNNs) Language Model

N-gram LM assigns probabilities to sentences by factorizing their likelihood into n-grams, whose modeling ability is limited because of data sparsity and long-distance dependency problem. NNLM have been proposed to model NL bymikolov2010recurrent, and outperform N-gram LM in many tasks(Mirowski and Vlachos, 2015; Tran et al., 2016) due to its ability of (1) each word w is represented as a low-dimensional density vector (2) retain long-span context information, which is failed captured by n-gram language model.

Recurrent Neural Networks (RNNs) have shown impressive performances on many sequential modeling tasks, thus we hypothesize that the performance of long/short form choice can be improved by adopting RNNs LM. Training a RNNs LM is difficult because of the vanishing and exploding gradient problems. Several variants of RNNs have been proposed to tackle with these two problems, among which Long Short-Term Memory is one of the most successful variants. In this work, we employ LSTM-RNNLMs to solve long/short form choice question. The LSTM adopted in this work is described as follows:

$$i_t = \sigma(U_i x_t + W_i s_{t-1} + V_i c_{t-1} + b_i)$$
$$f_t = \sigma(U_f x_t + W_f s_{t-1} + V_f c_{t-1} + b_f)$$
$$g_t = f\left(U x_t + W s_{t-1} + V c_{t-1} + b\right)$$
$$c_t = f_t \bigodot c_{t-1} + i_t \bigodot g_t$$
$$o_t = \sigma\left(U_o x_t + W_o s_{t-1} + V_o c_t + b_o\right)$$
$$s_t = o_t \cdot f\left(c_t\right)$$
$$y_t = g\left(V s_t + M x_t + d\right)$$

where x_t is input vector and y_t is output vector at time step t, i_t, f_t, o_t are input gate, forget gate and output gate respectively. c_{t-1} is the internal memory of unit, s_{t-1} is the LSTM hidden state at the previous time step. The uppercase (e.g., U_i and W) are weight matrices, the lowercase (e.g., b_i and b) is bias. f is the activation function and σ is the activation function for gates. The symbol $\odot$ is the Hadamard product or element-wise multiplication. Because of the architecture of LSTM-RNNLMs, the model has the potential to model long-span dependency.

3.3 Pre-trained Language Models

Language modeling aims to predict a distribution over a large scale of vocabulary items, by which solving the long/short form choice is a hard objective for our LSTM-RNNs acquired by limited size of training set and computation resource. We have a implicit assumption that the use of a powerful pre-trained language model is helpful to our task. Large-scale language models have achieved great success in many different Natural Language Understanding tasks. In this work, we focus on tackle our research question two very largely publicly LMs BERT and ERNIE.

LSTM-RNN LMs usually use the n preceding words as input to predict the next word $n+1$, which cannot capture subsequent words of the word $n+1$. BERT tackle this problem by retaining information of all the words in some fixed-length sequence. Thus, we re-implemented BERT as a long and short form predictor to assign probability for a target word in a given context. BERT' s model architecture is a multi-layer bidirectional Transformer encoder, whose success can be largely attributed to its Multi-Head Attention mechanism. By the attention mechanism, BERT is able to solving problems by learning the best representation through computing a weighted sum of the values of all words. The BERT-Base Chinese model adopted in this work is trained on a large scale of Chinese Simplified and Traditional corpus (based on an architecture of 12 layers, 768 hidden units, 12 heads, and 110M parameters). We tested the Bert with the methodology we used to test LSTM-RNNs.

ERNIE is a knowledge integration language representation model for Chinese, whose language representation is enhanced by using entity-level and phrase-level masking strategies in addition to a basic-level masking strategy. ERNIE has the same model structure as BERT-base, which uses 12 Transformer encoder layers, 768 hidden units and 12 attention heads.

4 Experiments and Results

Our baseline is a smoothed 4-gram language model, described in section 3.1. In our training data set, we keep the words occurring at least 50 times, and filter out 2-gram, 3-gram, and 4-gram that occur less than three times. For the model based on LSTM-RNN LM, we set the word embeddings as 300, the LSTM hidden states as 128, sentence max length as 50, and learning rate as 0.1.

4.1 Data Resources

A large scale corpus is used in this work, which is Chinese online news in June 2012 (approximately contains 64M Chinese words)[1]. We split the corpus into two parts: 90% of the corpus is used for training and 10% for testing. The same training set is employed to train the 5-gram LM and LSTM-RNN LM, which ensure the comparability of these two models.

To test our models, we carefully construct a test set based on the corpus. Firstly, we randomly choose 175 different long/short forms from. Then, 6 sentences for each of these long/short forms are extracted from the corpus, in which the sentences contain the same number of long and short forms. Finally, we get a test set by slightly editing these sentences manually, which consists of 1050 sentences.

[1]https://www.sogou.com/labs/resource/cs.php

4.2 Results

Table 2 summarizes our results tested by the identical test set, which shows that all our models based on NNLMs approaches perform better than the baseline model. The improvement in accuracy of LSTM-RNN is 3.43%; the accuracy has been improved 10.96% by adopting BERT; and ERNIE performs the best in our task whose accuracy reaches 82.67%. Our results show that NNLMs is more capable than Ngram LM in long and short form choice task. We think our model based on LSTM-RNNs LM is not as well-performed as the two pre-trained NNLMs is because of its simpler neural network architecture and a smaller training set.

4.3 Post-hoc Analysis

According to semantic relation of the two morphemes of long forms, the long and short forms can be categorized into 7 groups(Li et al., 2019). The X-XX category refers to reduplicated long and short forms such as 妈妈-妈 *(mama-ma, mother) or* 仅仅-仅 (jinjin-jin, only). All our models perform very well in predicting X-XX especially 5-gram LM performing the best, which suggests that the local context makes more contribution to the reduplication form choice than to other categories. Comparing with other categories of long and short forms, our models based on LSTM-RNN and ERNIE obviously perform bad in X-0X category, whose accuracy of this X-0X[2] is significant lower than the average accuracy (20.00% and 14.33% respectively). We think this is due to the comparatively low frequency of X-0X according to observation of our train set for LSTM-RNN LM.

Method	5-gram	LSTM-RNN	BERT	ERNIE
X-X'X	60.67%	77.33%	82.67%	88.00%
X-X0'	59.33%	78.00%	82.67%	78.67%
X-XY	62.67%	73.33%	82.67%	90.67%
X-0'X	66.67%	75.33%	75.33%	86.00%
X-XX	96.67%	88.00%	84.67%	87.33%
X-0X	71.33%	53.33%	76.00%	68.00%
X-X0	72.00%	68.00%	82.00%	80.00%
Accuracy	69.90%	73.33%	80.86%	82.67%

Table 2: Accuracy of language modeling methods tested by identical data set.

5 Conclusion

In this paper, we have investigated methods for answering long short form choice question. This question is significant because it is a key aspect of lexical choice which is still not well solved by many language processing systems. Through this work, we find that both all NNLM-based models do obviously outperform than Ngram LM. And our results show that all models perform very well in X-XX category but not very well in X-0X category. Our future work will be in the direction of eliminating the bias from NNLMs. Human evaluation for long and short form choice models also will be our further research content.

6 Acknowledgements

The first author of this paper received support from Qinghai Natural Science Foundation under Grant 2016-ZJ-931Q, Qinghai Major R&D Transformation Foundation under Grant 2019-GX-162, and National Natural Foundation under Grant 61862055, which is gratefully acknowledged.

[2]X-0X refers the long and short form like 小麦-麦 *(xiaomai-mai, wheat)*

References

Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. 1981. Alternation. *Journal of the Association for Computing Machinery*, 28(1):114–133.

San Duanmu. How many chinese words have elastic length. *Eastward flows the Great river: Festschrift in honor of Prof. William S.-Y. Wang on his 80th birthday*, pages 1–14, 2013.

Philip Edmonds. Choosing the word most typical in context using a lexical co-occurrence network. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 507–509. Association for Computational Linguistics, 1997.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. Style transfer in text: Exploration and evaluation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Diana Zaiu Inkpen and Graeme Hirst. Near-synonym choice in natural language generation. In *Recent Advances in Natural Language Processing*, volume 3, pages 141–152, 2004.

Toan Q Nguyen and David Chiang. Improving lexical choice in neural machine translation. *arXiv preprint arXiv:1710.01329*, 2017.

Geoffrey Zweig, John C Platt, Christopher Meek, Christopher JC Burges, Ainur Yessenalina, and Qiang Liu. Computational approaches to sentence completion. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 601–610. Association for Computational Linguistics, 2012.

Jerome L Packard. *The morphology of Chinese: A linguistic and cognitive approach*. Cambridge University Press, 2000.

Shaoyu Guo. the function of elastic word length in Chinese. *Yen Ching Hsueh Pao*, 24:1–34, 1938.

San Duanmu and Yan Dong. Elastic words in chinese. *The Routledge Encyclopedia of the Chinese Language*, pages 452–468, 2016.

Lijun Huang and San Duanmu. a quantitative study of elastic word length in modern Chinese. *Linguistic Sciences*, 12(1):8–16, 2013.

Yan Dong. *The prosody and morphology of elastic words in Chinese: annotations and analyses*. PhD thesis, University of Michigan, 2015.

Lin Li, Kees van Deemter, Denis Paperno, and Jingyu Fan. Choosing between long and short word forms in mandarin. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 34–39, 2019.

Philip Edmonds. Choosing the word most typical in context using a lexical co-occurrence network. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 507–509. Association for Computational Linguistics, 1997.

Tong Wang and Graeme Hirst. Near-synonym lexical choice in latent semantic space. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1182–1190. Association for Computational Linguistics, 2010.

Liang-Chih Yu, Wei-Nan Chien, and Shih-Ting Chen. A baseline system for chinese near-synonym choice. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1366–1370, 2011.

Aminul Islam and Diana Inkpen. Near-synonym choice using a 5-gram language model. *Research in Computing Sciences*, 46:41–52, 2010.

Mary Gardiner and Mark Dras. Predicting word choice in affective text. *Natural Language Engineering*, 22(1):97–134, 2016.

Piotr Mirowski and Andreas Vlachos. Dependency recurrent neural language models for sentence completion. *arXiv preprint arXiv:1507.01193*, 2015.

Ke Tran, Arianna Bisazza, and Christof Monz. Recurrent memory networks for language modeling. *arXiv preprint arXiv:1601.01272*, 2016.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368, 2019.

Dilin Liu. Is it a chief, main, major, primary, or principal concern?: A corpus-based behavioral profile study of the near-synonyms. *International Journal of Corpus Linguistics*, 15(1):56–87, 2010.

Diana Inkpen. A statistical model for near-synonym choice. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(1):1–17, 2007.

Sun, Yu and Wang, Shuohuan and Li, Yukun and Feng, Shikun and Tian, Hao and Wu, Hua and Wang, Haifeng Ernie 2.0: A continual pre-training framework for language understanding arXiv preprint arXiv:1907.12412,2019.

Devlin, J., Chang, M. W., Lee, K., Toutanova, K. Pre-training of deep bidirectional transformers for language understanding. CoRR, abs/1810.04805,2018.

Mikolov, M. Karafi´at, L. Burget, J. Cernock´y, and S. Khudanpur. Recurrent neural network based language model. In INTERSPEECH, pages 1045–1048, 2010.

Refining Data for Text Generation

Qianying Liu[12], **Tianyi Li**[1], **Wenyu Guan**[1] **and Sujian Li**[1]
[1] Key Laboratory of Computational Linguistics, MOE, Peking University
[2] Graduate School of Informatics, Kyoto University
ying@nlp.ist.i.kyoto-u.ac.jp; litianyi01@pku.edu.cn ;
guanwy@pku.edu.cn; lisujian@pku.edu.cn

Abstract

Recent work on data-to-text generation has made progress under the neural encoder-decoder architectures. However, the data input size is often enormous, while not all data records are important for text generation and inappropriate input may bring noise into the final output. To solve this problem, we propose a two-step approach which first selects and orders the important data records and then generates text from the noise-reduced data. Here we propose a learning to rank model to rank the importance of each record which is supervised by a relation extractor. With the noise-reduced data as input, we implement a text generator which sequentially models the input data records and emits a summary. Experiments on the ROTOWIRE dataset verifies the effectiveness of our proposed method in both performance and efficiency.

1 Introduction

Recently the task of generating text based on structured data has attracted a lot of interest from the natural language processing community. In its early stage, text generation (TG) is mainly accomplished with manually compiled rules or templates, which are inflexible and mainly based on expert knowledge (Kukich, 1983; Holmes-Higgin, 1994; Reiter and Dale, 1997). With the development of neural network techniques, especially sequence-to-sequence (seq2seq) models, generating short descriptive texts from structured data has achieved great successes, including generating wikipedia-style biographies (Lebret et al., 2016; Sha et al., 2017) and restaurant introductions (Novikova et al., 2017).

However, the task of generating long text, such as generating sports news from data, still fails to achieve satisfactory results. The existing models often forge fake context, lose sight of key facts and display inter-sentence incoherence (Wiseman et al., 2017). For the sports news generation task, one challenging problem is that the input records are both large and noisy. Specifically, the inputted box scores, which contains hundreds of data records, belong to 40 different categories, such as fouls, three-pointer, starting position and so on. Meanwhile, not all of the inputted records are reflected in the sports news, and there exists a serious non-parallelism between data records and texts. According to our statistics for 3000 parallel sports news and its data records which is shown in Table 1 and Figure 1, an average of only 19.3 data records out of 670.6 are mentioned in the summaries on average, namely only less than 5% of the data records are reflected in the human written news and rest 95% of them may bring noise into the model. Such large and noisy input has also caused the parameter amount of the embedding and encoder layer to be enormous, which leads to massive memory usage and limits the computation speed. In such situation, it is essential to refine data records and choose those important information before generating the final text.

In addition, sport news is far more complex than short descriptive text in that they need to consider overall coherence (Bosselut et al., 2018). For example, it would be weird if there is an abrupt topic change between neighboring sentences. If we just pour all the data records with no order into a model, it would be difficult for the summarization model to learn content planning by itself. Thus, it is a good practice to order the data records before text generation.

As stated above, in this paper, we propose to refine data records for the data-to-text generation task by training a model to select an appropriate subset of data records, which carries the key facts of the

Proceedings of the 19th China National Conference on Computational Linguistics, pages 881-891,
Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

Object	Number
Average Data Records Mentioned	19.30
Average Data Records in Box Data	670.65
Average Summary Length	348.93
Types of Data Records	40

Table 1: Statistics of data records in 3000 sports news.

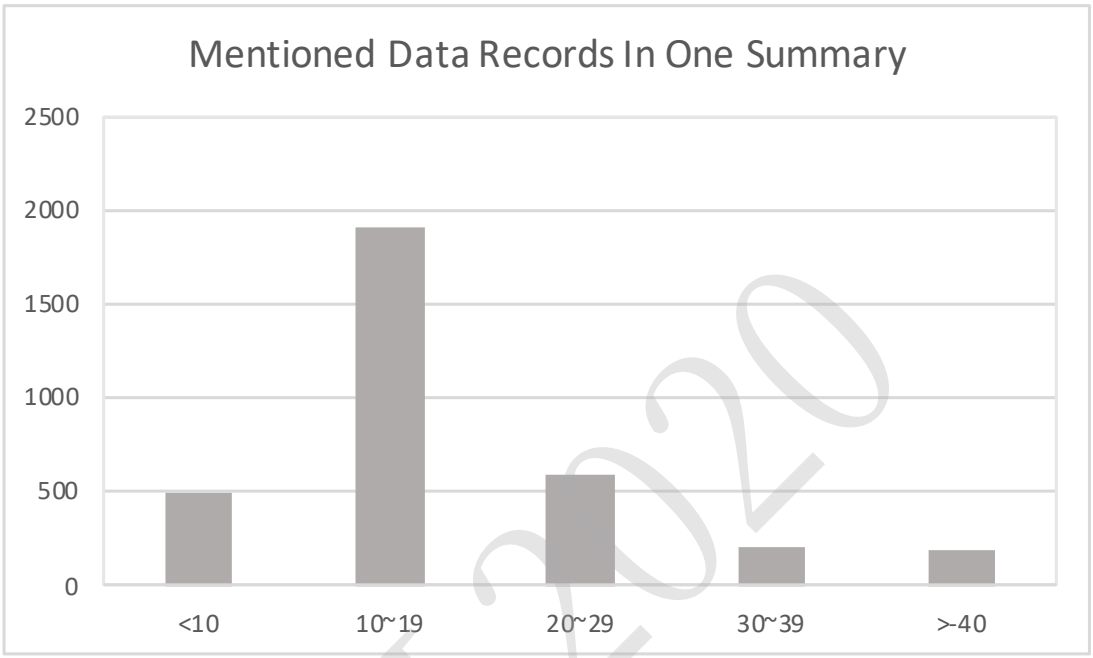


Figure 1: Statistics of data records mentioned in 3000 sports news. The horizontal axis stands for summary numbers and the vertical axis stands for data record numbers.

game, and further to plan an appropriate order for the selected records. This is also similar to the action of human writers who usually plan the important information to include before they write their articles.

Next, one key problem is to label the important records which would be time consuming and expensive. To solve this problem, inspired by Wiseman et al. (2017) which used an information extraction (IE) system for evaluation and Mintz et al. (2009) which used distance learning for relation extraction, we build an IE system based on distant supervision. The IE system extracts relations from gold text, matches them to the corresponding data records and its results can then be used to supervise the process of content selection and planning. Then, we design a ranking unit to learn which data records are selected and in what order they appear. Here we choose to use the learning-to-rank (L2R) method instead of a classifier, because there exists heavy imbalance between positive and negative instances. We also design a rule-based model to further help select the data records. We rank each data record by an overall score based on the two rankers and rule-based system. Finally, we feed the selected and ordered records, which not only the noise and the input size is reduced but also the content is planned, to the generator to obtain the summaries. In this way memory usage could be largely reduced, thus the training process could be accelerated.

We evaluate our method on the ROTOWIRE dataset (Wiseman et al., 2017). The results show how our system improves the model's ability of selecting appropriate context and ordering them. While we achieve comparable BLEU score, the efficiency of the model is greatly improved.

2 Related Work

Data-to-text generation has been an important topic of natural language generation for decades. Early approaches mainly use templates and rules to perform content selection and surface realization (Kukich, 1983; Holmes-Higgin, 1994; Reiter and Dale, 1997). These models have good interpretability and controllability, but the generated content often have problems in terms of diversity and consistency.

Recently, neural network techniques have greatly improved the results of generating short descriptive text from data. The E2E dataset (Lebret et al., 2016) stated the task of generating natural language descriptive text of the restaurants from structured information of the restaurants. The Wikibio dataset (Novikova et al., 2017) gives the infobox of wikipedia as the input data and the first sentence of the corresponding biography as output text. Various approaches have achieved good results on these two datasets which considered content selection and planning. Sha et al. (Sha et al., 2017) proposed a method that models the order of information via link-based attention between different types of data records. Perez-Beltrachini and Lapata (Perez-Beltrachini and Lapata, 2018) introduce a content selection method based on multi-instance learning.

Generating sport news summaries on the other hand,is more challenging because not only the output text is longer and more complex, but also the input data records are numerous and diversed. Wiseman et al. (Wiseman et al., 2017) proposed the ROTOWIRE data set and gave baselines model based on end-to-end neural networks with attention and copy mechanism, these models often overlook key facts, repeatedly output the same information and make up irrelevant content. Puduppully et al. (Puduppully et al., 2018) designed a system that uses gate mechanism and pointer network to select and plan the content. They only used the IE system to guide content planning, while we let the IE system guide both content selecting and planning. Meanwhile our system is lighter and has higher efficiency since we only feed the neural network with a small subset of the large set of data records.

3 Model

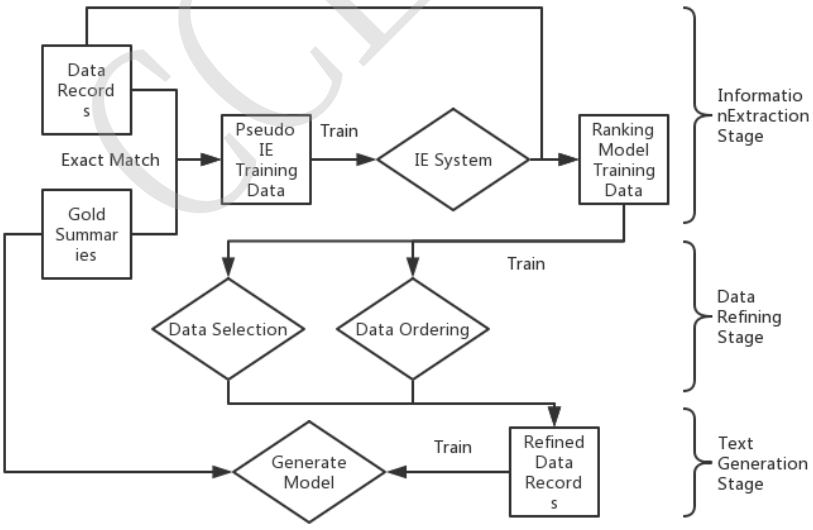


Figure 2: A brief flow graph of our model.

Our model consists of three modules: information extraction, data refining (record selection and planning) and text generation. Figure 2 is a brief flow chart showing the pipeline of our model, which illustrates the data flow and how the models are trained.

3.1 Information Extraction

This module aims to provide supervision for data refining and text generation, and is only used during training. We build a relation extractor similar to Wiseman et al. (2017), who used a relation extractor for automatic evaluation. We do not have human-annotated data for this specific domain, but this relation extractor can be trained by distance learning(Mintz et al., 2009), which uses exact match between candidate entity-value pairs and data records to build pseudo training data. For example, from a sentence *A scored 4 points and B scored 8 points*, which has two entities $\{A, B\}$ and two values $\{4, 8\}$, we can extract 4 candidate entity-value pairs $\{(A, 4), (A, 8), (B, 4), (B, 8)\}$. Then we compare them with the original data records and check whether these candidate pairs match with data records. In this example we can find *(A, 4, PTS)* and *(B, 8, PTS)* in the original data records, so we label the candidate pairs as $\{(A, 4, PTS), (A, 8, norel), (B, 4, norel), (B, 8, PTS)\}$, where *norel* is the label that stands for no relationship and form the pseudo data. To be noticed, there might be multiple data records that match with the candidate pair, so the training data here is multi-labeled. The reason why we use an IE system instead of using the pseudo data straight away is because with the help of context information, the IE system can make better decisions and generalize better than the exact-match method.

To train the IE system, we cast the relation extraction task into a classification problem by modeling whether an entity-value pair in the same sentence has relation or not (Zhang, 2004; dos Santos et al., 2015). We use neural network to train the relation extractor and ensemble various models to further improve the performance. Formally, given an input sentence $\mathbf{x} = \{x_t\}_{t=1}^{n}$ which contains an entity-value candidate pair *(r.E, r.M)*, we first embed each word into a vector e_t^W. The embedding is then concatenated with two position embedding vectors e_t^E and e_t^V, which stands for the distance between the word and the entity and the value. Then the final word embeddings $e_t = concat\{e_t^W, e_t^E, e_t^V\}$ are fed into a bi-directional long short-term memory network (BiLSTM) or a convolutional neural network (CNN) to model the sequential information.

$$\begin{aligned} h_t &= BiLSTM(e_t, h_{t-1}, h_{t+1}) \\ h_{LSTM} &= h_n \end{aligned} \tag{1}$$

$$h_{CNN} = CNN(concat\{e_t\}_{t=1}^{n}) \tag{2}$$

After encoding the sentence, we use multilayer perceptron network (MLP) with a rectified linear unit(ReLU) as active function to make classification decisions and maintain the model's prediction of the candidate pair $r.T$. To be minded, the output $r.T$ is a vector where each position indicates whether the candidate pair is aligned with the data record at this position. Since there could be multiple labels, the output vectors are not distributions.

$$r.T = ReLU(Wh + b) \tag{3}$$

Because the training data is multi-labeled, we use negative marginal log likelihood as the loss function, namely each position is optimized toward 1 if positive and 0 if negative. We then map the positive candidate pairs back to the data records as silver training labels for the next stage. If a positive candidate pair *(entity, value, $r.T$)*, which is extracted from the xth sentence, is also in the data records, we label this data record as *Appeared in the xth sentence of the summary*.

3.2 Data Refining

In this module, we use two ranking models to refine the data records. These two rankers have different targets to optimize and separately perform content selection and ordering.

For content selection, we use both ListNet(Cao et al., 2007) and rule-based methods to select data records. The training data of this stage is seriously imbalanced: more than 95% of the input data records do not appear in the summaries and are labeled as negative. This makes it difficult for classification models to achieve good results. So here we use the L2R method to perform content selection. Instead of a point-wise loss function, which looks at a single example at a time, pair-wise and list-wise loss functions try to come up with the optimal ordering of a pair or a list of examples. In this stage we use

Feature	Type	Explanation
Record Type	One hot	The one-hot representation of record type (i.e. PTS)
Is Team	Value	Boolean of team or player
Home Visit	Value	Boolean of home or visit team
Win Lose	Value	Boolean of win or loss
Win ratio	Value	The win ratio of previous matches
Lose ratio	Value	The lose ratio of previous matches
Team Performance	Values	All values of the team (i.e. PTS, PTS_QTR1, FG_PCT)
Player Performance	Values	All values of the player. Zeros if it is team record
Start Position	One hot	The start position of player. Zeros if it is team record
Pair Value	Value	The value of **f**, if not a number then 0
N/A	Value	Whether the value is N/A
Team Rank	Values	Whether the team value is larger that the other
Player Rank	Values	The rank of each record type of this player

Table 2: The details of features used for the ranking unit.

Type	Rule	Threshold
TEAM-PTS	all	\
TEAM-WINS	all	\
TEAM-LOSSES	all	\
AST	bar	9
PTS	bar	11
REB	bar	9
TEAM-FG3_PCT	bar	45
TEAM-FG_PCT	bar	10

Table 3: The details of rules for the ranking unit. 'all' stands for choosing all records of this type of data. 'bar' stands for choosing the data records which value is larger than the threshold.

ListNet, which optimizes a list-wise loss function, so the data imbalance problem can be relieved. Given a list of data records $\mathbf{r} = \{r_k\}_{k=1}^{n} = \{r.E_k, r.M_k, r.T_k\}_{k=1}^{n}$, we design hand-craft features and form a feature vector f_k for each data record as the input of the ranking model. We give the details of the features in the Table 2. Then the ranking model assigns a score s_k^S to each data record.

$$s_k^S = ListNet(f_k) \tag{4}$$

During inference stafe, we use a hyper-parameter threshold α tuned on the validation set to choose data records.

The rules are designed based on common sense and statistics of basketball news. We observe that several types of data records are chosen mainly according to whether the data record's value is larger than a specific threshold. Some other type of data records always appear in pairs, such as FTA and FTM. We give a table of details of the rules in the Table 3.

For content ordering, we use a pair-wise L2R method RankBoost(Freund et al., 2003) to reorder the selected data records. While training, we use the subset of data records $\mathbf{r} = \{r_k | r_k.t \neq negative\}$ to train this model. When we perform inference, the output of the content selecting unit is used as the input. We similarly embed r_k into a feature vector f_k and then use RankBoost to assign a score s_k^O to each r_k.

$$s_k^O = RankBoost(f_k) \tag{5}$$

We use s_k^O to reorder $\{r\}$ into $\{r^O\}$ and feed this ordered list of data records to the text generation module.

3.3 Text Generation

In the text generation module, we use a sequence-to-sequence encoder-decoder system to generate the summaries (Sutskever et al., 2014). Given a list of data records $r^O = [r_k^O]_{k=1}^n$. We map these data records to a feature vector e_k by embedding $r.E$, $r.T$ and $r.M$ and concatenate the three embedding vectors and then use one layer of MLP to merge them into the final embedding vector.

The embeddings are then fed into the encoder, which is a BiLSTM to sequentially model the input and maintain the encoder output vectors hidden states h_t.

$$h_t = [h_t^f; h_t^b] = BiLSTM(e_t, h_{t-1}^f, h_{t+1}^b) \tag{6}$$

The decoder is built based on the Gated Recurrent Network (GRU). At each time step the decoder receives an input e_t^d and calculates the output vector s_t^d. Meanwhile it updates its own hidden state h_t^d.

$$s_t^d, h_t^d = GRU(e_t^d, h_{t-1}^d) \tag{7}$$

Here we implement the attention mechanism, conditional copy mechanism and coverage mechanism to further improve the model's performance.

Attention and Coverage The attention at each step is calculated similar to See et al.(See et al., 2017), which is called perception attention. To calculate the attention weight between the hidden state of the decoder h_t^d and one output of the encoder h_i, we map the two vectors to fix size vectors seperately by two MLPs W_a and U_a with trainable bias b_a as h_{a_i}. Then we use a trainable vector v_a and dot multiply it with $tanh(h_{a_i})$ as the attention score s_{t_i}. At last we calculate the softmax over attention scores $\{s_{t_i}\}_{i=0}^n$ as the attention weights $\{a_{t_i}\}_{i=0}^n$. We finally dot-multiply the attention weights $\{a_{t_i}\}_{i=0}^n$ with the encoder outputs $\{h_i\}_{i=0}^n$ and sum them as the final attention vector $h_{t_{attn}}$.

$$h_{a_i} = W_a h_t^d + U_a h_i + b_a \tag{8}$$

$$s_{t_i} = v_a^T tanh(h_{a_i}) \tag{9}$$

$$a_{t_i} = softmax(s_{t_i}) = \frac{exp(s_{t_i})}{\sum_j exp(s_{t_j})} \tag{10}$$

$$h_{t_{attn}} = \sum_{i=0}^{n} a_{t_i} h_i \tag{11}$$

We also found that model often tends to repeatedly write about the same information, so we introduce coverage mechanism here to relief this problem. The key idea of coverage is to reduce the probability of paying attention to the information that is already generated.

If the sum of the previous attention weights is very high, there is a high probability that the information of this position is already generated. So in coverage model, we maintain a coverage score c_{t_i} for each encoder position at each decoder timestep, which is the sum of the attention weight of the previous timesteps $\{a_{t'_i}\}_{t'=0}^{t-1}$.

$$c_{t_i} = \sum_{t'=0}^{t-1} a_{t'_i} \tag{12}$$

We then modify the previous attention score with this coverage score. We assign a trainable weight vector w_c to c_{t_i} and sum it with h_{a_i} to maintain the adapted attention score.

$$s_{t_i} = v_a^T tanh(h_{a_i} + w_c c_{t_i}) \tag{13}$$

Conditional Copy The copy mechanism has shown great effectiveness as an augmentation of encoder-decoder models recently. At each step the model uses an additional variable z_t to choose to copy or generate a word. The model either copies a word from the input sequence or generates a word from the vocabulary at step t.

Although both $r_k.E$ and $r_k.M$ may appear in the summaries, we only consider the probability of copying $r_k.M$. Instead of directly marginalizing out the latent-variable z_t, when we train the model we assume that any word y_t that appears both in the source data records and the summary is copied, so that we can jointly optimize the negative log-likelihood of y_t and z_t. To be noticed, there might be not only one $r_k.M$ that matches with y_t. Because our input data shares the same sequential order with the information mentioned in the summaries, we map the values from the start of the data records and skip the ones that are already mapped to align the records and copied values.

$$y = \begin{cases} p_{copy}(y_t|z_t; y_{1:t-1}; h_{1:n})p(z_t|y_{1:t-1}; h_{1:n}); \\ \qquad\qquad z_t = 1 \\ p_{generate}(y_t|z_t; y_{1:t-1}; h_{1:n})p(z_t|y_{1:t-1}; h_{1:n}); \\ \qquad\qquad z_t = 0 \end{cases}$$

We use the attention weights explained previously as the distribution $p_{copy}(y_t|z_t; y_{1:t-1}; h_{1:n})$. We concatenate the decoder input e_t^d, the decoder output s_t^d and the attention vector $h_{t_{attn}}$ and feed them into one MLP layer with sigmoid to model $p(z_t|y_{1:t-1}; h_{1:n})$.

4 Experiments and Results

4.1 Dataset

Here we use the ROTOWIRE dataset(Wiseman et al., 2017), which contains 3378 data-text pair in the training data. In addition to BLEU, this data set provides three automatic evaluation metrics, which are content selection (CS), relation generation (RG), and content ordering (CO). The first primarily targets "what to say" while the latter two metrics target "how to say". These three metrics are calculated based on an information extraction system that serves to align entity-mention pairs in the text with data records. We use the code released by Wiseman et al. (2017) to maintain the evaluation scores of our model.

4.2 Implementation Details

We tune all the hyper-parameters according to the model performance on the validation set. The rules of the ranking unit are chosen according to their performance on the training set. We use grid search to tune parameters of the rankers. We use the implementation of RankLib to train the rankers. The embedding size, hidden size of both the encoder and decoder are all 1200. The layer number of the encoder and decoder are both two. The batch size is 12. We set dropout of 0.1 and use Adagrad to optimize the text generator with a learning rate of 0.01.

For the ranking units and text generator, we use the data records that the IE system extracts directly to reduce noise while training. During validation and test, we use the ranking units to extract the input of the generator.

We re-tokenize the original training data by separating numbers connected by '-' and ':'. We also delete one Latin summary from the training data.

4.3 Performance

The results of our model and other baseline systems are shown in Table 4.

From the results we can see the effectiveness of our model, since it has significantly improved all the content evaluation metrics. Thus we can say refining the input data can help the model to be faithful to the input (RG), select good content (CS) and order them considering overall coherence (CO). We can see the BLEU score of our model is slightly lower than the baseline models. We think this is acceptable in trade of the great improvement of other evaluation scores.

Model	RG		CS		CO	BLEU
	P	#	P	R	DLD	
Validation Set						
Template(Wiseman et al., 2017)	**99.35**	**49.7**	18.28	**65.52**	12.2	6.87
CC(Wiseman et al., 2017)	71.07	12.61	21.90	27.27	8.70	**14.46**
JC + TVD + Rec(Wiseman et al., 2017)	57.51	11.41	18.28	25.27	8.05	12.04
CC + R	76.86	16.43	**31.20**	38.94	**14.98**	13.27
Test Set						
Template(Wiseman et al., 2017)	**99.30**	**49.61**	18.50	**64.70**	8.04	6.78
CC (Wiseman et al., 2017)	71.82	12.61	21.90	27.16	8.68	**14.49**
JC + TVD +Rec (Wiseman et al., 2017)	60.27	9.18	23.11	23.69	8.48	12.96
CC + R	75.12	16.90	**32.79**	39.93	**15.62**	13.46

Table 4: The results of text generation on validation set and test set. CC stands for conditional copy, JC stands for joint copy, TVD stands for the total variation distance loss, Rec stands for reconstruction losses, R stands for ranking.

5 Analysis

5.1 Content Selection

The results of our content selection and ordering models on the valid set are shown in Table 5. The results can prove our models' ability of refining data. We can see, because of imbalanced training data, ranking models with a threshold can significantly out perform classification models.

Model	P	R	F1
ListNet	18.08	26.93	21.63
SVM	10.76	21.27	14.29
Random Forest	9.63	55.36	16.41
ListNet + Rule	59.02	59.98	59.50

Table 5: The results of content selection and data ordering on the valid set.

#	select	emb	hid	bs	GPU	time
1	False	600	600	2	9275	214
2	True	600	600	2	2163	45
3	True	600	600	16	10375	8
4	True	1200	1200	12	10525	16

Table 6: The results of original input and refined order input. 'emb' and 'hid' stands for embedding and hidden dimensions. 'bs' stands for batch size. 'GPU' stands for the maximum memory used on GPU. 'time' stands for the time used for every epoch, the unit is minute.

5.2 Model Efficiency

Our model also significantly improves the efficiency of the model. We show the comparison of our model and CC(Wiseman et al., 2017) model in Table 6 and Figure 3. Our model significantly reaches convergence faster and uses less memory and time to train. The parameter in the embedding and encoder layer is greatly reduced due to the refining of the input. For case #1 and #2, we can see the GPU memory usage and the time for each epoch is greatly reduced, which leads to faster convergence of the model. In case #3 and #4, we show that by refining the input, we can allow larger batch size, embedding size and

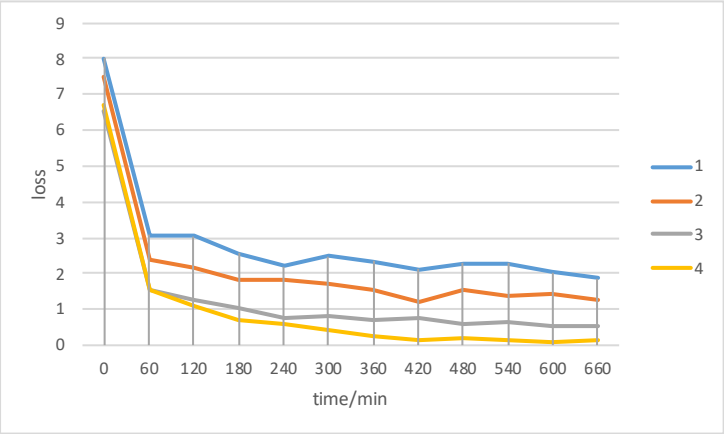


Figure 3: Statistics of how the loss changes over time. The number labels of the poly-lines match with the order in Table 4.

hidden state size for the model to further boost the performance. While the architecture of the generator of our model and CC is similar, we show refining the input can greatly improve the model's efficiency.

5.3 Case Study

Here we show one example of the pipeline on the validation set in Figure 4. We show the triples extracted by the IE system, triples extracted by the refining unit the gold text and the final generated text.

From this example we can see, the IE system has a strong ability of extracting relation pairs from the gold text. The IE system missed two information pairs which are (Pacers,35,TEAM-TEAM-PTS_QTR3) and (Knicks,12,TEAM-TEAM-PTS_QTR3), but succeeded in all other pairs, ending with an accuracy of 87.5% in this example.

The refining system shows a high precision comparing to the gold reference, covering 12 out of 16 triples.

The generated text is very faithful to the refined input at the first 5 sentences, but began making up false information when it tries to generate facts not given by the refined input. 2 - 3 3Pt , 3 - 3 FT are fake information about Jose Calderon where the corresponding information is not selected by the refining system. The following text contains more fake information. This shows the limitations in generating long text for seq2seq models and some shortages of pre-selected refined text. For further improvement, we should improve the ability of the model to generate long text, and also consider dynamically giving information that the model needs instead of feeding fixed triples.

6 Conclusion

In this paper we propose a data-to-text generating model which can learn data selecting and ordering from an IE system. Different from previous methods, our model learns what to say and how to say from the supervision of an IE system. To achieve our goal, we propose to use a ranking unit to learn selecting and ordering content from the IE system and refine the input of the text generator. Experiments on the ROTOWIRE dataset verifies the effectiveness of our proposed method.

Acknowledgement

We thank the anonymous reviewers for their helpful comments on this paper. This work was partially supported by National Key Research and Development Project (2019YFB1704002) and National Natural Science Foundation of China (61876009 and 61572049). The corresponding author of this paper is Sujian Li.

IE (New York Knicks, 82, TEAM-PTS), (New York Knicks, 9, TEAM-WINS), (Indiana Pacers, 31, TEAM-LOSSES), (Indiana Pacers, 17, TEAM-WINS), (Indiana Pacers, 103, TEAM-PTS), (New York Knicks, 38, TEAM-LOSSES), (Roy Hibbert, 18, PLAYER-PTS), (Roy Hibbert, 10, PLAYER-REB), (Carmelo Anthony, 7, PLAYER-FGM), (Carmelo Anthony, 16, PLAYER-FGA), (Carmelo Anthony, 18, PLAYER-PTS), (Rodney Stuckey, 22, PLAYER-PTS), (Rodney Stuckey, 13, PLAYER-FGA), (Rodney Stuckey, 8, PLAYER-FG)
Refine (Knicks, 38, TEAM-LOSSES), (Pacers, 31, TEAM-LOSSES), (Knicks, 9, TEAM-WINS), (Pacers, 17, TEAM-WINS), (Knicks, 42, TEAM-FG_PCT), (Pacers, 53, TEAM-FG_PCT), (Pacers, 33, TEAM-FG3_PCT), (Knicks, 31, TEAM-FG3_PCT), (Knicks, 82, TEAM-PTS), (Pacers, 103, TEAM-PTS), (Carmelo Anthony, 18, PLAYER-PTS), (Carmelo Anthony, 16, PLAYER-FGA), (Carmelo Anthony, 7, PLAYER-FGM), (Carmelo Anthony, 2, PLAYER-FG3M), (Carmelo Anthony, 4, PLAYER-FG3A), (Ian Mahinmi, 10, PLAYER-REB), (Carmelo Anthony, 25, PLAYER-MIN), (Lou Amundson, 17, PLAYER-PTS), (Rodney Stuckey, 22, PLAYER-PTS), (Jose Calderon, 9, PLAYER-PTS), (Jose Calderon, 28, PLAYER-MIN), (Jose Calderon, 4, PLAYER-FGM), (Jose Calderon, 7, PLAYER-FGA), (Carmelo Anthony, 2, PLAYER-FTM), (Carmelo Anthony, 2, PLAYER-FTA), (Carmelo Anthony, 1, PLAYER-REB), (Carmelo Anthony, 1, PLAYER-AST), (Roy Hibbert, 10, PLAYER-REB), (Roy Hibbert, 18, PLAYER-PT)
Gold The Indiana Pacers (17 - 31) throttled the New York Knicks (9 - 38) 103 - 82 Thursday at the Bankers Life Fieldhouse . The Knicks held a five - point deficit at halftime before having a complete meltdown in the third quarter . The Pacers outscored the Knicks 33 - 12 in the third to all but ice the game . Roy Hibbert scored 10 of his 18 points in the third and added 10 rebounds in the contest to notch his 12th double - double on the season . New York earned an impressive 100 - 92 win over Oklahoma City on Wednesday but was unable to carry that momentum into Thursday 's contest against Indiana . The Knicks were unable to get much of anything going in the second half . Carmelo Anthony had n't scored less than 20 points in any game this month , but was held to 18 points on 7 - of - 16 shooting against the Pacers . The Knicks have two days off before hosting the Lakers on Sunday . Indiana was able to rebound nicely after its 104 - 91 loss to Toronto on Tuesday . George Hill and C.J Miles were both able to play , but were on minutes restrictions and only saw 12 and 13 minutes respectively . Rodney Stuckey led the team in scoring , putting up 22 points on 8 - of - 13 shooting . The Pacers will look to secure their second straight win Saturday when they host the Kings .
Generated The Pacers (17 - 31) held on for a 103 - 82 win over the Knicks (9 - 38) on Friday . Carmelo Anthony led the way with a game - high 18 points on 7 - of - 16 shooting , to go along with one rebound , one assist and one block in 25 minutes . Jose Calderon was the only other starter to score in double figures , scoring 10 points on 4 - of - 7 shooting in 28 minutes off the bench . The Knicks shot 53 percent from the field and 33 percent from three - point range . The Knicks were led by Carmelo Anthony , who finished with 18 points (7 - 16 FG , 2 - 4 3Pt , 2 - 2 FT) , one rebound , one assist and one block in 25 minutes . Jose Calderon added 10 points (4 - 7 FG , 2 - 3 3Pt , 3 - 3 FT) , six assists , one rebound and one steal in 33 minutes . The Knicks shot 53 percent from the field and 33 percent from three - point range . Carmelo Anthony was the high - point man for the Knicks , with 18 points (7 - 16 FG , 2 - 4 3Pt , 2 - 2 FT) , to go with one rebound , one assist , one steal and one block . Jose Calderon chipped in 10 points (4 - 7 FG , 2 - 3 3Pt , 3 - 3 FT) , six assists , one rebound and one steal in 33 minutes . The Knicks shot 53 percent from the field and 31 percent from three - point range . The Knicks have now lost four of their last five games , as they continue to battle for the eighth seed in the Eastern Conference . They will look to get back on track as they play host to the Bucks on Sunday . The Pacers have now lost four of their last six games , as they continue to battle for a playoff spot in the Eastern Conference . They will look to bounce back as they play host to the Nets on Sunday .

Figure 4: One example of our system. IE stands for triples extracted by the IE system, Refine stands for triples selected by the refining system, Gold stands for the reference text and Generated stands for the generated result.

References

Antoine Bosselut, Asli Celikyilmaz, Xiaodong He, Jianfeng Gao, Po-Sen Huang, and Yejin Choi. 2018. Discourse-aware neural rewards for coherent text generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 173–184.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136. ACM.

Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2015. Classifying relations by ranking with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 626–634.

Yoav Freund, Raj Iyer, Robert E Schapire, and Yoram Singer. 2003. An efficient boosting algorithm for combining preferences. *Journal of machine learning research*, 4(Nov):933–969.

Paul Holmes-Higgin. 1994. Text generation—using discourse strategies and focus constraints to generate natural language text by kathleen r. mckeown, cambridge university press, 1992, pp 246,£ 13.95, isbn 0-521-43802-0. *The Knowledge Engineering Review*, 9(4):421–422.

Karen Kukich. 1983. Design of a knowledge-based report generator. In *21st Annual Meeting of the Association for Computational Linguistics*.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1203–1213.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011. Association for Computational Linguistics.

Jekaterina Novikova, Ondrej Dušek, and Verena Rieser. 2017. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, Saarbrücken, Germany. arXiv:1706.09254.

Laura Perez-Beltrachini and Mirella Lapata. 2018. Bootstrapping generators from noisy data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 1516–1527.

Ratish Puduppully, Li Dong, and Mirella Lapata. 2018. Data-to-text generation with content selection and planning. *arXiv preprint arXiv:1809.00582*.

Ehud Reiter and Robert Dale. 1997. Building applied natural language generation systems. *Natural Language Engineering*, 3(1):57–87.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1073–1083.

Lei Sha, Lili Mou, Tianyu Liu, Pascal Poupart, Sujian Li, Baobao Chang, and Zhifang Sui. 2017. Order-planning neural text generation from structured data. *arXiv preprint arXiv:1709.00155*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2253–2263.

Zhu Zhang. 2004. Weakly-supervised relation classification for information extraction. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 581–588. ACM.

Plan-CVAE: A Planning-based Conditional Variational Autoencoder for Story Generation

Lin Wang[1,2], **Juntao Li**[1,2], **Dongyan Zhao**[1,2], **Rui Yan**[1,2*]
[1]Center for Data Science, Academy for Advanced Interdisciplinary Studies,
Peking University, Beijing, China
[2]Wangxuan Institute of Computer Technology, Peking University, Beijing, China
{wanglin,lijuntao,zhaody,ruiyan}@pku.edu.cn

Abstract

Story generation is a challenging task of automatically creating natural languages to describe a sequence of events, which requires outputting text with not only a consistent topic but also novel wordings. Although many approaches have been proposed and obvious progress has been made on this task, there is still a large room for improvement, especially for improving thematic consistency and wording diversity. To mitigate the gap between generated stories and those written by human writers, in this paper, we propose a planning-based conditional variational autoencoder, namely Plan-CVAE, which first plans a keyword sequence and then generates a story based on the keyword sequence. In our method, the keywords planning strategy is used to improve thematic consistency while the CVAE module allows enhancing wording diversity. Experimental results on a benchmark dataset confirm that our proposed method can generate stories with both thematic consistency and wording novelty, and outperforms state-of-the-art methods on both automatic metrics and human evaluations.

1 Introduction

A narrative story is a sequence of sentences or words which describe a logically linked set of events (Mostafazadeh et al., 2016). Automatic story generation is a challenging task since it requires generating texts which satisfy not only thematic consistency but also wording diversity. Despite that considerable efforts have been made in the past decades, the requirement of thematic consistency and wording diversity is still one of the main problems in the task of story generation.

On the one hand, a well-composed story is supposed to contain sentences that are tightly connected with a given theme. To address this problem, most previous methods attempt to learn mid-level representations, such as events (Martin et al., 2018), prompts (Fan et al., 2018), keywords (Yao et al., 2019) or actions (Fan et al., 2019), to guide the sentences generation. Although these approaches have shown their encouraging effectiveness in improving the thematic consistency, most of them have no guarantee for the wording diversity. The main reason is that most of these methods are based on recurrent neural networks (RNNs), which tend to be entrapped within local word co-occurrences and cannot explicitly model holistic properties of sentences such as topic (Bowman et al., 2016; Li et al., 2018; Li et al., 2019). As a result, RNN tends to generate common words that appear frequently (Zhao et al., 2017) and this will lead to both high inter- and intra-story content repetition rates.

On the other hand, a well-composed story also needs to contain vivid and diversified words. To address the issue of wording diversity, some studies have employed models based on variational autoencoder (VAE) (Kingma and Welling, 2013) or conditional variational autoencoder (CVAE) as a possible solution. It has been proved that, through learning distributed latent representation of the entire sentences, VAE can capture global features such as topics and high-level syntactic properties, and thus can generate novel word sequences by preventing entrapping into local word co-occurrences (Bowman et al., 2016). As a modification of VAE, CVAE introduces an extra condition to supervise the generating process and

*Corresponding author: Rui Yan (ruiyan@pku.edu.cn).

has been used in multiple text generation tasks, e.g., dialogue response generation (Zhao et al., 2017), Chinese poetry generation (Yang et al., 2018). Recent researches (Li et al., 2019; Wang and Wan, 2019) in story generation task have confirmed that CVAE can generate stories with novel wordings. Despite the promising progress, how to keep thematic consistency while improving wording diversity is still a challenging problem, since these two requirements are to some extent mutually exclusive (Li et al., 2019). Specifically, consistent stories may limit the choice of words, while diversified wordings may lead to the risk of inconsistent themes.

In this paper, we propose to conquer these two challenges simultaneously by leveraging the advantages of mid-level representations learning and the CVAE model in improving wording novelty. Specifically, we propose a planning-based CVAE model, targeting to generate stories with both thematic consistency and wording diversity. Our introduced method can be divided into two stages. In the *planning stage*, keyword extraction and expansion modules are used to generate keywords as sub-topics representations from the title, while in the *generation stage*, a CVAE neural network module is employed to generate stories under the guidance of previously generated keywords. In our method, the planning strategy aims to improve the thematic consistency while the CVAE module is expected to keep the wording diversity of the story. To evaluate our proposed method, we conduct experiments on a benchmark dataset, i.e., the Rocstories corpus (Mostafazadeh et al., 2016). Experimental results demonstrate that our introduced method can generate stories that are more preferable for human annotators in terms of thematic consistency and wording diversity, and meanwhile outperforms state-of-the-art methods on automatic metrics.

2 Related Work

2.1 Neural Story Generation

In recent years, neural network models have been demonstrated effective in natural language processing tasks (Mikolov et al., 2010; Sutskever et al., 2014; Rush et al., 2015; Roemmele et al., 2017; Liu et al., 2020; Yu et al., 2020). In story generation, previous studies have employed neural networks for enhancing the quality of generated content. Jain et al. (2017) explored generating coherent stories from independent short descriptions by using a sequence to sequence (S2S) architecture with a bidirectional RNN encoder and an RNN decoder. Since this model is insufficient for generating stories with consistent themes, to improve the thematic consistency of the generated stories, many other methods have been explored. Martin et al. (2018) argued that using events representations as the guidance for story generation is able to improve the thematic consistency of generated content. Fan et al. (2018) presented a hierarchical method that first generates a prompt from the title, and then a story is generated conditioned on the previously generated prompt. Following the idea of learning mid-level representations, Xu et al. (2018) proposed a skeleton-based model that first extracts skeleton from previous sentences, and then generates new sentences under the guidance of the skeleton. Similarly, Yao et al. (2019) explored using a storyline planning strategy for guiding the story generation process to ensure the output story can describe a consistent topic. Fan et al. (2019) further adopted a structure-based strategy that first generates sequences of predicates and arguments, and then outputs a story by filling placeholder entities. Although these methods have achieved promising results, most of them are implemented with RNNs, which tend to encounter common words problem. In recent researches, the Conditional Variational Auto-Encoder model is regarded as a possible solution for improving the wording diversity in story generation (Li et al., 2019).

2.2 Conditional Variational Autoencoder

The Variational Auto-Encoder (VAE) model is proposed in (Kingma and Welling, 2013). Through forcing the latent variables to follow a prior distribution, VAE is able to generate diverse text successfully by randomly sampling from the latent space (Bowman et al., 2016). Conditional Variational AutoEncoder (CVAE), as a variant of VAE, can generate specific outputs conditioned on a given input. CVAE has been used in many other related text generation tasks, such as machine translation (Zhang et al., 2016), dialogue generation (Serban et al., 2017; Zhao et al., 2017; Shen et al., 2017), and poem composing (Yang et al., 2018; Li et al., 2018). Subsequently, in recent years, CVAE has begun to be applied in

story generation task to tackle the common wording problem. Li et al. (2019) explored adopting CVAE to generate stories with novel and diverse words, and Wang et al. (2019) alter the RNN encoder and decoder of CVAE architecture with the Transformer encoder and decoder (Vaswani et al., 2017) for the story completing task. Although the CVAE model has achieved encouraging performance on improving wording diversity, it is a still challenging problem to generate stories with both thematic consistency and diverse wordings. To solve this problem, in this paper, we propose a Plan-CVAE, which leverages the advantages of CVAE to generate diverse sentences and keeps the thematic consistency of the whole generated stories by using a planning strategy.

3 Preliminary

3.1 VAE and CVAE

A VAE model consists of two parts, an encoder which is responsible for mapping the input x to a latent variable z, and a decoder which works by reconstructing the original input x from the latent variable z. In theory, VAE forces z to follow a prior distribution $p_\theta(z)$, generally a standard Gaussian distribution ($\mu = 0$, $\sigma = 1$). It first learns a posterior distribution of z conditioned on the input x via the encoder network, denoted as $q_\theta(z|x)$, and then applies the decoder network to computes another distribution of x conditioned on z, denoted as $p_\theta(x|z)$, where θ are the parameters of the network.

The training objective of VAE is to maximize the log-likelihood of reconstructing the input x, denoted as $log\, p_\theta(x)$, which involves an intractable marginalization (Kingma and Welling, 2013). To facilitate model parameters learning, VAE can be trained alternatively by maximizing the variational lower bound of the log-likelihood, and the true posterior distribution $q_\theta(z|x)$ is substituted by its variational approximation $q_\phi(z|x)$, where ϕ denotes the parameters of q. The objective can be written as

$$L(\theta, \phi; x) = -KL(q_\phi(z|x)||p_\theta(z)) + E_{q_\phi(z|x)}[log\, p_\theta(x|z)] \tag{1}$$

The objective mentioned above contains two terms, where the first term $KL(\cdot)$ represents the KL-divergence loss, which encourages the model to keep the posterior distribution $q_\phi(z|x)$ close to the prior $p_\theta(z)$. The second term $E[\cdot]$ is the reconstruction loss for guiding the decoder to reconstruct the original input x as much as possible.

CVAE is a modification version of VAE, it introduces an extra condition c to supervise the generative process. Correspondingly, the encoder computes a posterior distribution $q_\theta(z|x, c)$, representing the probability of generating z conditioned both on x and c. Similarly, the distribution computed by decoder is $p_\theta(x|z, c)$, and the prior distribution of z is $p_\theta(z|c)$. Accordingly, the objective of CVAE can be formulated as

$$L(\theta, \phi; x, c) = -KL(q_\phi(z|x, c)||p_\theta(z|c)) + E_{q_\phi(z|x,c)}[log\, p_\theta(x|z, c)] \tag{2}$$

3.2 Problem Formulation

We formulate the story generation task with the following necessary notations:

Input: A title $T = (t_1, t_2, ..., t_n)$ is given to the model to guide the story generation, where t_i refers the i-th word and n denotes the length of the given title.

Output: A story $S = \{S_1, S_2, ..., S_m\}$ should be generated by the model based on the given title, where S_i represents the i-th sentence and m denotes the total number of sentences in the story.

Keywords: A keywords sequence $K = (k_1, k_2, ..., k_m)$ is generated from title to enhance the process of story generation, where k_i is the i-th keyword which serves as the sub-topic or extra hint for S_i.

4 Planning-based CVAE Method

4.1 Overview

The overview of our proposed method is shown in Figure 1. Our method contains two stages: a planning stage and a generation stage. In the planning stage, a *Keywords-Extraction module* followed by a *Keywords-Expansion module* are used. In this stage, several keywords are first extracted from the title,

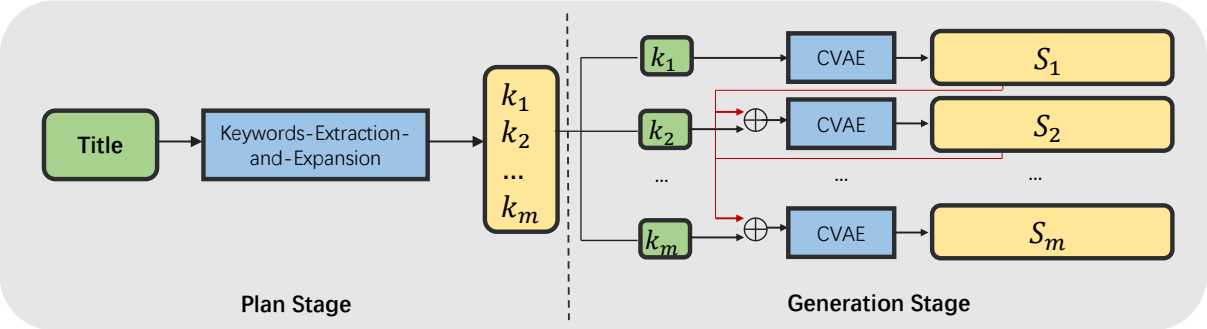


Figure 1: An overview of our proposed method.

and then the extracted keywords are expanded to match the number of sentences to be generated. In the generation stage, a *CVAE module* generates the story sentence-by-sentence conditioned on the keywords, i.e., keyword k_i is used as the sub-topic or hint of sentence S_i.

4.2 Planning Stage

In the planning stage, we first utilize RAKE algorithm (Rose et al., 2010) to extract keywords from the title. Since each sentence is to be generated under the guidance of a keyword, when the number of extracted keywords is not enough, we need to expand more keywords from existing ones. We adopt a language model with a long short-term memory network (LSTM) to predict the subsequent keywords based on the previously generated keywords.

To train the model, we collect training data from the story corpus. Specifically, for each story that contains m sentences in the corpus, we use RAKE to extract one keyword from one sentence. Then a keyword sequence $(k_1, k_2, ...k_m)$ corresponding to a story forms a sample in the training data. The language model is trained to maximize the log-likelihood of the subsequent keyword:

$$L(\theta) = log\, p_\theta(k_i|k_{1:i-1}) \tag{3}$$

where θ refers to the parameters of the language model, and $k_{1:i-1}$ denotes the preceding keywords.

Additionally, keywords can be directly generated by an RNN model from the title. Different from the straight-forward method, our method first extracts keywords from the title and then expands keywords to a sufficient number. Intuitively, the keywords extracted from the title possess a better consistency with the title. Thus, compared to the direct method, our method can lead to a better thematic consistency. To prove the superiority of our method, an ablation study is conducted to compare our method with the directed method, where the results are given in Table 2.

4.3 Generation Stage

We adopt the CVAE model for the generation stage. As demonstrated in Figure 2, the CVAE model contains an encoder and a decoder. The encoder is implemented with a bidirectional GRU network to encode both the sentences and the keywords with shared parameters. At each step, the current sentence S_i, preceding sentences $S_{1:i-1}$ (denoted as ctx) and the keyword k_i are encoded as the concatenation of the forward and backward hidden states of the GRU, i.e. $h_i = [\overrightarrow{h_i}, \overleftarrow{h_i}]$[0], $h_{ctx} = [\overrightarrow{h_{ctx}}, \overleftarrow{h_{ctx}}]$, $h_k = [\overrightarrow{h_k}, \overleftarrow{h_k}]$, respectively. h_i corresponds to x in Equation 2, and $[h_{ctx}, h_k]$ corresponds to c in Equation 2.

Following previous work (Kingma and Welling, 2013; Zhao et al., 2017; Li et al., 2019), we hypothesize that the approximated variational posterior follows an isotropic multivariate Gaussian distribution, i.e. $q_\phi(z|x, c) \sim \mathcal{N}(\mu, \sigma I)$, where I is the diagonal covariance. Thus modeling the approximated variational posterior is equal to learning μ and σ. As shown in Figure 2, a recognition network is used to learn μ and σ. Specifically, we have

$$\begin{bmatrix} \mu \\ log\, \sigma \end{bmatrix} = W_r \begin{bmatrix} x \\ c \end{bmatrix} + b_r \tag{4}$$

[0] $\rightarrow$ denotes forward and $\leftarrow$ denotes backward

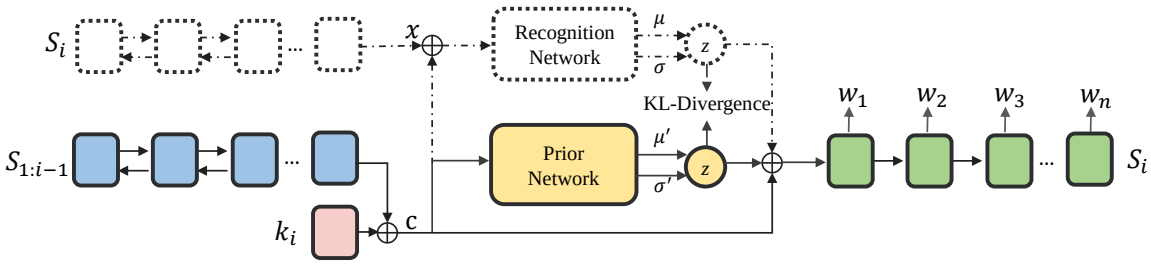


Figure 2: The architecture of the CVAE module used in the generation stage. All components are used for training, while only the components with solid lines are for testing. $\oplus$ denotes the vector concatenation operation.

where W_r and b_r are trainable parameters. Similarly, the prior is assumed to follow another multivariate Gaussian distribution, i.e. $p_\theta(z|c) \sim \mathcal{N}(\mu', \sigma' I)$, and μ' and σ' are learned by the prior network in Figure 2, which is a one-layer fully-connected network (denoted as MLP) with $tanh(\cdot)$ as the activation function. Formally, it can be written as

$$\begin{bmatrix} \mu' \\ log\, \sigma' \end{bmatrix} = MLP_p(c) \tag{5}$$

The decoder is a one-layer GRU. The initial state of the decoder is computed as

$$S_{i,0} = W_d \left[z, c\right] + b_d \tag{6}$$

where W_d is a matrix for dimensional transformation, z is sampled from the recognition network during training and the prior network during testing. Meanwhile, a reparametrization trick (Kingma and Welling, 2013) is used to sample z.

Moreover, previous researches proved that CVAE intends to encounter the latent variable vanishing problem in training (Bowman et al., 2016). Thus, in our implementation, KL cost annealing (Bowman et al., 2016) and bag-of-word loss (Zhao et al., 2017) are used to tackle the problem.

5 Experiments

5.1 Dataset

We conduct experiments on the ROCStories corpus (Mostafazadeh et al., 2016), which contains 98159 stories. In our experiments, the corpus is randomly split into training, validation, and test datasets with 78527, 9816, 9816 stories. Every story in the dataset is comprised of one title and exactly five sentences, and the average word number of one story is 50.

5.2 Baselines

We utilize several strong and highly related methods of story generation as our baselines.

S2S, the sequence to sequence model (Sutskever et al., 2014) which has been widely used in multiple text generation tasks, such as machine translation and summarization. We implement it to generate stories in a sentence-by-sentence fashion, and the i-th sentence is generated by taking all the previous $i-1$ sentences as input.

AS2S, the sequence to sequence model enhanced by an attention mechanism (Bahdanau et al., 2015), which is an improved version of S2S. It takes the same generation pipeline as S2S.

CVAE, the CVAE model without planning strategy. This pure CVAE model takes only the previous $i-1$ sentences as the condition c to generate the i-th sentence. This baseline is for demonstrating the performance of CVAE without planning strategy.

Plan-and-Write, the AS2S model with planning strategy proposed in (Yao et al., 2019). Two different schema (static and dynamic) for keywords generation are proposed in the original paper. As the authors have proved that the static one is better, we implement the static scheme as our baseline.

Table 1: Descriptions of human evaluation metrics.

Readability	Is the story formed with correct grammar?
Consistency	Does the story describe a consistent theme?
Creativity	Is the story narrated with diversified wordings?

Table 2: Results of BLUE and Distinct scores. B-n and D-n represent the BLUE scores and Distinct scores on n-grams respectively. The final results are scaled to [0, 100]. The difference between Plan-CVAE* and Plan-CVAE is the former generates keywords directly from the title, while the latter generates keywords using our keywords-extraction and keyword-expansion module.

Model	**Automatic Evaluation**							
	B-1	**B-2**	**B-3**	**B-4**	**D-1**	**D-2**	**D-3**	**D-4**
S2S	23.65	9.30	4.07	1.97	0.90	4.11	10.70	19.37
AS2S	24.70	9.68	4.27	2.07	0.93	4.53	11.13	19.41
CVAE	28.53	10.21	3.63	1.39	1.67	15.82	46.88	**76.64**
Plan-and-Write	27.39	11.78	**5.57**	**2.85**	0.84	5.15	14.67	28.28
Plan-CVAE*	29.57	11.32	4.43	1.85	1.52	14.13	42.30	71.42
Plan-CVAE	**30.25**	**12.05**	4.89	2.03	**1.75**	**16.38**	**46.98**	75.73
Human	-	-	-	-	2.87	26.74	62.92	86.67

5.3 Model Settings

We train our model with the following parameters and hyper-parameters. The word embedding size is set to 300, and the vocabulary is limited to the most frequent 30000 words. The hidden state size of encoder, decoder, and prior network are 500, 500, 600 respectively. And the size of the latent variable z is set to 300. To train our model, we adopt the Adam (Kingma and Ba, 2015) optimization algorithm with an initial learning rate of 0.001 and gradient clipping of 5. All initial weights are sampled from a uniform distribution $[-0.08, 0.08]$. The batch size is 80.

5.4 Evaluation

We utilize both automatic and human metrics to evaluate the performance of our method.

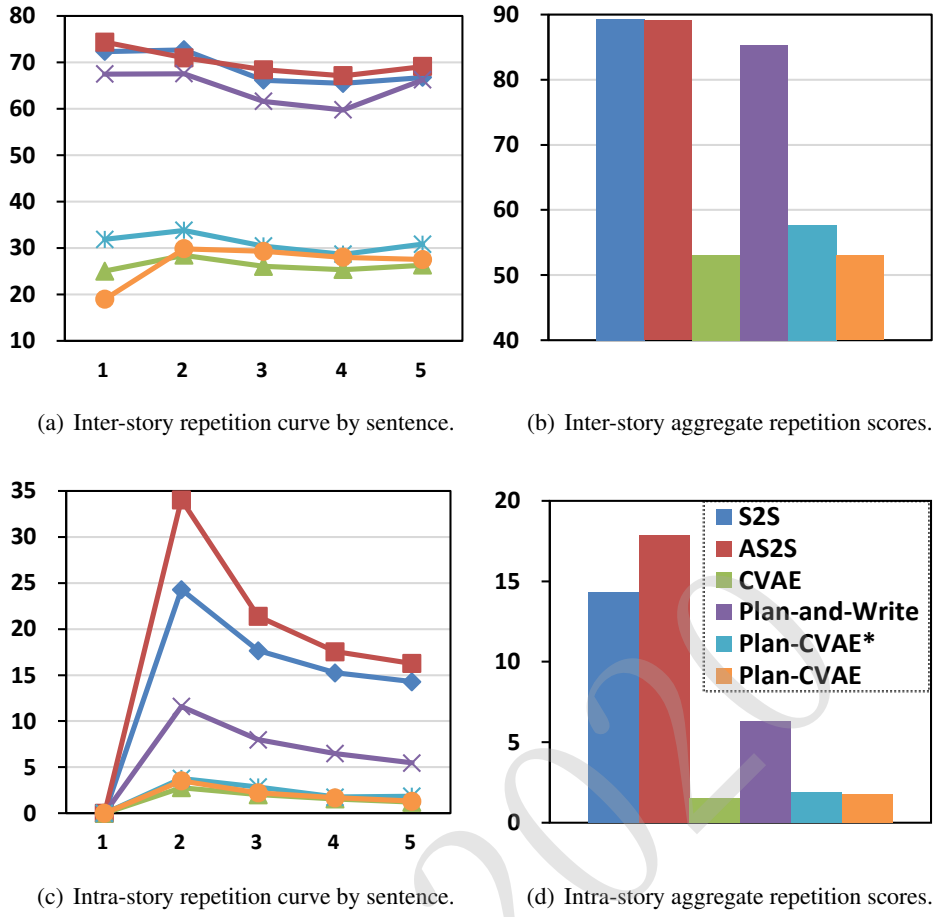
BLUE Score. This metric is designed for calculating the word-overlap score between the golden texts and the generated ones (Papineni et al., 2002), and has been used in many previous story generation works (Yao et al., 2019; Li et al., 2019).

Distinct Score. To measure the diversity of the generated stories, we employ this metric to compute the proportion of distinct n-grams in the generated outputs (Li et al., 2016). Note that the final distinct scores are scaled to [0, 100].

Inter- and intra-story repetition. These two metrics are proposed in (Yao et al., 2019) and used for calculating the inter- and intra-story tri-grams [1] repetition rates by sentences and for the whole stories. The final results are also scaled to [0, 100].

Human Evaluation. We also employ three metrics for human evaluation, i.e., Readability, Consistency, and Creativity. Their descriptions are shown in Table 1. We randomly sample 100 generated stories from each baseline model and our method and then perform pairwise comparisons between our method and baselines. That is, for two stories with the same titles but generated by different two models, five well-educated human evaluators are asked to select the one they prefer on the three metrics. In comparison, no *equally good* option is given since the *equally good* option may leads to a careless comparison.

[1] Results on four and five-grams have the same trends.



(a) Inter-story repetition curve by sentence. (b) Inter-story aggregate repetition scores.

(c) Intra-story repetition curve by sentence. (d) Intra-story aggregate repetition scores.

Figure 3: Inter- and intra-story repetition rates by sentences (curves) and for the whole stories (bars). Final results are scaled to [0, 100], the lower the better.

6 Results and Analysis

Table 2 and Figure 3 present the results of automatic evaluation, and Table 3 shows the results of human evaluation. Through analyzing these evaluation results, we have the following observations.

6.1 The Effect of the Planning Strategy

The planning strategy is effective for improving thematic consistency. As shown in Table 2, BLEU-[1-4] scores of Plan-CVAE are significantly higher than the pure CVAE model. Higher BLEU scores indicate that the planning strategy can improve the word-overlapping between the generated stories and the gold standard ones, which means the generated stories are more relevant with thematically consistent cases. For the subjective feelings of humans, as indicated by the human consistency evaluation in Table 3), Plan-CVAE can generate stories with better thematic consistency than the CVAE model. Meanwhile, Plan-CVAE outperforms all baselines on thematic consistency in human evaluation, this means the CVAE model gains a significant improvement on thematic consistency by using the planning strategy.

The planning strategy does not affect the wording diversity. The planning strategy aims to enhance the CVAE model with better thematic consistency while preventing poor wording diversity. Plan-CVAE has a comparable performance with CVAE and outperforms other baselines on distinct scores in Table 2 and the creativity metric in Table 3, which prove that the planning strategy does not affect the wording novelty. In Figure 3, we also have a similar observation that both Plan-CVAE* and Plan-CVAE models achieve a quite low inter- and intra-story repetition rates, which means our proposed model can learn to create stories rather than copy and concatenate frequently occurred phrases in the training corpus.

Table 3: Results of human evaluation.

Readability			
Plan-CVAE	44%	**56%**	S2S
Plan-CVAE	**58%**	42%	AS2S
Plan-CVAE	**67%**	33%	CVAE
Plan-CVAE	47%	**53%**	Plan-and-Write
Consistency			
Plan-CVAE	**65%**	35%	S2S
Plan-CVAE	**61%**	39%	AS2S
Plan-CVAE	**84%**	16%	CVAE
Plan-CVAE	**58%**	42%	Plan-and-Write
Creativity			
Plan-CVAE	**93%**	7%	S2S
Plan-CVAE	**86%**	14%	AS2S
Plan-CVAE	**57%**	43%	CVAE
Plan-CVAE	**81%**	19%	Plan-and-Write

6.2 The Effect of CVAE

The CVAE model can effectively improve the wording diversity. Plan-CVAE outperforms all baselines (excepts for CVAE) on automatic evaluations including distinct scores in Table 2 and inter- and intra-story repetition rates in Figure 3, and on creativity score of human evaluation in Table 3. Specifically, all baselines based on RNNs, i.e., S2S, AS2S, and Plan-and-Write, achieve a quite low distinct score and high inter- and intra-story repetition rates, while Plan-CVAE significantly outperforms them by nearly doubling the distinct scores, reducing the repetition rates to about half of theirs, and achieving similar scores with the pure CVAE model. Results on the creativity metric in human evaluation also indicate the same conclusion. These results support the intuition that CVAE can address the poor wording diversity problem of RNN by randomly sampling from the latent space.

The latent variable in CVAE reduces the readability. CVAE improves the wording diversity by randomly sampling from the latent space. Thus, CVAE produces more uncertainty than RNNs and leads to inferior readability. This intuition is supported by the readability metric in human evaluation (Table 3).

6.3 Case Study

We present two groups of example stories in Table 4 to compare the performance of our proposed method with Plan-and-Write and CVAE since they perform well on either thematic consistency or wording novelty. When compared with the pure CVAE model without planning strategy, we can observe that stories generated by CVAE are formed with novel words but without consistent topics, while stories generated by Plan-CVAE describe more consistent themes and are also narrated with novel wordings. On the other hand, when compared with the planning-based RNN method, we can see that stories generated by Plan-and-Write contain many repeated words and phrases, while stories composed by Plan-CVAE consist of more diverse words.

Although our proposed method has achieved a promising improvement in thematic consistency and wording diversity than baseline models, there is still a gap between stories generated by our method and humans. We also observed some bad cases generated by our method. These bad cases reflect three major problems, i.e., logical inconsistency, lacking sentiment, and weak readability. As for the logical inconsistency problem, one can introduce an extra control to dynamically adjust the keywords planning and content generation process, or establish a polishing mechanism to check and rewrite the generated content. To solve the sentiment problem, one can utilize a sentiment planning strategy to add sentimental information into the generated stories. In order to improve readability, it is worth to use pre-training strategy on larger related corpus for story generation. Our future work will focus on these issues.

Table 4: Two group of example stories automatically generated by our proposed Plan-CVAE model and baseline methods.

		Title: Lake fun
Plan-and-Write	Keywords	friends → lake → water → day → home
	Story	I went to the lake with my friends. We went to a lake. We went to the lake. We went to a water park. We had a great time.
CVAE	Story	The kids rode outside of the neighborhood. The mom helped her to get away from the house. They played with the best and hung them out. Finally, they were a couple. Then, they played games.
Plan-CVAE	Keywords	lake → friends → water → swim → shore
	Story	One summer day, my friends invited me to a different lake. While walking around the lake, we moved along. It was a great summer and my dad built a lot of stuff. Now the rest of the day I can swim. I was very late and went back to shore.
		Title: Tax evasion
Plan-and-Write	Keywords	money → taxes → pay → debt → bank
	Story	Bill had a lot of money. Bill had to pay for his taxes. Bill had to pay for his taxes. Bill was able to pay his taxes. Bill was able to pay his taxes .
CVAE	Story	Sam and her husband wanted to be a lawyer. He worked hard to make it with his lawyer. He just needed to buy more and more things. He made a lot of money. He told her they both hurt more.
Plan-CVAE	Keywords	tax evasion → money → taxes → pay → debt
	Story	Neil had recently moved to a tax preparer. He had applied for a few jobs before the loan. But he didn't notice that the bank was on his list! He was told he owed pay for tax fees. It was about to accept his taxes, but he had no interest.

7 Conclusion

In this paper, we proposed a planning-based conditional variational autoencoder model (Plan-CVAE) for story generation. Our proposed method involves two stages. In the planning stage, the keyword-extraction and keyword-expansion modules are used to generate keywords from the title. As for the generation stage, a CVAE neural network module is employed to generate stories under the guidance of keywords. In our method, the planning strategy aims to improve the thematic consistency while the CVAE module is expected to keep the wording diversity of the generated story. Experimental results of both automatic and human evaluations on a benchmark dataset, i.e., ROCStories corpus, show that our method performs better than existing methods on thematic consistency and wording diversity. The case study also confirms the effectiveness of our method.

Acknowledgements

We would like to thank the reviewers for their constructive comments. This work was supported by the National Key Research and Development Program of China (No. 2017YFC0804001), the National Science Foundation of China (NSFC No. 61876196 and NSFC No. 61672058) and the foundation of Key Laboratory of Artificial Intelligence, Ministry of Education, P.R. China. Rui Yan was sponsored by

Beijing Academy of Artificial Intelligence (BAAI).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany, August. Association for Computational Linguistics.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia, July. Association for Computational Linguistics.

Angela Fan, Mike Lewis, and Yann Dauphin. 2019. Strategies for structuring story generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2650–2660, Florence, Italy, July. Association for Computational Linguistics.

Parag Jain, Priyanka Agrawal, Abhijit Mishra, Mohak Sukhwani, Anirban Laha, and Karthik Sankaranarayanan. 2017. Story generation from sequence of independent short descriptions.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California, June. Association for Computational Linguistics.

Juntao Li, Yan Song, Haisong Zhang, Dongmin Chen, Shuming Shi, Dongyan Zhao, and Rui Yan. 2018. Generating classical Chinese poems via conditional variational autoencoder and adversarial training. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3890–3900, Brussels, Belgium, October-November. Association for Computational Linguistics.

Juntao Li, Lidong Bing, Lisong Qiu, min D Chen, Dongyan Zhao, and Rui Yan. 2019. Learning to write creative stories with thematic consistency. In *AAAI 2019 : Thirty-Third AAAI Conference on Artificial Intelligence*.

Danyang Liu, Juntao Li, Meng-Hsuan Yu, Ziming Huang, Gongshen Liu, Dongyan Zhao, and Rui Yan. 2020. A character-centric neural model for automated story generation. In *AAAI*, pages 1725–1732.

Lara J. Martin, Prithviraj Ammanabrolu, William Hancock, Shruti Singh, Brent Harrison, and Mark O. Riedl. 2018. Event representations for automated story generation with deep neural nets. *ArXiv*, abs/1706.01331.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California, June. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July. Association for Computational Linguistics.

Melissa Roemmele, Sosuke Kobayashi, Naoya Inoue, and Andrew Gordon. 2017. An RNN-based binary classifier for the story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 74–80, Valencia, Spain, April. Association for Computational Linguistics.

Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley, 2010. *Automatic Keyword Extraction from Individual Documents*, pages 1 – 20. 03.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 3295–3301. AAAI Press.

Xiaoyu Shen, Hui Su, Yanran Li, Wenjie Li, Shuzi Niu, Yang Zhao, Akiko Aizawa, and Guoping Long. 2017. A conditional variational framework for dialog generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 504–509, Vancouver, Canada, July. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Tianming Wang and Xiaojun Wan. 2019. T-cvae: Transformer-based conditioned variational autoencoder for story completion. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5233–5239. International Joint Conferences on Artificial Intelligence Organization, 7.

Jingjing Xu, Xuancheng Ren, Yi Zhang, Qi Zeng, Xiaoyan Cai, and Xu Sun. 2018. A skeleton-based model for promoting coherence among sentences in narrative story generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4306–4315, Brussels, Belgium, October-November. Association for Computational Linguistics.

Xiaopeng Yang, Xiaowen Lin, Shunda Suo, and Ming Li. 2018. Generating thematic chinese poetry using conditional variational autoencoders with hybrid decoders. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, page 4539–4545. AAAI Press.

Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7378–7385.

Meng-Hsuan Yu, Juntao Li, Danyang Liu, Bo Tang, Haisong Zhang, Dongyan Zhao, and Rui Yan. 2020. Draft and edit: Automatic storytelling through multi-pass hierarchical conditional variational autoencoder. In *AAAI*, pages 1741–1748.

Biao Zhang, Deyi Xiong, Jinsong Su, Hong Duan, and Min Zhang. 2016. Variational neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 521–530, Austin, Texas, November. Association for Computational Linguistics.

Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–664, Vancouver, Canada, July. Association for Computational Linguistics.

Towards Causal Explanation Detection with Pyramid Salient-Aware Network

Xinyu Zuo[1,2], **Yubo Chen**[1,2], **Kang Liu**[1,2], **Jun Zhao**[1,2]
[1]National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing, 100190, China
[2]School of Artificial Intelligence,
University of Chinese Academy of Sciences, Beijing, 100049, China
{xinyu.zuo,yubo.chen,kliu,jzhao}@nlpr.ia.ac.cn

Abstract

Causal explanation analysis (CEA) can assist us to understand the reasons behind daily events, which has been found very helpful for understanding the coherence of messages. In this paper, we focus on *Causal Explanation Detection*, an important subtask of causal explanation analysis, which determines whether a causal explanation exists in one message. We design a **P**yramid **S**alient-**A**ware **N**etwork (PSAN) to detect causal explanations on messages. PSAN can assist in causal explanation detection via capturing the salient semantics of discourses contained in their keywords with a bottom graph-based word-level salient network. Furthermore, PSAN can modify the dominance of discourses via a top attention-based discourse-level salient network to enhance explanatory semantics of messages. The experiments on the commonly used dataset of CEA shows that the PSAN outperforms the state-of-the-art method by 1.8% F1 value on the *Causal Explanation Detection* task.

1 Introduction

Causal explanation detection (CED) aims to detect whether there is a causal explanation in a given message (e.g. a group of sentences). Linguistically, there are coherence relations in messages which explain how the meaning of different textual units can combine to jointly build a discourse meaning for the larger unit. The explanation is an important relation of coherence which refers to the textual unit (e.g. discourse) in a message that expresses explanatory coherent semantics (Jurafsky, 2010). As shown in Figure 1, M1 can be divided into three discourses, and D2 is the explanation that expresses the reason why it is advantageous for the equipment to operate at these temperatures. CED is important for tasks that require an understanding of textual expression (Son et al., 2018). For example, for question answering, the answers of questions are most likely to be in a group of sentences that contains causal explanations (Oh et al., 2013). Furthermore, the summarization of event descriptions can be improved by selecting causally motivated sentences (Hidey and McKeown, 2016). Therefore, CED is a problem worthy of further study.

The existing methods mostly regard this task as a classification problem (Son et al., 2018). At present, there are mainly two kinds of methods, feature-based methods and neural-based methods, for similar semantic understanding tasks in discourse granularity, such as opinion sentiment classification and discourse parsing (Nejat et al., 2017; Jia et al., 2018; Soricut and Marcu, 2003). The feature-based methods can extract the feature of the relation between discourses. However, these methods do not deal well with the implicit instances which lack explicit features. For CED, as shown in Figure 1, D2 lacks explicit features such as *because of*, *due to*, or the features of tenses, which are not friendly for feature-based methods. The methods based on neural network are mainly Tree-LSTM model (Wang et al., 2017) and hierarchical Bi-LSTM model (Son et al., 2018). The Tree-LSTM models learn the relations between words to capture the semantics of discourses more accurately but lack further understanding of the semantics between discourses. The hierarchical Bi-LSTM models can employ sequence structure to implicitly learn the relations between words and discourses. However, previous work shows that compared with Tree-LSTM, Bi-LSTM lacks a direct understanding of the dependency relations between words. Therefore, the method of implicit learning of inter-word relations is not prominent in the tasks related

Proceedings of the 19th China National Conference on Computational Linguistics, pages 903-914,
Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

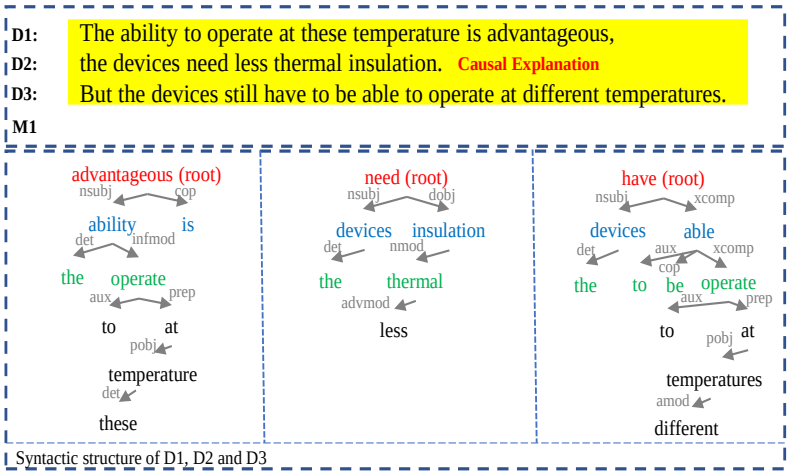


Figure 1: Instance of causal explanation analysis (CEA). The top part is a message which contains its segmented discourses and a causal explanation. The bottom part is the syntactic dependency structures of three discourses divided from M1.

to understanding the semantic relations of messages (Li et al., 2015). Therefore, how to directly learn the relations between words effectively and consider discourse-level correlation to further filter the key information is a valuable point worth studying.

Further analysis, why do the relations between words imply the semantics of the message and its discourses? From the view of computational semantics, the meaning of a text is not only the meaning of words but also the relation, order, and aggregation of the words. In other simple words is that the meaning of a text is partially based on its syntactic structure (Jurafsky, 2010). In detail, in CED, the core and subsidiary words of discourses contain their basic semantics. For example, as D1 shown in Figure 1, according to the word order in syntactic structure, we can capture the *ability* of *temperature* is *advantageous*. We can understand the basic semantic of D1 which expresses some kind of *ability* is *advantageous* via root words *advantageous* and its affiliated words. Additionally, why the correlation and key information at the discourse level are so important to capture the causal explanatory semantics of the message? Through observation, the different discourse has a different status for the explanatory semantics of a message. For example, in M1, combined with D1, D2 expresses the explanatory semantics of *why the ability to work at these temperatures is advantageous*, while D3 expresses the semantic of transition. In detail, D1 and D2 are the keys to the explanatory semantics of M1, and if not treated D1, D2, and D3 differently, the transitional semantic of D3 can affect the understanding of the explanatory semantic of M1. Therefore, how to make better use of the information of keywords in the syntactic structure and pay more attention to the discourses that are key to explanatory semantics is a problem to be solved.

To this end, we propose a **P**yramid **S**alient-**A**ware **N**etworks (PSAN) which utilizes keywords on the syntactic structure of each discourse and focuses on the key discourses that are critical to explanatory semantics to detect causal explanation of messages. First, what are the keywords in a syntactic structure? From the perspective of syntactic dependency, the root word is the central element that dominates other words, while it is not be dominated by any of the other words, all of which are subordinate to the root word (Zhang and Xiaojun, 2014). From that, the root and subsidiary words in the dependency structure are the keywords at the syntax level of each discourse. Specifically, we sample 100 positive sentences from training data to illuminate whether the keywords obtained through the syntactic dependency contain the causal explanatory semantics. And we find that the causal explanatory semantics of more than 80% sentences be captured by keywords in dependency structure[1]. Therefore, we extract the root word and its surrounding words on the syntactic dependency of each discourse as its keywords.

[1]Five Ph.D. students majoring in NLP judge whether sentences could be identified as which containing causal explanatory semantics by the root word and its surrounding words in syntactic dependency, and the agreement consistency is 0.8

Next, we need to consider how to make better use of the information of keywords contained in the syntactic structure. To pay more attention to keywords, the common way is using attention mechanisms to increase the attention weight of them. However, this implicitly learned attention is not very interpretable. Inspired by previous researches (Vashishth et al., 2019; Bastings et al., 2017), we propose a bottom graph-based word-level salient network which merges the syntactic dependency to capture the salient semantics of discourses contained in their keywords. Finally, how to consider the correlation at the discourse level and pay more attention to the discourses that are key to the explanatory semantics? Inspired by previous work (Li et al., 2016), we propose a top attention-based discourse-level salient network to focus on the key discourses in terms of explanatory semantics.

In summary, the contributions of this paper are as follows:

- We design a **P**yramid **S**alient-**A**ware **N**etwork (PSAN) to detect causal explanations of messages which can effectively learn the pivotal relations between keywords at word level and further filter the key information at discourse level in terms of explanatory semantics.
- PSAN can assist in causal explanation detection via capturing the salient semantics of discourses contained in their keywords with a bottom graph-based word-level salient network. Furthermore, PSAN can modify the dominance of discourses via a top attention-based discourse-level salient network to enhance explanatory semantics of messages.
- Experimental results on the open-accessed commonly used datasets show that our model achieves the best performance. Our experiments also prove the effectiveness of each module.

2 Related Works

Causal Semantic Detection: Recently, causality detection which detects specific causes and effects and the relations between them has received more attention, such as the researches proposed by Li (Li and Mao, 2019), Zhang (Zhang et al., 2017), Bekoulis (Bekoulis et al., 2018), Do (Do et al., 2011), Riaz (Riaz and Girju, 2014), Dunietz (Dunietz et al., 2017a) and Sharp (Sharp et al., 2016). Specifically, to extract the causal explanation semantics from the messages in a general level, some researches capture the causal semantics in messages from the perspective of discourse structure, such as capturing counterfactual conditionals from a social message with the PDTB discourse relation parsing (Son et al., 2017), a pre-trained model with Rhetorical Structure Theory Discourse Treebank (RSTDT) for exploiting discourse structures on movie reviews (Ji and Smith, 2017), and a two-step interactive hierarchical Bi-LSTM framework (Xia and Ding, 2019) to extract emotion-cause pair in messages. Furthermore, Son (2018) defines the causal explanation analysis task (CEA) to extract causal explanatory semantics in messages and annotates a dataset for other downstream tasks. In this paper, we focus on causal explanation detection (CED) which is the fundamental and important subtask of CEA.

Syntactic Dependency with Graph Network: Syntactic dependency is a vital linguistic feature for natural language processing (NLP). There are some researches employ syntactic dependency such as retrieving question answering passage assisted with syntactic dependency (Cui et al., 2005), mining opinion with syntactic dependency (Wu et al., 2009) and so on. For tasks related to causal semantics extraction from relevant texts, dependency syntactic information may evoke causal relations between discourse units in text (Gao et al., 2019). And recently, there are some researches (Marcheggiani and Titov, 2017; Zhang et al., 2018) convert the syntactic dependency into a graph with graph convolutional network (GCN) (Kipf and Welling, 2016) to effectively capture the syntactic dependency semantics between words in context, such as a semantic role model with GCN (Marcheggiani and Titov, 2017), a GCN-based model assisted with a syntactic dependency to improving relation extraction (Zhang et al., 2018). In this paper, we capture the salient explanatory semantics based on the syntactic-centric graph.

Self-attention Mechanism: Self-attention has been introduced to machine translation by Vaswani (Vaswani et al., 2017) for capturing global dependencies between input and output and achieved state-of-the-art performance. For language understanding tasks, Shen (Shen et al., 2018) exploits self-attention to learn long-range dependencies. Tan (Tan et al., 2018) applies self-attention for semantic role labeling

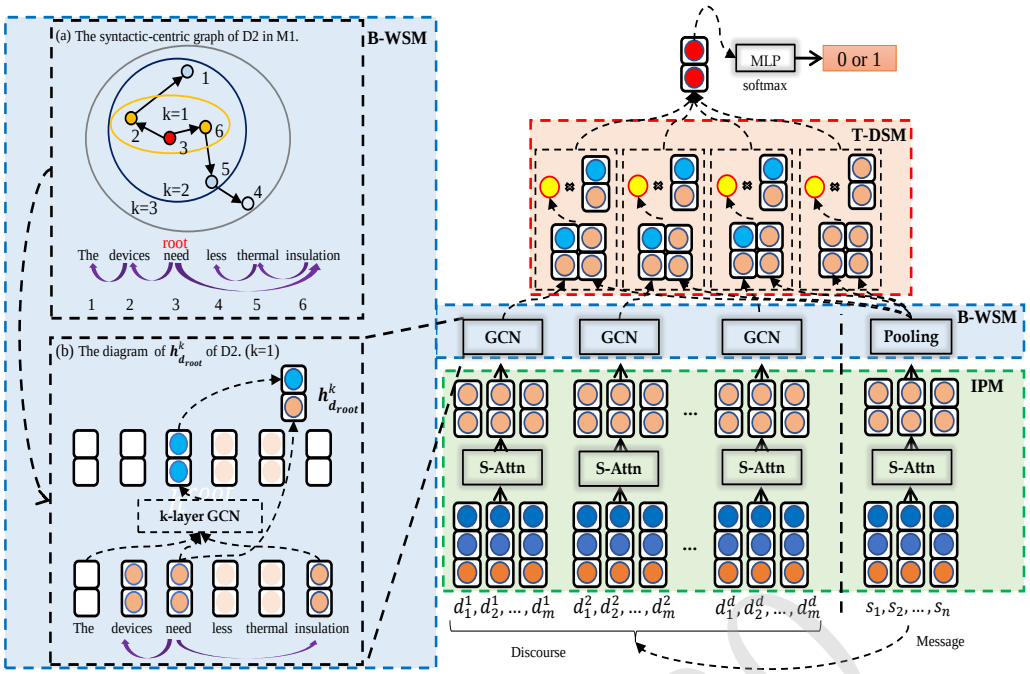


Figure 2: The structure of the pyramid salient-aware network (PSAN). The left side is the detail of the bottom word-level salient-aware module (B-WSM), the top of right side is the top discourse-level salient-aware module (T-DSM) and the bottom of right side is the input processing module (IPM).

task and achieves state-of-the-art results. In this paper, we utilize a multi-head self-attention encoder to capture the representation of words.

3 Methodology

The architecture of our proposed model is illustrated in Figure 2. In this paper, the Pyramid Salient-Aware Network (PSAN) primarily involves the following three components: (i) **input processing module (IPM)**, which processes and encodes the input message and its discourses via self-attention module; (ii) **bottom word-level salient-aware module (B-WSM)**, which captures the salient semantics of discourses contained in their keywords based on the syntactic-centric graph; (iii) **top discourse-level salient-aware module (T-DSM)**, which modifies the dominance of different discourse based on the message-level constraint in terms of explanatory semantic via an attention mechanism, and obtain the final causal explanatory representation of input message m.

3.1 Input Processing Module

In this component, we split the input message m into discourses d. Specially, we utilize the self-attention encoder to encode input messages and their corresponding discourses.

3.1.1 Discourse Extraction

As shown in Figure 1, we split the message into discourses with the same segmentation methods as Son (2018) based on semantic coherence. In detail, first, we regard (‘,’), (‘.’), (‘!’), (‘?’) tags and periods as discourse makers. Next, we also extract the discourse connectives set from PDTB2 as discourse makers. Specifically, we remove some simple connectives (e.g. I like running **and** basketball) from extracted discourse marks. Finally, we divide messages into discourses by the discourse makers.

3.1.2 Embedding Layer

For the input message $s = \{s_1, ..., s_n\}$ and discourse $d = \{d^d_1, ..., d^d_m\}$ separated from s, we lookup embedding vector of each word s_n (d^d_m) as $\boldsymbol{s_n}$ ($\boldsymbol{d^d_m}$) from the pre-trained embedding. Finally, we obtain

the word representation sequence $\boldsymbol{s} = \{\boldsymbol{s_1}, ..., \boldsymbol{s_n}\}$ of message s and $\boldsymbol{d} = \{\boldsymbol{d_1^d}, ..., \boldsymbol{d_m^d}\}$ of discourse d corresponding to s.

3.1.3 Word Encoding

Inspired by the application of self-attention to multiple tasks (Tan et al., 2018; Cao et al., 2018), we exploit multi-head self-attention encoder to encode input words. The scaled dot-product attention can be described as follows:

$$(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{\boldsymbol{d}}}\right)\boldsymbol{V} \tag{1}$$

where $\boldsymbol{Q} \in \mathbb{R}^{N \times 2dim_h}$, $\boldsymbol{K} \in \mathbb{R}^{N \times 2dim_h}$ and $\boldsymbol{V} \in \mathbb{R}^{N \times 2dim_h}$ are query matrices, keys matrices and value matrices, respectively. In our setting, $\boldsymbol{Q} = \boldsymbol{K} = \boldsymbol{V} = \boldsymbol{s}$ for encoding sentence, and $\boldsymbol{Q} = \boldsymbol{K} = \boldsymbol{V} = \boldsymbol{d}$ for encoding discourse.

Multi-head attention first projects the queries, keys, and values h times by using different linear projections. The results of attention are concatenated and once again projected to get the final representation. The formulas are as following:

$$head_i = \text{Attention}\left(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V\right) \tag{2}$$

$$\mathbf{H}' = (head_i \oplus \ldots \oplus head_h)\,\mathbf{W}_o \tag{3}$$

where, $\mathbf{W}_i^Q \in \mathbb{R}^{2dim_h \times dim_k}$, $\mathbf{W}_i^K \in \mathbb{R}^{2dim_h \times dim_k}$, $\mathbf{W}_i^V \in \mathbb{R}^{2dim_h \times dim_k}$ and $\mathbf{W}_o \in \mathbb{R}^{2dim_h \times 2dim_h}$ are projection parameters and $dim_k = 2dim_h/h$. And the output is the encoded message $\boldsymbol{H}_S^{ed} = \{\boldsymbol{h}_{s_1}^{ed}, ..., \boldsymbol{h}_{s_n}^{ed}\}$ and discourse $\boldsymbol{H}_{D^d}^{ed} = \{\boldsymbol{h}_{d_1^d}^{ed}, ..., \boldsymbol{h}_{d_m^d}^{ed}\}$.

3.2 Bottom Word-Level Salient-Aware Module

In this component, we aim to capture the salient semantics of discourses contained in their keywords based on syntactic-centric graphs. For each discourse, first, it extracts the syntactic dependency and constructs the syntactic-centric graph. Second, it collects the keywords and their inter-relations to capture the discourse-level salient semantic based on the syntactic-centric graph.

3.2.1 Syntactic-Centric Graph Construction

We construct a syntactic-centric graph of each discourse based on syntactic dependency to assist in capturing the semantics of discourses. We utilize Stanford CoreNLP tool[2] to extract the syntactic dependency of each discourse and convert them into syntactic-centric graphs. Specifically, in the syntactic-centric graph, the nodes represent words, and the edges represent whether there is a dependency relation between two words or not. As shown in the subplot (a) of Figure 2, *need* is the root word in the syntactic dependency of *"the devices need less thermal insulation"* (D2 in S1), and words which are syntactically dependent on each other are connected with solid lines.

3.2.2 Keywords Collection and Salient Semantic Extraction

For each discourse, we collect the keywords based on the syntactic-centric graph and capture the salient semantic based on the syntactic-centric graph from its keywords. Firstly, as illustrated in section 1, we combine the root word and the affiliated words that connected with the root word in k hops as the keywords. For example, as shown in Figure 2, when $k = 1$, the keywords are {*need, devices and insulation*}, and the keywords are {*need, devices, insulation, the and thermal*} when $k = 2$. Secondly, inspired by previous works, we utilize k-layer graph convolutional network (GCN) (Kipf and Welling, 2016) to encode the k hops connected keywords based on the syntactic-centric graph. For example, when $k = 1$, we encode 1-hop keywords with 1-layer GCN to capture the salient semantic. Specifically, we can capture different degrees of salient semantics by changing the value of k. However, it is not the larger the value of k, the deeper the salient semantics are captured. Conversely, the larger the k, the more

[2]https://stanfordnlp.github.io/CoreNLP/

noises are likely to be introduced. For example, when $k = 1$, *need*, *devices* and *insulation* are enough to express the salient semantic of D2 (working at these temperatures need less insulation). Finally, we select the representation of the root word in the final layer as the discourse-level representation which contains the salient semantic.

The graph convolutional network (GCN) (Kipf and Welling, 2016) is a generalization of convolutional neural network (LeCun et al., 1998) for encoding graphs. In detail, given a syntactic-centric graph with v nodes, we utilize an $v \times v$ adjacency matrix $\boldsymbol{A}$, where $A_{ij} = 1$ if there is an edge between node i and node j. In each layer of GCN, for each node, the input is the output $\boldsymbol{h}_i^{k-1}$ of the previous layer (the input of the first layer is the original encoded input words and features) and the output of node i at k-th layer is $\boldsymbol{h}_i^k$, the formula is as following:

$$\boldsymbol{h}_i^k = \sigma\left(\sum_{j=1}^{v} A_{ij} W^k \boldsymbol{h}_j^{k-1} + b^k\right) \tag{4}$$

where W^k is the matrice of linear transformation, b^k is a bias term and σ is a nonlinear function.

However, naively applying the graph convolution operation in Equation (3) could lead to node representations with drastically different magnitudes because the degree of a token varies a lot. This issue may cause the information in h_i^{k-1} is never carried over to h_i^k because nodes never connect to themselves in a dependency graph (Zhang et al., 2018). In order to resolve the issue that the information in h_i^{k-1} may be never carried over to h_i^k due to the disconnection between nodes in a dependency graph, we utilize the method raised by Zhang (2018) which normalizes the activations in the GCN, and adds self-loops to each node in graph:

$$\boldsymbol{h}_i^k = \sigma\left(\sum_{j=1}^{v} \tilde{A}_{ij} W^k \boldsymbol{h}_j^{k-1} / d_i + b^k\right), \tag{5}$$

where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, $\mathbf{I}$ is the $v \times v$ identify matrix and $d_i = \sum_{j=1}^{v} \tilde{A}_{ij}$ is the degree of word i in graph.

Finally, We select the representation $\boldsymbol{h}_{d_{root}}^k$ of the root word in final layer GCN as the salient representation of d-th discourse in message s. For example, as shown in the subplot (b) of Figure 2, we choose the representation of *need* in the final layer as the salient representation of the discourse *"the devices need less thermal insulation"*.

3.3 Top Discourse-Level Salient-Aware Module

How to make better use of the relation between discourse and extract the message-level salient semantic? We modify the dominance of different discourse based on the message-level constraint in terms of explanatory semantic via an attention mechanism. First, we extract the global semantic of message s which contains its causal explanatory tendency. Next, we modify the dominance of different discourse based on global semantic. Finally, we combine the modified representation to obtain the final causal explanatory representation of input message s.

3.3.1 Global Semantic Extraction

Inspired by previous research (Son et al., 2018), the average encoded word representation of all the words in message can represent its overall semantic simply and effectively. We utilize the average pooling on the encoded representation $\boldsymbol{H}_S^{ed}$ of message s to obtain the global representation which contains the global semantic of its causal explanatory tendency. The formula is as following:

$$\boldsymbol{h}_s^{glo} = \sum_{\boldsymbol{h}_s^{ed} \in \boldsymbol{H}_S^{ed}} \boldsymbol{h}_s^{ed} / n, \tag{6}$$

where $\boldsymbol{h}_s^{glo}$ is the global representation of message s via average pooling operation and n is the number of words.

3.3.2 Dominance Modification

We modify the dominance of different discourse based on the global semantic which contains its causal explanatory tendency via an attention mechanism. In detail, after obtaining the global representation $\boldsymbol{h}_s^{glo}$, we modify the salient representation $\boldsymbol{h}_{d_{root}}^k$ of discourses d constrained with $\boldsymbol{h}_s^{glo}$. Finally, we obtain final causal representation $\boldsymbol{h}_s^{caul}$ of message s via attention mechanism:

$$\alpha_{ss} = \boldsymbol{h}_s^{glo} \boldsymbol{W}_f (\boldsymbol{h}_s^{glo})^T \tag{7}$$

$$\alpha_{sd} = \boldsymbol{h}_s^{glo} \boldsymbol{W}_f (\boldsymbol{h}_{d_{root}}^k)^T \tag{8}$$

$$\left[\alpha_{ss}^{'}, \cdots, \alpha_{sd}^{'}\right] = softmax([\alpha_{ss}, ..., \alpha_{sd}]) \tag{9}$$

$$\boldsymbol{h}_s^{caul} = \alpha_{ss}^{'} \boldsymbol{h}_s^{glo} + ... + \alpha_{sd}^{'} \boldsymbol{h}_{d_{root}}^k, \tag{10}$$

where the $\boldsymbol{W}_f$ is matrice of linear transformation, $\alpha_{ss}^{'}$, $\alpha_{sd}^{'}$ are the attention weight. Finally, we mapping $\boldsymbol{h}_s^{caul}$ into a binary vector and get the output via a softmax operation.

4 Experiment

Dataset We mainly evaluate our model on a unique dataset devoted to causal explanation analysis released by Son (2018). This dataset contains 3,268 messages consist of 1598 positive messages that contain a causal explanation and 1670 negative sentences randomly selected. Annotators annotate which messages contain causal explanations and which text spans are causal explanations (a discourse with a tendency to interpret something). We utilize the same 80% of the dataset for training, 10% for tuning, and 10% for evaluating as Son (2018). Additionally, to further prove the effectiveness of our proposed model, we regard sentences with causal semantic discourse relations in PDTB2 and sentences containing causal span pairs in BECauSE Corpus 2.0 (Dunietz et al., 2017b) as supplemental messages with causal explanations to evaluate our model. In this paper, PDTB-CED and BECauSE-CED are used to represent the two supplementary datasets respectively.

Parameter Settings We set the length of the sentence and discourse as 100 and 30 respectively. We set the batch size as 5 and the dimension of the output in each GCN layer as 50. Additionally, we utilize the 50-dimension word vector pre-trained with Glove. For optimization, we utilize Adam (Kingma and Ba, 2014) with 0.001 learning rate. We set the maximum training epoch as 100 and adopt an early stop strategy based on the performance of the development set. All the results of different compared and ablated models are the average result of five independent experiments.

Compared Models We compare our proposed model with feature-based and neural-based model: (1) **Lin et al. (2014)**: an end-to-end discourse relation parser on PDTB, (2) **Linear SVM**: a linear designed feature based SVM classifier, (3) **RBF SVM**: a complex designed feature based SVM classifier, (4) **Random Forest**: a random forest classifier which relies on designed features, (5) **Son et al. (2018)**: a hierarchical LSTM sequence model which is designed specifically for CEA. (6) **H-BiLSTM + BERT**[3][4]: a fine-tuned language model (BERT) which has been shown to improve the performance in some other classification tasks based on (5), (7) **H-Atten.**: a well-used Bi-LSTM model that captures hierarchical key information based on hierarchical attention mechanism, (8) **Our model**: our proposed pyramid salient-aware network (PSAN). Furthermore, we evaluate the performance of the model (5), (7), and (8) on the supplemental dataset to prove the effectiveness of our proposed model. Additionally, we design different ablation experiments to demonstrate the effectiveness of the bottom word-level salient-aware module (B-WSM), top discourse-level salient-aware module (T-DSM), and the influence of different depths in the syntactic-centric graph.

[3]https://github.com/huggingface/transformers

[4]BERT can not be applied to the feature-based model suitably, so we deploy BERT on the latest neural network model to make the comparison to prove the effectiveness of our proposed model.

4.1 Main Results

Model	F1	F1	F1
	Facebook	PDTB-CED	BEcuasE-CED
Lin et al. (2014)	63.8	-	-
Linear SVM (Son et al., 2018)	79.1	-	-
RBF SVM (Son et al., 2018)	77.7	-	-
Random Forest (Son et al., 2018)	77.1	-	-
Son et al. (2018)	75.8	63.6	69.6
H-Atten.	80.9	70.6	76.5
H-BiLSTM + BERT	85.0	-	-
Our model	**86.8**	**76.6**	**81.7**

Table 1: Comparisons of the state-of-the-art methods on causal explanation detection.

Table 1 shows the comparison results on the Facebook dataset and two supplementary datasets. From the results, we have the following observations.

(1) Comparing with the current best feature-based and neural-based models on CED: **Lin et al. (2014)**, **Linear SVM** and **Son et al. (2018)**, **our model** improves the performance by 23.0, 7.7 and 11.0 points on F1, respectively. It illustrates that the pyramid salient-aware network (PSAN) can effectively extract and incorporate the word-level key relation and discourse-level key information in terms of explanatory semantics to detect causal explanation. Furthermore, comparing with the well-used hierarchical key information captured model (**H-Atten.**), **our model** improves the performance by 5.9 points on F1. This confirms the statement in section 1 that directly employing the relation between words with syntactic structure is more effective than the implicit learning.

(2) Comparing the **Son et al. (2018)** with pre-trained language model (**H-BiLSTM+BERT**), there is 9.2 points improvement on F1. It illustrates that the pre-trained language model (LM) can capture some causal explanatory semantics with the large-scale corpus. Furthermore, **our model** can further improve performance by 1.8 points compared with **H-BiLSTM+BERT**. We believe the reason is that the LM is pre-trained with large-scale regular sentences that do not contain causal semantics only, which is not specifically suitable for CED compared to the proposed model for explanatory semantic. Furthermore, the performance of **H-Atten.** is better than **Son et al. (2018)** which indicates focusing on salient keywords and key discourses helps understand explanatory semantics.

(3) It is worth noting that, regardless of our proposed model, comparing the **Linear SVM** with **Son et al. (2018)**, the simple feature classifier is better than the simple deep learning model for CED on the Facebook dataset. However, when combining the syntactic-centric features with deep learning, we could achieve a significant improvement. In other words, our model can effectively combine the *interpretable information* of the feature-based model with the *deep understanding* of the deep learning model.

(4) To further prove the effectiveness of the proposed model, we evaluate **our model** on supplemental messages with causal semantics in other datasets (PDTB-CED and BEcausE-CED). As shown in Table 1, the results show that the proposed model performs significantly better than the **Son et al. (2018)** and **H-Atten.** on the other two datasets[5]. It further demonstrates the effectiveness of our proposed model.

(5) Moreover, **our model** is twice as fast as the **Son et al. (2018)** during training because of the computation of self-attention and GCN is parallel. It illustrates that our model can consume less time and achieve significant improvement in causal explanation detection. Moreover, compared with the feature-based models, the neural-based models rely less on artificial design features.

Dataset	Facebook		PDTB-CED		BEcausE-CED	
Model	F1	∇	F1	∇	F1	∇
our model	**86.8**	-	**76.6**	-	**81.7**	-
w/o B-WSM + root	80.1	-6.7	69.9	-6.7	75.8	-5.9
w/o B-WSM + ave	84.7	-2.1	74.4	-2.2	79.8	-1.9

Table 2: Effectiveness of B-WSM. (**w/o** B-WSM denotes the models without B-WSM. + denotes repalcing the B-WSM with the module after +. **root** denotes using the encoded representation of the root word in each discourse to represent it. **ave** denotes using the average encoded representation of words in discourse to represent it.)

4.2 Effectiveness of Bottom Word-Level Salient-Aware Module (B-WSM)

Table 2 tries to show the effectiveness of the salient information contained in the keywords of each discourse captured via the proposed B-WSM for causal explanation detection (3.2). The results illustrate B-WSM can effectively capture the salient information which contains the most causal explanatory semantics. It is worth noting that when using the average encoded-word representation to represent each discourse (**w/o B-WSM + ave**), the model also achieves acceptable performance. This confirms the conclusion from Son (2018) that the average word representation at word level contains certain causal explanatory semantic. Furthermore, only the root word of each discourse also contains some causal semantics (**w/o B-WSM + root**) which proves the effectiveness of capturing salient information via syntactic dependency from the keywords.

4.3 Effectiveness of Top Discourse-Level Salient-Aware Module (T-DSM)

Dataset	Facebook		PDTB-CED		BEcausE-CED	
Model	F1	∇	F1	∇	F1	∇
our model	**86.8**	-	**76.6**	-	**81.7**	-
w/o T-DSM + seq D	83.8	-3.0	72.9	-3.7	78.1	-3.6
w/o T-DSM + ave S/D	84.0	-2.8	73.5	-3.1	77.8	-3.9

Table 3: Effectiveness of T-DSM. (**w/o** T-DSM denotes models without T-DSM. + denotes replacing the T-DSM with the module after +. **seq D** denotes mapping the representation of discourses via a sequence LSTM to represent the whole message. **ave S/D** denotes using the average encoded representation of words in message and its discourses to represent the whole message.)

Table 3 tries to show the effectiveness of the salient information of the key discourses modified and incorporated via T-DSM for causal explanation detection (3.3). The results compared with **w/o T-DSM + seq D** illustrate our T-DSM can effectively modify the dominance of different discourses based on the global semantic constraint via an attention mechanism to enhance the causal explanatory semantic. Specifically, the results of **w/o T-DSM + ave S/D** show that both discourse-level representation and global representation contain efficient causal explanatory semantics, which further proves the effectiveness of the proposed T-DSM.

4.4 Comparisons of Different Depths of Syntactic-Centric Semantic

To demonstrate the influence of the causal explanatory semantics contained in the syntactic-centric graph with different depths, we further compare the performance of our proposed model with a different number of GCN layers. As shown in Figure 3, when the number of GCN layers is 2, the most efficient syntactic-centric information can be captured for causal explanation detection.

[5]We obtain the performance with the publicly released code by Son et al. (2018). The supplementary datasets are not specifically suitable for this task, and the architectural details of designed feature-based models are not public, so we only compare the performance of the latest model to prove the effectiveness of our proposed model.

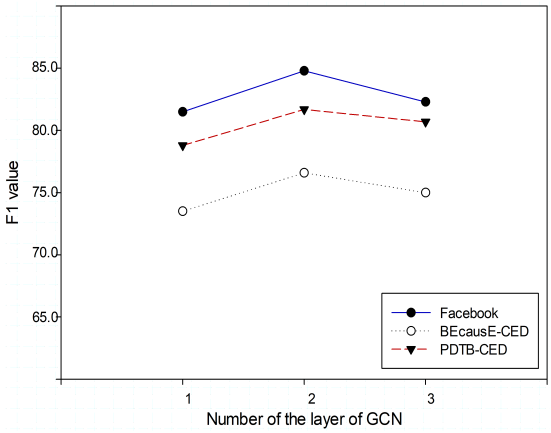


Figure 3: Comparisons of different number of GCN layers.

4.5 Error Analysis

As shown in Figure 4, we find the two main difficulties in this task:

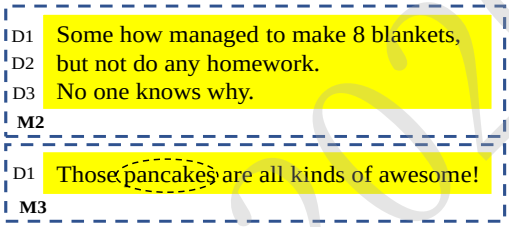


Figure 4: Predictions of the proposed model.

(1) **Emotional tendency** The same expression can convey different semantic under different emotional tendencies, especially in this kind of colloquial expressions. As M2 shown in Figure 4, *make 8 blankets* expresses *anger* over *not do any homework*, and our model wrongly predicts the *make 8 blankets* is the reason for *not do any homework*.

(2) **Excessive semantic parsing** Excessive parsing of causal intent by the model will lead to identifying messages that do not contain causal explanations as containing. As shown in Figure 4, M3 means pancakes are awesome, but the model overstates the reason for *awesome* is a pancake.

5 Conclusion

In this paper, we devise a pyramid salient-aware network (PSAN) to detect causal explanations in messages. PSAN can effectively learn the key relation between words at the word level and further filter out the key information at the discourse level in terms of explanatory semantics. Specifically, we propose a bottom word-level salient-aware module to capture the salient semantics of discourses contained in their keywords based on a the syntactic-centric graph. We also propose a top discourse-level salient-aware module to modify the dominance of different discourses in terms of global explanatory semantic constraint via an attention mechanism. Experimental results on the open-accessed commonly used datasets show that our model achieves the best performance.

Acknowledgements

This work is supported by the Natural Key RD Program of China (No.2018YFB1005100), the National Natural Science Foundation of China (No.61533018, No.61922085, No.61976211, No.61806201) and the Key Research Program of the Chinese Academy of Sciences (Grant NO. ZDBS-SSW-JSC006). This work is also supported by Beijing Academy of Artificial Intelligence (BAAI2019QN0301), CCF-Tencent Open Research Fund and independent research project of National Laboratory of Pattern Recognition.

References

Joost Bastings, Ivan Titov, Wilker Aziz, Diego Marcheggiani, and Khalil Sima'an. 2017. Graph convolutional encoders for syntax-aware neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1957–1967, Copenhagen, Denmark, September. Association for Computational Linguistics.

Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. 2018. Adversarial training for multi-context joint entity and relation extraction. *arXiv preprint arXiv:1808.06876*.

Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2018. Adversarial transfer learning for Chinese named entity recognition with self-attention mechanism. In *Empirical Methods in Natural Language Processing*, pages 182–192, Brussels, Belgium, October-November. Association for Computational Linguistics.

Hang Cui, Renxu Sun, Keya Li, Min-Yen Kan, and Tat-Seng Chua. 2005. Question answering passage retrieval using dependency relations. In *ACM SIGIR*, pages 400–407. ACM.

Quang Xuan Do, Yee Seng Chan, and Dan Roth. 2011. Minimally supervised event causality identification. In *Empirical Methods in Natural Language Processing*, pages 294–303. Association for Computational Linguistics.

Jesse Dunietz, Lori Levin, and Jaime Carbonell. 2017a. Automatically tagging constructions of causation and their slot-fillers. *TACL*, 5:117–133.

Jesse Dunietz, Lori Levin, and Jaime Carbonell. 2017b. The BECauSE corpus 2.0: Annotating causality and overlapping relations. In *Proceedings of the 11th Linguistic Annotation Workshop*, pages 95–104, Valencia, Spain, April. Association for Computational Linguistics.

Lei Gao, Prafulla Kumar Choubey, and Ruihong Huang. 2019. Modeling document-level causal structures for event causal relation identification. In *North American Chapter of the Association for Computational Linguistics*, pages 1808–1817, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Christopher Hidey and Kathy McKeown. 2016. Identifying causal relations using parallel Wikipedia articles. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1424–1433, Berlin, Germany, August. Association for Computational Linguistics.

Yangfeng Ji and Noah Smith. 2017. Neural discourse structure for text categorization. *arXiv preprint arXiv:1702.01829*.

Yanyan Jia, Yuan Ye, Yansong Feng, Yuxuan Lai, Rui Yan, and Dongyan Zhao. 2018. Modeling discourse cohesion for discourse parsing via memory network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 438–443, Melbourne, Australia, July. Association for Computational Linguistics.

Daniel Jurafsky. 2010. *Speech and Language Processing: An Introduction to Natural Language*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Pengfei Li and Kezhi Mao. 2019. Knowledge-oriented convolutional neural network for causal relation extraction from natural language texts. *Expert Systems with Applications*, 115:512–523.

Jiwei Li, Thang Luong, Dan Jurafsky, and Eduard Hovy. 2015. When are tree structures necessary for deep learning of representations? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2304–2314, Lisbon, Portugal, September. Association for Computational Linguistics.

Qi Li, Tianshi Li, and Baobao Chang. 2016. Discourse parsing with attention-based hierarchical neural networks. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 362–371, Austin, Texas, November. Association for Computational Linguistics.

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2014. A pdtb-styled end-to-end discourse parser. *Natural Language Engineering*, 20(2):151–184.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Empirical Methods in Natural Language Processing*, pages 1506–1515, sep.

Bita Nejat, Giuseppe Carenini, and Raymond Ng. 2017. Exploring joint neural model for sentence level discourse parsing and sentiment analysis. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 289–298, Saarbrücken, Germany, August. Association for Computational Linguistics.

Jong-Hoon Oh, Kentaro Torisawa, Chikara Hashimoto, Motoki Sano, Stijn De Saeger, and Kiyonori Ohtake. 2013. Why-question answering using intra-and inter-sentential causal relations. In *Association for Computational Linguistics*, volume 1, pages 1733–1743.

Mehwish Riaz and Roxana Girju. 2014. In-depth exploitation of noun and verb semantics to identify causation in verb-noun pairs. In *SIGDIAL*, pages 161–170.

Rebecca Sharp, Mihai Surdeanu, Peter Jansen, Peter Clark, and Michael Hammond. 2016. Creating causal embeddings for question answering with minimal supervision. *arXiv preprint arXiv:1609.08097*.

Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Shirui Pan, and Chengqi Zhang. 2018. Disan: Directional self-attention network for rnn/cnn-free language understanding. In *AAAI*.

Youngseo Son, Anneke Buffone, Joe Raso, Allegra Larche, Anthony Janocko, Kevin Zembroski, H Andrew Schwartz, and Lyle Ungar. 2017. Recognizing counterfactual thinking in social media texts. In *Association for Computational Linguistics*, pages 654–658.

Youngseo Son, Nipun Bayas, and H Andrew Schwartz. 2018. Causal explanation analysis on social media. In *Empirical Methods in Natural Language Processing*.

Radu Soricut and Daniel Marcu. 2003. Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 228–235.

Zhixing Tan, Mingxuan Wang, Jun Xie, Yidong Chen, and Xiaodong Shi. 2018. Deep semantic role labeling with self-attention. In *AAAI*.

Shikhar Vashishth, Manik Bhandari, Prateek Yadav, Piyush Rai, Chiranjib Bhattacharyya, and Partha Talukdar. 2019. Incorporating syntactic and semantic information in word embeddings using graph convolutional networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3308–3318, Florence, Italy, July. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *ArXiv*, abs/1706.03762.

Yizhong Wang, Sujian Li, Jingfeng Yang, Xu Sun, and Houfeng Wang. 2017. Tag-enhanced tree-structured neural networks for implicit discourse relation classification. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 496–505, Taipei, Taiwan, November. Asian Federation of Natural Language Processing.

Yuanbin Wu, Qi Zhang, Xuanjing Huang, and Lide Wu. 2009. Phrase dependency parsing for opinion mining. In *Empirical Methods in Natural Language Processing*, pages 1533–1541, Singapore, August. Association for Computational Linguistics.

Rui Xia and Zixiang Ding. 2019. Emotion-cause pair extraction: A new task to emotion analysis in texts. In *Association for Computational Linguistics*, pages 1003–1012, Florence, Italy, July. Association for Computational Linguistics.

Zhang and Xiaojun. 2014. Chengqing zong: Statistical natural language processing (second edition). *Machine Translation*, 28(2):155–158.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Empirical Methods in Natural Language Processing*, pages 35–45.

Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Empirical Methods in Natural Language Processing*, pages 2205–2215.

Named Entity Recognition with Context-Aware Dictionary Knowledge

Chuhan Wu[†], **Fangzhao Wu**[‡], **Tao Qi**[†], **Yongfeng Huang**[†]
[†]Department of Electronic Engineering & BNRist, Tsinghua University, Beijing 100084, China
[‡]Microsoft Research Asia, Beijing 100080, China
{wuchuhan15, wufangzhao, taoqi.qt}@gmail.com
yfhuang@tsinghua.edu.cn

Abstract

Named entity recognition (NER) is an important task in the natural language processing field. Existing NER methods heavily rely on labeled data for model training, and their performance on rare entities is usually unsatisfactory. Entity dictionaries can cover many entities including both popular ones and rare ones, and are useful for NER. However, many entity names are context-dependent and it is not optimal to directly apply dictionaries without considering the context. In this paper, we propose a neural NER approach which can exploit dictionary knowledge with contextual information. We propose to learn context-aware dictionary knowledge by modeling the interactions between the entities in dictionaries and their contexts via context-dictionary attention. In addition, we propose an auxiliary term classification task to predict the types of the matched entity names, and jointly train it with the NER model to fuse both contexts and dictionary knowledge into NER. Extensive experiments on the CoNLL-2003 benchmark dataset validate the effectiveness of our approach in exploiting entity dictionaries to improve the performance of various NER models.

1 Introduction

Named entity recognition (NER) aims to extract entity names from texts and classify them into several pre-defined categories, such as person, location and organization (Levow, 2006). It is an important task in natural language processing, and a prerequisite for many downstream applications such as entity linking (Derczynski et al., 2015) and relation extraction (Lin et al., 2016; Luo et al., 2018; Zeng et al., 2018). Thus, NER is a hot research topic. In this paper, we focus on the English NER task.

Many methods have been proposed for English NER, and most of them model this task as a word-level sequence labeling problem (Chiu and Nichols, 2016). For example, Ma and Hovy (2016) proposed a CNN-LSTM-CRF model for English NER. They used CNN to learn word representations from characters, LSTM to model the contexts of words, and CRF to decode labels. These existing NER methods usually rely on massive labeled data for model training, which is costly and time-consuming to annotate. When training data is scarce, their performance usually significantly declines (Peng et al., 2019). In addition, their performance on recognizing entities that rarely or do not appear in training data is usually unsatisfactory (Wang et al., 2019).

Fortunately, many large-scale entity dictionaries such as Wikipedia (Higashinaka et al., 2012) and Geonames[1] are off-the-shelf, and they can be easily derived from knowledge bases and webpages (Neelakantan and Collins, 2014). These entity dictionaries contain both popular and rare entity names, and can provide important information for NER models to identify these entity names. There are a few researches on incorporating entity dictionary into NER (Liu et al., 2019; Magnolini et al., 2019) and most of them are based on dictionary matching features. For example, Wang et al. (2019) proposed to combine token matching features with token embeddings and LSTM outputs. However, in many cases entities are context-dependent. For instance, in Table 1, the word "Jordan" can be a person name or a location name in different contexts. Thus, it is not optimal to directly apply entity dictionaries to NER without considering the contexts.

[1]https://www.geonames.org

Proceedings of the 19th China National Conference on Computational Linguistics, pages 915-926, Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

1	**Jordan** won against **Houston**. He will give talks in **Jordan** and **Houston**.
2	**Brown** is the former prime minister. **Brown** shoes are my favourate.

Red: PER
Orange: ORG
Blue: LOC

Table 1: Two examples of context-dependent entities.

In this paper, we propose a neural approach for named entity recognition with context-aware dictionary knowledge (CADK). We propose to exploit dictionary knowledge in a context-aware manner by modeling the relatedness between the entity names matched by entity dictionaries and their contexts. In addition, we propose an auxiliary term classification task to predict the types of the matched entity names in different contexts. Besides, we propose a unified framework to jointly train the NER model and the term classification model to incorporate entity dictionary knowledge and contextual information into the NER model. Extensive experiments show our approach can effectively exploit entity dictionaries to improve the performance of various NER models and reduce their dependence on labeled data.

2 Related Work

Named entity recognition is usually modeled as a sequence labeling problem (Wan et al., 2011). Many traditional NER methods are based on statistical sequence modeling methods, such as Hidden Markov Models (HMM) and Conditional Random Fields (CRF) (Cohen and Sarawagi, 2004; Ratinov and Roth, 2009; Passos et al., 2014; Arora et al., 2019). Usually, a core problem in these methods is how to build the feature vector for each word, and these features are traditionally constructed via manual feature engineering (Ratinov and Roth, 2009). For example, Ratinov and Roth (2009) used many features such as word n-grams, gazetteers and prediction histories as the word features. Passos et al. (2014) used features such as character n-grams, word types, capitalization pattern and lexicon matching features. They also incorporated lexicon embedding learned by skip-gram model to enhance the word representations. Designing these hand-crafted features usually needs a huge amount of domain knowledge. In addition, the feature vectors may be very sparse and their dimensions can be huge.

In recent years, many neural network based NER methods have been proposed (Collobert et al., 2011; Lample et al., 2016; Chiu and Nichols, 2016; Ma and Hovy, 2016; Peters et al., 2017; Li et al., 2017; Rei, 2017; Peters et al., 2018; Akbik et al., 2018; Lin and Lu, 2018; Clark et al., 2018; Chen et al., 2019; Zhu and Wang, 2019; Devlin et al., 2019). For example, Lample et al. (2016) proposed to use LSTM to learn the contextual representation of each token based on global context in sentences and use CRF for joint label decoding. Chiu and Nichols (2016) proposed to use CNN to learn word representations from original characters and then learn contextual word representation using Bi-LSTM. Ma and Hovy (2016) proposed to combine the CNN-LSTM framework with CRF for better performance. Peters et al. (2017) proposed a semi-supervised approach named TagLM for NER by pre-training a language model on a large corpus to provide contextualized word representations. Devlin et al. (2019) proposed a bidirectional pre-trained language model named BERT, which can empower downstream tasks like NER by using deep Transformers (Vaswani et al., 2017) to model contexts accurately. However, these neural network based methods heavily rely on labeled sentences to train NER models, which need heavy effort of manual annotation. In addition, their performance on recognizing entities which rarely or do not appear in labeled data is usually unsatisfactory (Wang et al., 2019).

There are several approaches on utilizing entity dictionaries for named entity recognition (Cohen and Sarawagi, 2004; Lin et al., 2007; Yu et al., 2008; Rocktäschel et al., 2013; Passos et al., 2014; Song et al., 2015; Wang et al., 2019; Liu et al., 2019). In traditional methods, dictionaries are often incorporated as additional features. For example, Cohen et al. (2004) proposed to extract dictionary features based on entity matching and similarities, and they incorporated these features into an HMM based model. There are also a few methods to incorporate dictionary knowledge into neural NER models (Chiu and Nichols, 2016; Wang et al., 2019; Liu et al., 2019). For example, Wang et al. (2019) proposed to incorporate dictionaries into neural NER model for detecting clinical entities. They manually designed several features

based on the matches with a clinical dictionary and then concatenated these features with the embedding vector as the input of the LSTM-CRF model. These methods rely on domain knowledge to design these dictionary based features, and these handcrafted features may not be optimal. Different from these methods, in our approach we introduce a term-level classification task to exploit the useful information in entity dictionary without manual feature engineering. We jointly train our model in both the NER and term classification tasks to enhance the performance of NER model in an end-to-end manner.

There are also a few methods that explore to incorporate dictionary knowledge into Chinese NER models in an end-to-end manner by using graph neural networks (Sui et al., 2019; Gui et al., 2019). For example, Sui et al. (2019) propose a character-based collaborative graph neural network to learn the representations of characters and words matched by dictionaries from three word-character graphs, i.e., a containing graph that describes the connection between characters and matched words, a transition graph that builds the connections between characters and the nearest contextual matched words, and a Lattice graph that connects each word with its boundary characters. However, these methods mainly model the interactions between matched entities and their local contexts, while ignore the relations between entities and their long-distance contexts. Different from these methods, our approach can model the interactions between the matched terms with the global contexts via entity-dictionary attention.

3 CADK Approach for NER

In this section, we introduce our NER approach with Context-Aware Dictionary Knowledge (CADK). The architecture of our approach is illustrated in Fig. 1. Our approach mainly contains five components, i.e., *text representation*, *term representation*, *context-dictionary attention*, *term classification* and *sequence tagging*. Next, we introduce the details of each module as follows.

3.1 Text Representation

The first module is a text representation model, which is used to learn the contextual representation of each word in an input text. It can be implemented by various neural text representation models, such as CNN (Zhu and Wang, 2019), LSTM (Huang et al., 2015) and GRU (Peters et al., 2017) or pre-trained language models like ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019). We denote the word sequence of the input text as $[w_1, w_2, ...w_N]$, where N is the number of words. The text representation model outputs a sequence that contains the contextual representation of each word, which is denoted as $\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N]$.

3.2 Term Representation

The second module is *term representation*, which is used to obtain the representations of the terms matched by the entity dictionaries. Usually, entity dictionaries contain both popular (e.g., America) and rare entity names (e.g., Chatham), and can help NER models recognize these entity names correctly. Thus, entity dictionaries have the potential to improve the performance of NER and reduce the dependence on labeled data. To incorporate useful information in entity dictionaries, we use them to match the input text and obtain a candidate list with M entity terms. We denote the word sequence of the i_{th} term as $[w_{i1}, w_{i2}, ...w_{iP}]$, where P represents the number of words in this term. In the *term representation* module, we first use a word embedding layer to convert the sequence of words in each term into a sequence of low-dimensional vectors. The word embedding parameters in this layer are shared with the *text representation* model. The word embedding sequence of the i_{th} term is denoted as $[\mathbf{w}_{i1}, \mathbf{w}_{i2}, ...\mathbf{w}_{iP}]$. Then, we apply a word-level Bi-GRU network to the word embedding sequence of each term to learn a hidden term representation. The GRU layer scans the word embedding sequence of each term in two directions, and combines the last hidden states in both directions as the representation of this term. For the i_{th} term, its representation is denoted as $\mathbf{t}_i$. We denote the sequence of the representations of the M matched terms as $\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_M]$.

3.3 Context-Dictionary Attention

The third module is *context-dictionary attention*. Many entity names are context-dependent. For example, in the sentence "Jordan is a famous NBA player", the word "Jordan" is in a person name, while it is

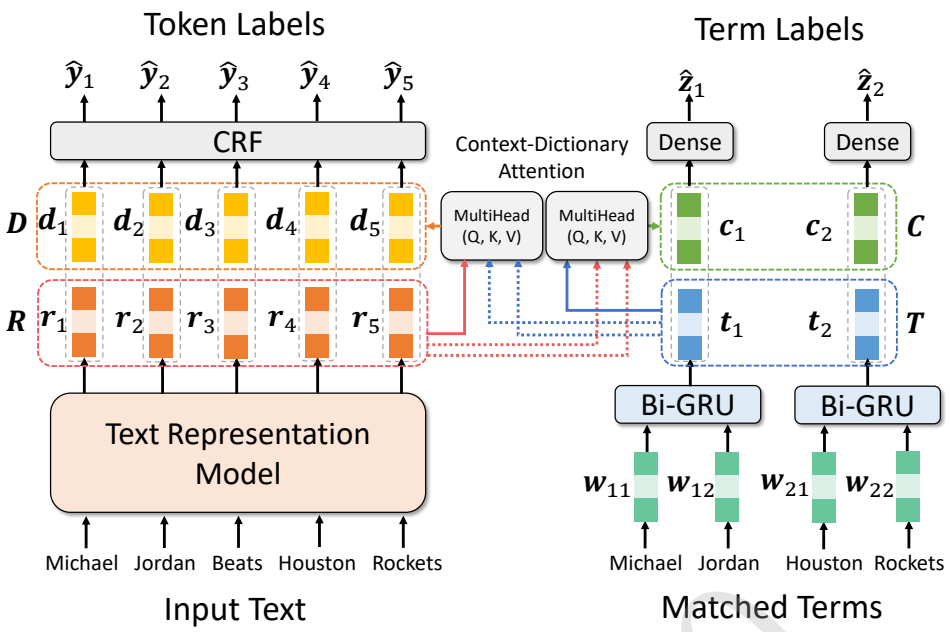


Figure 1: The architecture of our CADK approach.

also frequently used as a location name. Thus, we propose to incorporate dictionary knowledge in a context-aware manner by modeling the relationships between the matched entity terms and their contexts. It is used to model the interactions between terms matched by dictionaries with the contexts in sentences. Usually, entity names may interact with other words in the same text, and such interactions are important for recognizing these entities. For example, in the sentence "Jordan is a basketball player", the interaction between the entity "Jordan" and the word "player" is very informative for identifying the type of this entity is "person". In addition, an entity may interact with multiple words. For instance, in the sentence "He travels from Houston to Seattle", the interactions between the entity "Houston" and its contexts like "travels" and "Seattle" are useful clues for recognizing this entity. Motivated by these observations, we propose a context-dictionary attention module to model the interactions between the terms matched by dictionaries with all words in texts. The context-dictionary attention network takes both the sequences of word representations $\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N]$ and term representations $\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_M]$ (N and M are numbers of words and terms) as inputs, and outputs dictionary-aware representations of words in texts (denoted as $\mathbf{D}$) and context-aware representations of terms (denoted as $\mathbf{C}$). We use the multi-head productive attention mechanism (Vaswani et al., 2017) to model the interactions between terms and contexts. The dictionary-aware word representation sequence $\mathbf{D}$ is computed as follows:

$$\mathbf{D}^i = \text{Softmax}[\mathbf{W}_Q^i \mathbf{R}(\mathbf{W}_K^i \mathbf{T})^T](\mathbf{W}_V^i \mathbf{T}), \tag{1}$$

$$\mathbf{D} = \text{Concat}(\mathbf{D}^1, \mathbf{D}^2, ..., \mathbf{D}^h), \tag{2}$$

where $\mathbf{W}_Q^i$, $\mathbf{W}_K^i$, and $\mathbf{W}_V^i$ respectively stand for the parameters in the i_{th} head for transforming the query, key and value, h represents the number of parallel attention heads. The context-aware term representation sequence $\mathbf{C}$ is computed in a similar way as follows:

$$\mathbf{C}^i = \text{Softmax}[\mathbf{U}_Q^i \mathbf{T}(\mathbf{U}_K^i \mathbf{R})^T](\mathbf{U}_V^i \mathbf{R}), \tag{3}$$

$$\mathbf{C} = \text{Concat}(\mathbf{C}^1, \mathbf{C}^2, ..., \mathbf{C}^h), \tag{4}$$

where $\mathbf{U}_Q^i$, $\mathbf{U}_K^i$, and $\mathbf{U}_V^i$ are parameters. We concatenate $\mathbf{D}$ with the word representations $\mathbf{R}$, and $\mathbf{C}$ with the term representations $\mathbf{T}$, in a position-wise manner. In this way, entity dictionary with contextual information can be incorporated into a neural NER model.

3.4 Term Classification

The fourth module is *term classification*, which is used to classify the types of the terms matched by dictionaries based on the representations of terms and their interactions with the contexts. To fully exploit the useful information in the entity dictionary, we propose an auxiliary term classification task which predicts the type of the entity names matched by the entity dictionary. For example, in the sentence "Michael Jordan Beats Houston Rockets", if the terms "Michael Jordan" and "Houston Rockets" are matched by the dictionary, our model is required to classify the types of these terms in the context of this sentence. We use a dense layer with the softmax activation function to classify the type of each term as follows:

$$\hat{z}_i = \text{softmax}(\mathbf{U}[\mathbf{c}_i; \mathbf{t}_i] + \mathbf{v}), \tag{5}$$

where $\mathbf{U}$ and $\mathbf{v}$ are parameters, $\mathbf{c}_i$ is the context-aware representation of the i_{th} term, and $\hat{z}_i$ is the predicted type label of this term. The gold type label of the matched term can be derived from the token labels of the input sentence. For example, if the label sequence of a sentence is "O-BLOC-ELOC-O", we can know that the gold type of the entity in this sentence is "location". The loss function of the term classification task is the cross-entropy of the gold and the predicted labels of all terms, which is evaluated as follows:

$$\mathcal{L}_{Term} = -\sum_{i=1}^{S}\sum_{j=1}^{M}\sum_{k=1}^{K} \hat{z}_{ijk} \log(z_{ijk}), \tag{6}$$

where S is the number of sentences for model training, K is the number of entity categories, z_{ijk} and $\hat{z}_{ijk}$ are the gold and predicted type labels of the j_{th} term from the i_{th} sentence in the k_{th} category.

3.5 Sequence Tagging

The last module is *sequence tagging*. Usually the label at each position may have relatedness with the previous ones. For example, in the *BIOES* tagging scheme, the label "I-LOC" can only appear after "B-LOC" and "I-LOC". Thus, a CRF layer is usually employed to jointly decode the label sequence. Given a tag sequence $\mathbf{y} = [y_1, y_2, ..., y_N]$, the score of the tag sequence $\mathbf{y}$ in sentence $\mathbf{x}$ is defined as:

$$\mathbf{s}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{N} U_{i,y_i} + \sum_{i=1}^{N-1} A_{y_i,y_{i+1}}, \tag{7}$$

where U_{i,y_i} is the unary score of assigning the tag y_i to the i_{th} token, and $A_{y_i,y_{i+1}}$ represents the score of jumping from tag y_i to y_{i+1}. The unary score $\mathbf{U}_i$ is calculated as:

$$\mathbf{U}_i = \mathbf{W}_u \mathbf{h}_i + \mathbf{b}_u, \tag{8}$$

where $\mathbf{W}_u$ and $\mathbf{b}_u$ are trainable parameters. In CRF, the likelihood probability of the tag sequence $\mathbf{y}$ is formulated as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{e^{\mathbf{s}(\mathbf{x},\mathbf{y})}}{\sum_{\mathbf{y}' \in \mathcal{Y}_\mathbf{x}} e^{\mathbf{s}(\mathbf{x},\mathbf{y}')}}, \tag{9}$$

where $\mathcal{Y}_\mathbf{x}$ represents the set of all possible tag sequences. Then the loss function of the NER task is evaluated as:

$$\mathcal{L}_{NER} = -\sum_{y_i \in \mathcal{S}} \log(p(\mathbf{y}_i|\mathbf{x}_i)), \tag{10}$$

where $\mathcal{S}$ denotes the training dataset, and $\mathbf{y}_i$ is the ground-truth tag sequence of sentence $\mathbf{x}_i$.

To incorporate the useful information in entity dictionary into NER models more effectively, we propose a unified framework based on multi-task learning to jointly train our model in both NER and term classification tasks. The final loss function is the weighted summation of the NER and term classification loss, which is formulated as follows:

$$\mathcal{L} = (1 - \lambda)\mathcal{L}_{NER} + \lambda \mathcal{L}_{Term}, \tag{11}$$

where $\mathcal{L}_{NER}$ is the loss of CRF model, $\lambda \in [0, 1]$ is a coefficient to control the relative importance of the term classification task.

Model	10%			25%			100%		
	P	R	F	P	R	F	P	R	F
LSTM-CRF	84.23	88.22	86.18	87.75	87.86	87.81	90.75	90.14	90.36
LSTM-CRF+Feature	84.90	89.02	86.91	88.33	88.40	88.37	91.14	90.18	90.66
LSTM-CRF+GNN	85.54	88.74	87.11	88.53	88.56	88.54	90.99	90.51	90.75
LSTM-CRF+CADK	85.94	89.27	87.58	89.34	88.72	89.03	91.58	90.81	91.19
TagLM	85.63	88.70	87.14	88.64	89.05	88.85	92.01	91.40	91.71
TagLM+Feature	85.77	90.14	87.90	89.44	89.25	89.35	92.41	91.64	92.02
TagLM+GNN	86.27	90.02	88.10	89.79	89.34	89.56	92.62	91.91	92.26
TagLM+CADK	86.56	90.68	88.57	89.98	90.14	90.06	93.03	92.33	92.68
ELMo	85.34	89.24	87.25	88.76	89.13	88.95	92.42	92.23	92.30
ELMo+Feature	86.01	89.96	87.94	89.51	89.39	89.45	92.73	92.19	92.46
ELMo+GNN	86.71	89.97	88.31	89.70	89.65	89.68	92.92	92.28	92.60
ELMo+CADK	87.09	90.36	88.70	90.40	89.82	90.11	93.49	92.57	93.03
BERT	84.76	87.87	86.29	87.91	88.11	88.01	91.89	91.23	91.49
BERT+Feature	85.48	88.86	87.14	88.60	88.43	88.51	91.99	91.41	91.70
BERT+GNN	85.73	88.72	87.20	88.65	88.90	88.77	92.12	91.64	91.88
BERT+CADK	86.20	89.30	87.72	89.19	89.32	89.26	92.40	92.00	92.20

Table 2: Performance of different NER methods under different ratios of training data. P, R, F respectively stand for precision, recall and Fscore.

4 Experiments

4.1 Dataset and Experimental Settings

Our experiments were conducted on the CoNLL-2003 dataset (Tjong Kim Sang and De Meulder, 2003), which is a widely used benchmark dataset for NER. This dataset contains four different types of named entities, i.e., locations, persons, organizations, and miscellaneous entities that do not belong in the three previous categories. Following previous works (Ratinov and Roth, 2009), we used the BIOES labeling scheme. In our experiments, we used an entity dictionary provided by (Higashinaka et al., 2012), which is derived from the WikiPedia database. This dictionary contains 297,073,139 entity names. The coefficient λ in Eq. (11) was 0.4. We used Adam (Kingma and Ba, 2014) with gradient norms clipped at 5.0 as the optimizer for model training, and the learning rate was 0.001. The batch size was 64. These hyperparameters were tuned on the validation set. Each experiment was repeated 5 times independently, and the average performance in terms of precision, recall and Fscore were reported.

4.2 Comparison with Baseline Methods

To verify the effectiveness of the proposed *CADK* method, we compare several popular models and their variants using different methods for incorporating entity dictionaries. The methods to be compared including: (1) LSTM-CRF (Huang et al., 2015), a neural NER method that uses LSTM to learn word representations and CRF to decode labels; (2) TagLM (Peters et al., 2017), a neural NER model which uses GRU networks and a pre-trained language model to learn word representations, and uses CRF to decode labels; (3) ELMo (Devlin et al., 2019), a pre-trained language model with bidirectional deep LSTM network. We apply an LSTM-CRF network based on the contextualized word embeddings generated by the ELMo model; (4) BERT (Devlin et al., 2019), a pre-trained language model with bidirectional transformers. We fine-tune the BERT-base version in the NER task; The methods for incorporating entity dictionaries including: (a) Feature (Wang et al., 2019), incorporating entity dictionaries using feature engineering. We combines the dictionary matching features with the hidden representations learned by the aforementioned methods; (b) GNN (Sui et al., 2019), using graph neural networks to incorporate entity dictionary knowledge; (c) CADK, our proposed method with context-aware dictionary knowledge.

We randomly sampled different ratios (i.e., 10%, 25% and 100%) of samples from the data for model

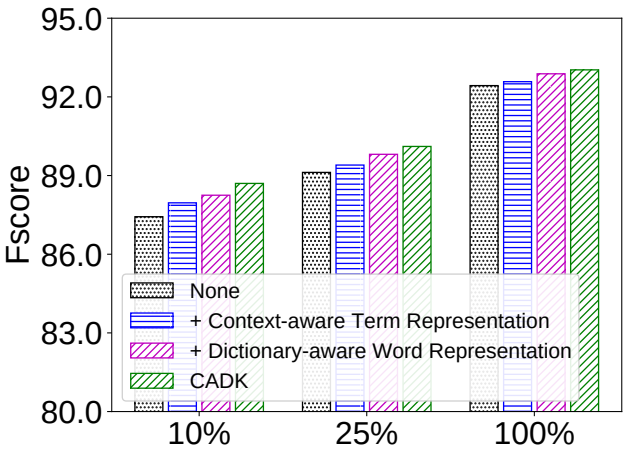


Figure 2: Effectiveness of the context-dictionary attention module.

training to evaluate these methods under different amounts of labeled data. The results are summarized in Table 2.[2] From Table 2, we find that when the training data is scarce, the performance of the methods without dictionary knowledge declines significantly. This is probably because these neural network based methods are data-intensive and require a large amount of labeled data for model training. When training data is scarce, many entities in the test set are unseen in the training data, making it difficult for existing NER methods to recognize them. Compared with methods without dictionaries, the methods that consider dictionary knowledge achieve better performance, and their advantage is larger when training data is more scarce. This is probably because incorporating dictionary knowledge can help recognize unseen or rare entities more effectively, which can reduce the dependency on labeled data. In addition, compared with the methods using dictionary matching features, the methods that can model the contexts of matched entities (*GNN* and *CADK*) perform better. This is probably because manually crafted features may be not optimal to utilize entity dictionaries, and the contexts of the matched entity names in different texts are not considered. Besides, our *CADK* method is better than the *GNN* method in exploiting dictionary knowledge for NER. Different from the *GNN* method that can only model the local contexts of matched entity names, in our approach we use the context-dictionary attention model to capture the global contexts of the matched terms, and we jointly train our model in both NER and term classification tasks to incorporate dictionary knowledge in a unified framework. Thus, our method can exploit dictionary information more accurately to improve neural NER model.

4.3 Effectiveness of Context-Dictionary Attention

In this section, we conduct several ablation studies to validate the effectiveness of the context-dictionary attention module in our *CADK* method. Since it mainly aims to generate the dictionary-aware word representation and the context-aware term representation, we compare the performance of *ELMo-CADK* under different ratios of training data by removing one or both of them. The results are shown in Fig. 3. According to the results, we find that the dictionary-aware word representation can effectively improve the performance of our approach. This is because the dictionary-aware word representation encodes the information of the entities matched by dictionaries, which is helpful for recognizing them more accurately. In addition, incorporating the context-aware term representation can also improve the model performance. This is because many entities are context-dependent, and modeling their relations with the contexts is beneficial for NER. These results show the effectiveness of context-dictionary attention in injecting context-aware dictionary knowledge into neural NER models.

[2]The performance of BERT is surprisingly unsatisfactory though we used the officially released model and carefully tuned hyperparameters.

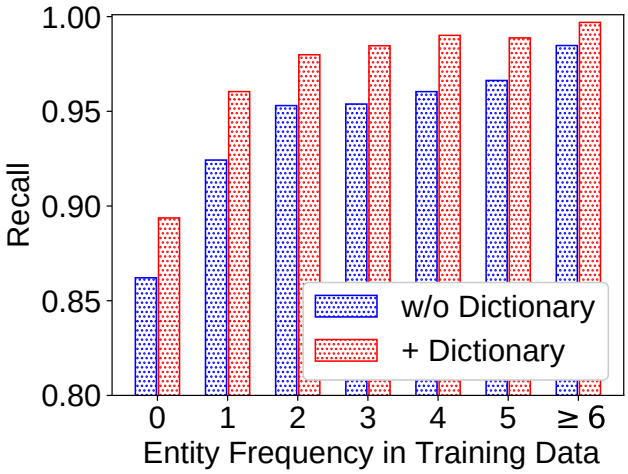


Figure 3: Recall of the entities in the test set with different frequencies in the training data.

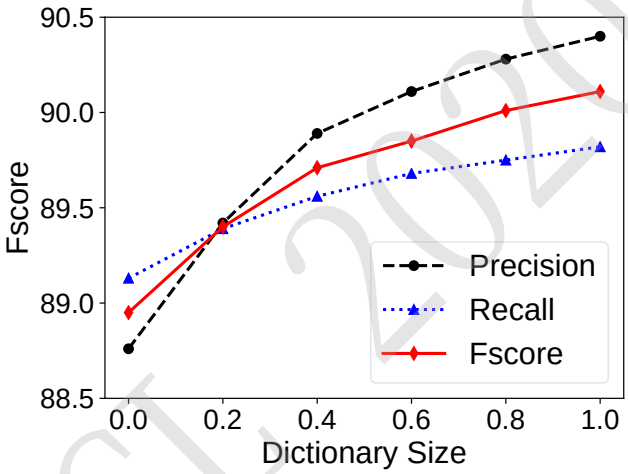


Figure 4: Model performance under different dictionary size.

4.4 Performance on Rare Entities

In this section, we explore the influence of incorporating dictionary knowledge on recognizing the entities rarely appearing in the training data. We evaluate the recall of the entities in the test set with different appearance times in the training data. We conduct experiments under 25% of training data and the results of the *ELMo+CADK* model are shown in Fig. 3, which reveals two findings. First, the performance on entities that do not or rarely appear in the training data is much lower than recognizing common entities. This result shows that rare entities are more difficult to recognize. Second, our approach can effectively improve the performance on entities that rarely appear in the training data. This is because our approach can utilize dictionary knowledge to help neural NER model recognize these rare entities more accurately.

4.5 Influence on Dictionary Size

In this section, we study the influence of the size of entity dictionaries. We randomly sampled different ratios of entities from the dictionary for entity matching and compare the performance of the *ELMo-CADK* model under 25% of training data. The results are shown in Fig. 4. We find that the model performance consistently improves when the dictionary size grows. This is because a larger dictionary usually has better entity coverage, and our approach can exploit richer information from the entity dictionary to help recognize entities more accurately.

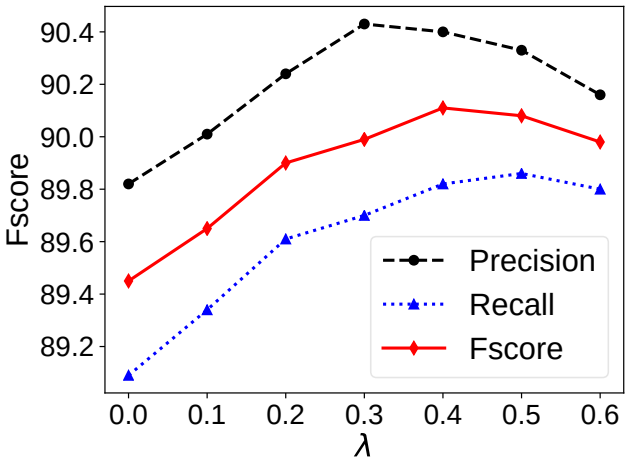


Figure 5: The performance of our approach with different λ values under different ratios of training data.

4.6 Influence of Hyper-parameters

In this section we explore the influence of an important hyper-parameter on our approach, i.e., λ in Eq. (11), which is used to control the relative importance of the term classification loss. The experimental results on λ using the *ELMo-CADK* model with 25% of training data are shown in Fig. 5. According to Fig. 5, the performance of our approach improves when λ increases. However, when λ becomes too large the performance declines. This is because when λ is too small, the useful information in the term classification task is not fully exploited. Thus, the performance is sub-optimal. When λ goes too large, the auxiliary task is dominant and the NER task is not fully respected. Thus, the performance of our approach is also sub-optimal. These results lead to a moderate selection of λ (e.g., 0.4).

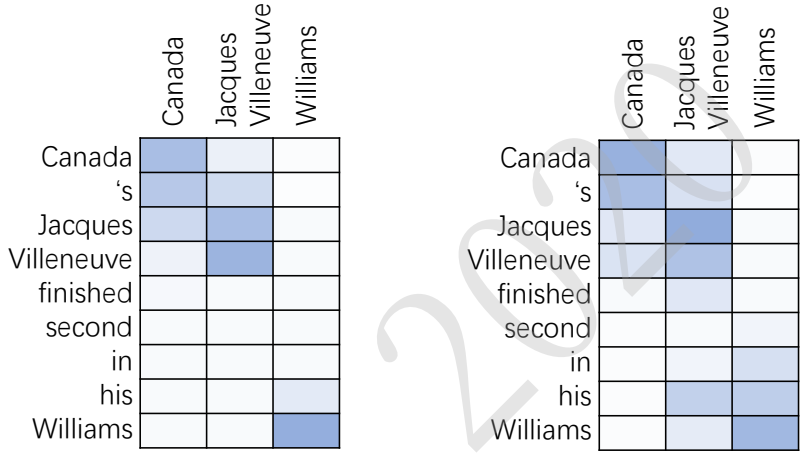
4.7 Case Study

In this section, we conducted several case studies to better understand our approach in incorporating dictionary knowledge in a context-aware manner. Several representative samples are shown in Table 3. This experiment is conducted using 10% of training data. According to Table 3, incorporating entity dictionaries can help a NER model better recognize rare entities. For example, "Partizan" is a name of a football team, which only appears once in the training set. The basic NER model recognized it as a person name, while the approaches using dictionaries can make correct predictions. Our approach can also correctly recognize the context-dependent entities which the basic model and the model based on dictionary features fail to recognize. For example, the entity "Florida" is recognized as a location by *ELMo* and *ELMo+Feature*, since it is usually used as a location name. Our approach can recognize this entity correctly based on its contexts. These results show that our approach can effectively exploit the useful information in entity dictionaries with contextual information.

Next, we visualize the attention weights in the context-dictionary attention to better understand the interactions between contexts and matched terms. The visualization results are shown in Fig. 6. According tothe results, we can see that our approach can effectively model the interactions between entity terms and contexts. For example, in Fig. 6(a), the interaction between the word "Jacques" and the term "Jacques Villeneuve" is highlighted, which is important for identifying the word "Jacques" belongs to an entity name. In addition, in Fig. 6(b), the interaction between the term "Jacques Villeneuve" and the word "his" is also highlighted, which is an important clue for inferring the type of this entity is "person". These results indicate that our approach can effectively capture the relationships between the entity names matched by dictionaries and their contexts to learn context-aware dictionary knowledge.

Example	Method	NER result
1	ELMo	Third one-day match : December 8, in Karachi.
	ELMo+Feature	Third one-day match : December 8, in Karachi.
	ELMo+CADK	Third one-day match : December 8, in Karachi.
2	ELMo	Partizan - Dejan Koturovic 21
	ELMo+Feature	Partizan - Dejan Koturovic 21
	ELMo+CADK	Partizan - Dejan Koturovic 21
3	ELMo	Bolesy (Florida manager John Boles) told me ...
	ELMo+Feature	Bolesy (Florida manager John Boles) told me ...
	ELMo+CADK	Bolesy (Florida manager John Boles) told me ...

Table 3: Several named entity recognition examples. Red, orange, and blue words represent the predicted person, location and organization entities respectively.



(a) Using word representations as attention query and term representations as key.

(b) Using term representations as attention query and word representations as key.

Figure 6: Visualization of the attention weights in the context-dictionary attention network.

5 Conclusion

In this paper we propose a neural NER approach which can incorporate entity dictionaries with contextual information. In our approach, we propose a context-dictionary attention network to model the interactions between entity names matched by dictionaries and their contexts in texts. In addition, we propose an auxiliary term classification task to classify the types of the terms matched by dictionaries based on contexts, and we jointly train our model in both NER and term classification tasks to incorporate the information of entity dictionaries and contexts into NER. Extensive experiments on the CoNLL-2003 benchmark dataset show that our approach can effectively improve the performance of NER especially when training data is insufficient.

Acknowledgments

This work was supported by the National Key Research and Development Program of China under Grant number 2018YFC1604002, the National Natural Science Foundation of China under Grant numbers U1936208, U1936216, U1836204, and U1705261.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *COLING*, pages 1638–1649.

Ravneet Arora, Chen-Tse Tsai, Ketevan Tsereteli, Prabhanjan Kambadur, and Yi Yang. 2019. A semi-Markov structured support vector machine model for high-precision named entity recognition. In *ACL*.

Hui Chen, Zijia Lin, Guiguang Ding, Jianguang Lou, Yusen Zhang, and Borje Karlsson. 2019. Grn: Gated relation network to enhance convolutional neural network for named entity recognition. In *AAAI*.

Jason Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional lstm-cnns. *TACL*, 4(1):357–370.

Kevin Clark, Minh-Thang Luong, Christopher D Manning, and Quoc Le. 2018. Semi-supervised sequence modeling with cross-view training. In *EMNLP*, pages 1914–1925.

William W Cohen and Sunita Sarawagi. 2004. Exploiting dictionaries in named entity extraction: combining semi-markov extraction processes and data integration methods. In *KDD*, pages 89–98. ACM.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *JMLR*, 12(Aug):2493–2537.

Leon Derczynski, Diana Maynard, Giuseppe Rizzo, Marieke van Erp, Genevieve Gorrell, Raphaël Troncy, Johann Petrak, and Kalina Bontcheva. 2015. Analysis of named entity recognition and linking for tweets. *Information Processing & Management*, 51(2):32–49.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186.

Tao Gui, Yicheng Zou, Qi Zhang, Minlong Peng, Jinlan Fu, Zhongyu Wei, and Xuan-Jing Huang. 2019. A lexicon-based graph neural network for chinese ner. In *EMNLP-IJCNLP*, pages 1039–1049.

Ryuichiro Higashinaka, Kugatsu Sadamitsu, Kuniko Saito, Toshiro Makino, and Yoshihiro Matsuo. 2012. Creating an extended named entity dictionary from wikipedia. In *COLING*, pages 1163–1178.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *NAACL-HLT*, pages 260–270.

Gina-Anne Levow. 2006. The third international chinese language processing bakeoff: Word segmentation and named entity recognition. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 108–117.

Peng-Hsuan Li, Ruo-Ping Dong, Yu-Siang Wang, Ju-Chieh Chou, and Wei-Yun Ma. 2017. Leveraging linguistic structures for named entity recognition with bidirectional recursive neural networks. In *EMNLP*, pages 2664–2669.

Bill Yuchen Lin and Wei Lu. 2018. Neural adaptation layers for cross-domain named entity recognition. In *EMNLP*, pages 2012–2022.

Hongfei Lin, Yanpeng Li, and Zhihao Yang. 2007. Incorporating dictionary features into conditional random fields for gene/protein named entity recognition. In *PAKDD*, pages 162–173. Springer.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *ACL*, pages 2124–2133.

Tianyu Liu, Jin-ge Yao, and Chin-Yew Lin. 2019. Towards improving neural named entity recognition with gazetteers. In *ACL*, pages 5301–5307.

Xiong Luo, Wenwen Zhou, Weiping Wang, Yueqin Zhu, and Jing Deng. 2018. Attention-based relation extraction with bidirectional gated recurrent unit and highway network in the analysis of geological data. *IEEE Access*, 6:5705–5715.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *ACL*, volume 1, pages 1064–1074.

Simone Magnolini, Valerio Piccioni, Vevake Balaraman, Marco Guerini, and Bernardo Magnini. 2019. How to use gazetteers for entity recognition with neural models. In *Proceedings of the 5th Workshop on Semantic Deep Learning*, pages 40–49.

Arvind Neelakantan and Michael Collins. 2014. Learning dictionaries for named entity recognition using minimal supervision. In *EACL*, pages 452–461.

Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon infused phrase embeddings for named entity resolution. *CoNLL-2014*, page 78.

Minlong Peng, Xiaoyu Xing, Qi Zhang, Jinlan Fu, and Xuanjing Huang. 2019. Distantly supervised named entity recognition using positive-unlabeled learning. In *ACL*, pages 2409–2419.

Matthew Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models. In *ACL*, volume 1, pages 1756–1765.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL-HLT*, pages 2227–2237.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *CoNLL*, pages 147–155.

Marek Rei. 2017. Semi-supervised multitask learning for sequence labeling. In *ACL*, pages 2121–2130.

Tim Rocktäschel, Torsten Huber, Michael Weidlich, and Ulf Leser. 2013. Wbi-ner: The impact of domain-specific features on the performance of identifying and classifying mentions of drugs. In *SemEval 2013*, volume 2, pages 356–363.

Min Song, Hwanjo Yu, and Wook-Shin Han. 2015. Developing a hybrid dictionary-based bio-entity recognition technique. *BMC medical informatics and decision making*, 15(1):S9.

Dianbo Sui, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2019. Leverage lexical knowledge for chinese named entity recognition via collaborative graph network. In *EMNLP-IJCNLP*, pages 3821–3831.

Erik F Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *NAACL-HLT*, pages 142–147.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.

Xiaojun Wan, Liang Zong, Xiaojiang Huang, Tengfei Ma, Houping Jia, Yuqian Wu, and Jianguo Xiao. 2011. Named entity recognition in chinese news comments on the web. In *IJCNLP*, pages 856–864.

Qi Wang, Yangming Zhou, Tong Ruan, Daqi Gao, Yuhang Xia, and Ping He. 2019. Incorporating dictionaries into deep neural networks for the chinese clinical named entity recognition. *Journal of biomedical informatics*, 92:103133.

Xiaofeng Yu, Wai Lam, Shing-Kit Chan, Yiu Kei Wu, and Bo Chen. 2008. Chinese ner using crfs and logic for the fourth sighan bakeoff. In *Proceedings of the Sixth SIGHAN Workshop on Chinese Language Processing*.

Weixin Zeng, Jiuyang Tang, and Xiang Zhao. 2018. Entity linking on chinese microblogs via deep neural network. *IEEE Access*, 6:25908–25920.

Yuying Zhu and Guoxin Wang. 2019. Can-ner: Convolutional attention network for chinese named entity recognition. In *NAACL-HLT*, pages 3384–3393.

Chinese Named Entity Recognition via Adaptive Multi-pass Memory Network with Hierarchical Tagging Mechanism

Pengfei Cao[1,2], **Yubo Chen**[1,2], **Kang Liu**[1,2] and **Jun Zhao**[1,2]
[1] National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing, 100190, China
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences,
Beijing, 100049, China
{pengfei.cao, yubo.chen, kliu, jzhao}@nlpr.ia.ac.cn

Abstract

Named entity recognition (NER) aims to identify text spans that mention named entities and classify them into pre-defined categories. For Chinese NER task, most of the existing methods are character-based sequence labeling models and achieve great success. However, these methods usually ignore lexical knowledge, which leads to false prediction of entity boundaries. Moreover, these methods have difficulties in capturing tag dependencies. In this paper, we propose an **A**daptive **M**ulti-pass **M**emory **N**etwork with **H**ierarchical **T**agging Mechanism (**AMMNHT**) to address all above problems. Specifically, to reduce the errors of predicting entity boundaries, we propose an adaptive multi-pass memory network to exploit lexical knowledge. In addition, we propose a hierarchical tagging layer to learn tag dependencies. Experimental results on three widely used Chinese NER datasets demonstrate that our proposed model outperforms other state-of-the-art methods.

1 Introduction

The task of named entity recognition (NER) is to recognize the named entities from a plain text and classify them into pre-defined types. NER is a fundamental and preliminary task in natural language processing (NLP) area and is beneficial for many downstream NLP tasks such as relation extraction (Bunescu and Mooney, 2005), event extraction (Chen et al., 2015) and question answering (Yahya et al., 2013). In recent years, numerous methods have been carefully studied for NER task, including Conditional Random Fields (CRFs) (Lafferty et al., 2001) and Support Vector Machines (SVMs) (Isozaki and Kazawa, 2002). Currently, with the development of deep learning methods, neural networks have been introduced for the NER task. In particular, sequence labeling neural network models have achieved state-of-the-art performance (Lample et al., 2016; Zhang and Yang, 2018).

Though sequence labeling neural network methods have achieved great success for Chinese NER task, some challenging issues still have not been well addressed. One significant drawback is that previous methods usually **fail to correctly predict entity boundaries**. To conduct a quantitative analysis, we perform a BiLSTM+CRF model proposed by Huang et al. (2015), which is the most representative Chinese NER sequence labeling system, on WeiboNER dataset (Peng and Dredze, 2015; He and Sun, 2016), OntoNotes 4 dataset (Weischedel et al., 2011) and MSRA dataset (Levow, 2006). The F1 scores are 55.84%, 63.17% and 89.13%, respectively. We do a further analysis and find that the errors of predicting entity boundaries are particularly serious. The average proportion of predicting entity boundaries errors is 82% on these three datasets. For example, the character-based BiLSTM+CRF model fails to predict the entity boundaries of "北海道 (Hokkaido)" in Figure 1. To reduce the errors of predicting entity boundaries, some works (Peng and Dredze, 2016; Cao et al., 2018) try to jointly perform Chinese NER with Chinese word segmentation (CWS) for using word boundaries information. However, the joint model requires additional annotated training data for CWS task.

Fortunately, existing lexicons can provide information on word boundaries and we refer to the information as lexical knowledge. In addition, the cost of obtaining lexicon is low and almost all fields have their lexicons, such as biomedical, social science fields and so on. Recently, Zhang and Yang (2018) propose a lattice LSTM model capable of leveraging lexicon for Chinese NER. Though effective, the lattice LSTM

Proceedings of the 19th China National Conference on Computational Linguistics, pages 927-938, Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

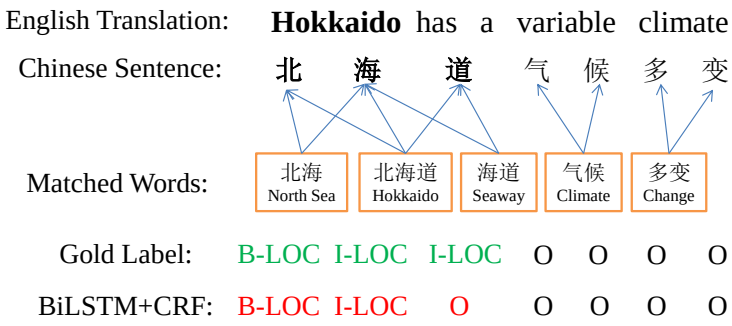


Figure 1: An example of Chinese NER with wrong entity boundaries using the BiLSTM+CRF model. It also shows the matched words for each character.

cannot exploit all matched words. When the candidate labeled character is within a matched word (i.e. the character is not the first or the last character of the matched word), the lattice model cannot explicitly and directly exploit the matched word. For example, for the candidate labeled character "海 (Sea)", it can match "北海 (North Sea)", "海道 (Seaway)" and "北海道 (Hokkaido)" in lexicon according to its context. When exploiting the matched words for character "海 (Sea)", the lattice model only considers "北海 (North Sea)" and "海道 (Seaway)", ignoring "北海道 (Hokkaido)" which can help determine that the character "海 (Sea)" is the middle of an entity rather than beginning or ending. Moreover, the lattice model only processes the matched words once, when learning the lexical knowledge for a character. However, it needs more reasoning passes on the matched words to better learn lexical knowledge in complex sentences intuitively. Take the sentence "南京市长江大桥 (Nanjing Yangtze River Bridge)" for example, it is more complicated than the sentence in Figure 1 because it is prone to be misunderstood as "南京市长/江大桥 (The mayor of Nanjing is Jiang Daqiao)". Thus, it needs more reasoning passes to learn the lexical knowledge for recognizing the entity "长江大桥 (Yangtze River Bridge)" than the entity "北海道 (Hokkaido)" in Figure 1. However, if the reasoning passes are too many, the performance will decrease in word sense disambiguation task (Luo et al., 2018). We argue that the problem also exists in Chinese NER task. Hence, how to exploit all matched words and perform flexible multi-pass reasoning according to the complexity of sentences should be well investigated.

Another issue is that most of the existing methods **cannot efficiently capture tag dependencies**. In sequence labeling neural network models, CRF is usually used as a decoding layer. Although the CRF decoder has achieved improvements, the transition matrix in CRF layer only learns the neighboring tag dependencies, which are typically first order dependencies (Zhang et al., 2018). Thus, CRF cannot well handle long-distance tag dependency problems. For example, in the sentence "耐克拥有比李宁更大的市场 (Nike has a larger market than Li Ning)", the tag of "李宁 (Li Ning)" is dependent on the tag of "耐克 (Nike)", as they should be the same entity type. Since "李宁 (Li Ning)" can be a person or an organization, it is more difficult to predict the tag of "李宁 (Li Ning)" than "耐克 (Nike)". However, it is easy to tag "耐克 (Nike)" as an organization. If we capture the dependencies between "李宁 (Li Ning)" and "耐克 (Nike)", we will have ample evidence to tag "李宁 (Li Ning)" as an organization. To address the issue, Zhang et al. (2018) exploit the LSTM as decoder instead of CRF. However, the unidirectional LSTM decoder only leverages the past labels and ignores the future labels. In another sentence "李宁努力地同耐克竞争 (Li Ning strives to compete with Nike)", when predicting the tag of "李宁 (Li Ning)", the future tag of "耐克 (Nike)" can help us to determine the tag of "李宁 (Li Ning)". Thus, how to capture bidirectional (past and future) tag dependencies in the whole sentence is another challenging problem.

In this paper, we propose an **A**daptive **M**ulti-pass **M**emory **N**etwork with **H**ierarchical **T**agging Mechanism (**AMMNHT**) to address the aforementioned problems. To exploit all matched words and perform multi-pass reasoning across matched words for a character, memory network (Sukhbaatar et al., 2015) can be utilized for Chinese NER. However, conventional memory network follows pre-defined passes to perform multi-pass reasoning and cannot perform adaptive and proper deliberation passes according to

the change of input sentence. We utilize reinforcement learning (Sutton et al., 1998) to adaptively determine the deliberation passes of memory network according to the complexity of sentences. Although we do not have explicit supervision for the reasoning passes of the memory network, we can obtain long-term feedback (or *reward*) from the final prediction, which inspires us to utilize reinforcement learning techniques. To capture bidirectional tag dependencies in the whole sentence, we propose a hierarchical tagging mechanism for Chinese NER task.

In summary, the contributions of this paper are listed as follows:

- We propose a novel framework to integrate lexical knowledge from the lexicon for Chinese NER task, which can explicitly exploit all matched words and adaptively choose suitable reasoning passes for each sentence. To our best knowledge, this is the first work to automatically determine the reasoning passes of memory network via reinforcement learning techniques.

- We propose a hierarchical tagging mechanism for Chinese NER to capture bidirectional tag dependencies in the whole sentence. To our knowledge, this is the first work to devise the hierarchical tagging mechanism for Chinese NER task.

- Experiments on three widely used Chinese NER datasets show that our proposed model outperforms previous state-of-the-art methods.

2 Related Work

In recent years, the NER task has attracted much research attention. Many methods have been proposed to perform the task. Early studies on NER often exploit CRFs (Lafferty et al., 2001) and SVMs (Isozaki and Kazawa, 2002). These methods rely heavily on feature engineering. However, the designed features may be not appropriate for the task, which can lead to error propagation problem. Currently, neural network methods have been introduced into NER task and achieved state-of-the-art performance (Lample et al., 2016). Huang et al. (2015) use the bidirectional long short term memory (BiLSTM) for feature extraction and the CRF for decoding. The model is trained via the end-to-end paradigm. After that, the BiLSTM+CRF model is usually exploited as the baseline model for NER task. Ma and Hovy (2016) use a character convolutional neural network (CNN) to represent spelling characteristic. Then the charcter representation vector is concatenated with word embedding as the input of the LSTM. Peters et al. (2017) leverage a character language model to enhance the input of the model.

For Chinese NER, character-based methods have been the dominant approaches (Lu et al., 2016; Dong et al., 2016). These methods only focus on character sequence information, ignoring word boundaries information, which can cause errors of predicting entity boundaries. Thus, how to better exploit lexical knowledge has received much research attention. Word segmentation information is used as extra features for Chinese NER task (Peng and Dredze, 2015; He and Sun, 2016). Peng and Dredze (2016) and Cao et al. (2018) propose a joint model for Chinese NER, which is jointly trained with CWS task. Zhang and Yang (2018) investigate a lattice LSTM to encode a sequence of input characters as well as words that match a lexicon. However, the lattice model cannot exploit all matched words and only processes the matched words once. Recently, graph-based models have been proposed for Chinese NER (Gui et al., 2019; Sui et al., 2019). Based on the lattice structure, Sui et al. (2019) propose a graph neural network to encode word information.

Tag dependencies is also a challenging problem, but few attention has been paid to tackling the problem. Zhang et al. (2018) leverages LSTM as decoder for sequence labeling task. However, the unidirectional LSTM decoder only exploits the past predicted tags information, ignoring the future un-predicted tags. Hence, we propose a hierarchical tagging mechanism to capture bidirectional tag dependencies in the whole sentence. To our best knowledge, we are the first to introduce the hierarchical tagging mechanism to Chinese NER task. Moreover, to better capture the dependencies between tags, we try different hierarchical tagging mechanism.

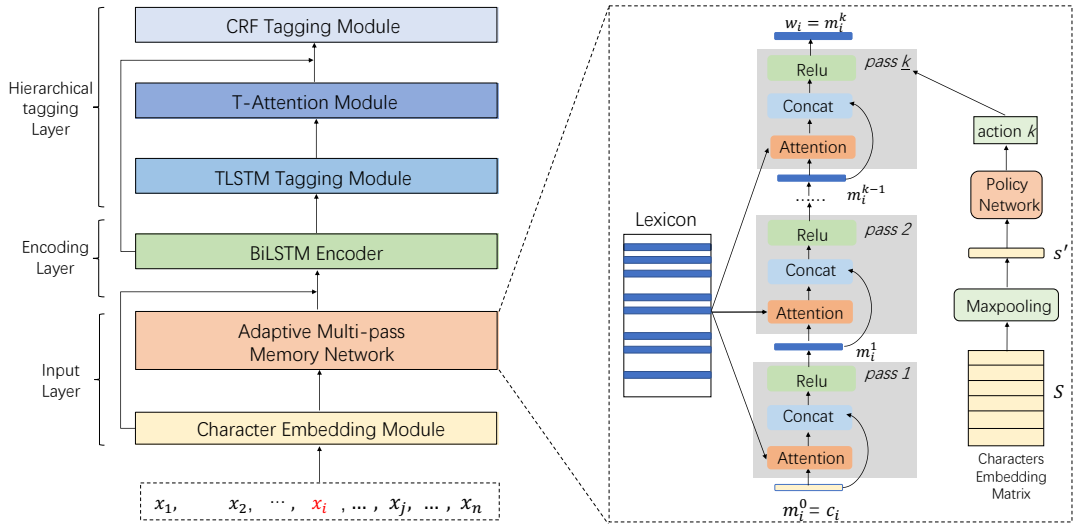


Figure 2: The architecture of our proposed adaptive multi-pass memory network with hierarchical tagging mechanism. The right part is the adaptive multi-pass memory network (AMMN). For each character, the lexical knowledge (i.e., w_i in the figure) is obtained via the AMMN. We concatenate the character embeddings and lexical knowledge as the input of the encoding layer. In this figure, we use the character x_i as an example to illustrate the process.

3 Method

The architecture of our proposed model is shown in Figure 2. The proposed model consists of three components: input layer, BiLSTM encoding layer and hierarchical tagging layer. In the following sections, we will describe the details of our proposed model.

3.1 Input Layer

The inputs of our proposed model are character embeddings and lexical knowledge, which are obtained via character embedding module and adaptive multi-pass memory network, respectively.

Character Embedding Module Similar to other methods using neural networks, the first step of our proposed model is to map discrete language symbols to distributed representations. Formally, given a Chinese sentence $s = \{x_1, x_2, \ldots, x_n\}$, each character x_i is represented by looking up embedding vector from a pre-trained character embedding table:

$$c_i = E^c(x_i) \tag{1}$$

where E^c is a pre-trained character embedding table and $c_i \in \mathbb{R}^{d_c}$. We obtain the characters embedding matrix, denoted as $S = \{c_1, c_2, \ldots, c_n\}$.

Adaptive Multi-pass Memory Network The adaptive multi-pass memory network has three inputs: the candidate character embedding c_i as the initial query vector, the characters embedding matrix S and the matched words $\{w_{i1}, w_{i2}, \ldots, w_{iN_i}\}$ of the character x_i as the external memory, where N_i is the number of matched words. Since a candidate character may match multiple words in a lexicon and one-pass attention calculation may not accurately learn lexical knowledge, memory network is exploited to perform a deep reasoning process to highlight the correct lexical knowledge. After each pass, we need to update the query vector for the next pass. Therefore, the memory network contains two phases: ***attention calculation*** and ***update mechanism***.

Attention Calculation: During each pass, the query vector is the output of the former pass. We use attention to model the relationship between the query vector and the matched words. At pass k, the

attention calculation can be formulated as follows:

$$e_{it}^k = w_{it}^T m_i^{k-1}$$
$$\alpha_{it}^k = \frac{\exp(e_{it}^k)}{\sum_{j=1}^{N_i} \exp(e_{ij}^k)} \tag{2}$$

where m_i^{k-1} denotes the output of pass $k-1$. We treat the candidate character embedding c_i as m_i^0.

Update Mechanism: After calculating the attention, we can obtain the memory state at the current pass:

$$u_i^k = \sum\nolimits_{t=1}^{N_i} \alpha_{it}^k w_{it} \tag{3}$$

We update the query vector by taking the former pass output and memory state of current pass into consideration for the next pass:

$$m_i^k = \text{Relu}(W_m[m_i^{k-1} : u_i^k] + b_m) \tag{4}$$

where $[:]$ is the concatenation operation. $W_m \in \mathbb{R}^{d_w \times 2d_w}$ and $b_m \in \mathbb{R}^{d_w}$ are trainable parameters. We use the output of the last pass as the lexical knowledge of the character x_i, denoted as w_i.

Empirically, different reasoning passes may obtain different performances (Luo et al., 2018). We assume that less reasoning passes are enough to tackle simple sentences than complicated sentences. However, conventional memory network cannot perform adaptive and proper deliberation passes according to the complexity of the input sentence. Therefore, we utilize reinforcement learning to automatically control the reasoning passes of the memory network. We will introduce ***state***, ***action*** and ***reward*** as follows:

State: We use the sentence embedding s' as the state. After getting the characters embedding matrix S, we perform the max-pooling operation and treat the result as the sentence embedding:

$$s' = \text{Maxpooling}(S) \tag{5}$$

Action: We regard the reasoning pass as the action $a \in \{1, 2, \ldots, N\}$, where N is the maximal pass. We sample the value of a by a policy network $\pi_\Theta(a|s')$, which can be formulated as follows:

$$\pi_\Theta(a|s') = \text{Softmax}(W_p s' + b_p) \tag{6}$$

where $W_p \in \mathbb{R}^{N \times d_c}$ and $b_p \in \mathbb{R}^N$ are trainable parameters. $\Theta = \{W_p, b_p\}$.

Reward: We can obtain a terminal reward after finishing the final prediction. In this work, we use the F1 score of each sentence as the reward r.

Given T training instances, the objective function of policy network is defined as :

$$J_1 = \sum\nolimits_{i=1}^{T} \log\pi_\Theta(a_{(i)}|s'_{(i)}) r_{(i)} \tag{7}$$

where $a_{(i)}$, $s'_{(i)}$ and $r_{(i)}$ are the action, state and reward of the training instance i, respectively. We use the policy gradient method to learn the parameter set Θ.

3.2 BiLSTM Encoding Layer

After obtaining character embeddings and lexical knowledge, we concatenate them as the input of the encoding layer. Long short term memory (LSTM) is a variant of recurrent neural network (RNN), which is designed to address the gradient vanishing and exploding problems in RNN via introducing gate mechanism and memory cell. In order to incorporate information from both sides of sequence, we use BiLSTM to extract features. The hidden state of BiLSTM can be defined as follows:

$$h_i = [\overrightarrow{h}_i : \overleftarrow{h}_i] \tag{8}$$

where $\overrightarrow{h}_i \in \mathbb{R}^{d_h}$ and $\overleftarrow{h}_i \in \mathbb{R}^{d_h}$ are the hidden states at position i of the forward and backward LSTM, respectively.

3.3 Hierarchical Tagging Layer

In the hierarchical tagging layer, we exploit the LSTM as the first tagging module named as TLSTM and the CRF as the second tagging module.

The First Tagging Module: TLSTM When detecting the tag of character x_i, the inputs of the first tagging module are: h_i from the BiLSTM encoding layer, former hidden state $\hat{h}_{i-1}$, and former predicted tag vector $\widehat{T}_{i-1}$. Formally, the TLSTM can be written precisely as follows:

$$\begin{aligned}
\begin{bmatrix} i_i \\ o_i \\ f_i \\ \tilde{c}_i \end{bmatrix} &= \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} \left(W_d^T \begin{bmatrix} h_i \\ \hat{h}_{i-1} \\ \widehat{T}_{i-1} \end{bmatrix} + b_d \right) \\
\hat{c}_i &= \hat{c}_{i-1} \odot f_i + \tilde{c}_i \odot i_i \\
\hat{h}_i &= o_i \odot \tanh(\hat{c}_i) \\
\widehat{T}_i &= W_{td} \hat{h}_i + b_{td}
\end{aligned} \tag{9}$$

where i, f, o are the input gate, forget gate and output gate, respectively. $\widehat{T}$ is the predicted tagging vector.

Tagging Attention Module: T-Attention Tagging attention aims to dynamically leverage the hidden states and preliminary predictions of the TLSTM. $\widehat{H} = \{\hat{h}_1, \hat{h}_2, \ldots, \hat{h}_n\}$ and $T_{raw} = \{\widehat{T}_1, \widehat{T}_2, \ldots, \widehat{T}_n\}$ denote the hidden states and preliminary predictions of the TLSTM, respectively. The attention is expressed as follows:

$$\begin{aligned}
\hat{h}_{di} &= [\hat{h}_i : \widehat{T}_i] \\
m_i &= u_d^T \tanh(W_{da} \hat{h}_{di} + b_{da}) \\
\alpha_i &= \frac{\exp(m_i)}{\sum_{j=1}^{n} \exp(m_j)} \\
r_i &= \tanh(\sum\nolimits_{j=1}^{n} \alpha_j \hat{h}_{dj})
\end{aligned} \tag{10}$$

where $u_d \in \mathbb{R}^{d_{da}}$ is the context vector, which is randomly initialized and learned during the training process (Yang et al., 2016b). r_i denotes the representation of the hidden states and preliminary predictions of the TLSTM.

The Second Tagging Module: CRF $H = \{h_1, h_2, \ldots, h_n\}$ and $R = \{r_1, r_2, \ldots, r_n\}$ denote the outputs of BiLSTM encoding layer and tagging attention module, respectively, which are concatenated as the input of the CRF module, denoted as $H_c = \{h_{c1}, h_{c2}, \ldots, h_{cn}\}$.

Given a sentence $s = \{x_1, x_2, \ldots, x_n\}$ with a final predicted tag sequence $y = \{y_1, y_2, \ldots, y_n\}$, the CRF tagging process is formalized as follows:

$$\begin{aligned}
o_i &= W_o h_{ci} + b_o \\
s(s, y) &= \sum\nolimits_{i=1}^{n} (o_{i,y_i} + T_{y_{i-1}, y_i}) \\
y^* &= \arg\max_{y \in Y_s} s(s, y)
\end{aligned} \tag{11}$$

where o_{i,y_i} is the score of the y_i-th tag of the character x_i. T denotes the transition matrix which defines the scores of two successive labels. Y_s represents all candidate tag sequences for given sentence s. We use the Viterbi algorithm to get the final best-scoring tag sequence y^*.

3.4 Training

The probability of the ground-truth tag sequence $\bar{y}$ can be computed by:

$$p(\bar{y}|s) = \frac{\exp(s(s, \bar{y}))}{\sum_{\tilde{y} \in Y_s} \exp(s(s, \tilde{y}))} \tag{12}$$

Dataset	# Train sentence	# Dev sentence	# Test sentence
MSRA	41.4k	4.6k	4.0k
OntoNotes 4	22.7k	3.9k	2.7k
WeiboNER	1.4k	0.27k	0.27k

Table 1: Statistics of the datasets.

Given a set of manually labeled training data $\{s^{(i)}, \bar{y}^{(i)}\}|_{i=1}^{T}$, the objective function of the tagging layer can be defined as follows:

$$J_2 = \sum_{i=1}^{T} \log p(\bar{y}^{(i)}|s^{(i)}) \tag{13}$$

The objective function of the whole model is listed as follows:

$$J = \lambda J_1 + J_2 \tag{14}$$

As the adaptive multi-pass memory network and hierarchical tagging layer are correlated mutually, we train them jointly. We pre-train the model before the joint training process starts using the objective function J_2. Then, we jointly train the model using the objective function J.

4 Experiments

4.1 Datasets

We evaluate our proposed model on three widely used datasets, including MSRA (Levow, 2006), OntoNotes 4 (Weischedel et al., 2011) and WeiboNER (Peng and Dredze, 2015; He and Sun, 2016). The MSRA dataset contains three entity types (person, location and organization). The OntoNotes 4 dataset annotates 18 named entity types. In this work, we use the four most common named entity types (person, location, organization and geo-political), as same as previous studies (Che et al., 2013; Zhang and Yang, 2018). The WeiboNER dataset is annotated with four entity types (person, location, organization and geo-political), including named entities and nominal mentions.

For MSRA dataset, we use the same data split as Dong et al. (2016). Since MSRA dataset does not have development set, we sample 10% data of training set as development set. For OntoNotes 4 dataset, we take the same data split as Che et al. (2013) and Zhang and Yang (2018). For WeiboNER dataset, we use the same training, development and testing splits as Peng and Dredze (2015) and He and Sun (2016). The details of the datasets are shown in Table 1.

4.2 Evaluation Metrics and Experimental Settings

For evaluation metrics, we use the Micro averaged Precision (P), Recall (R) and F1 score as metrics in our experiments, as the same as previous works (Che et al., 2013; Zhang and Yang, 2018), which are calculated per-span.

Hyper-parameters tuning is made through adjustments according to the performance on the development sets. The dimension of character embedding d_c is 100. The size of word embedding d_w is 50. The hidden size of LSTM d_h is set to 300. The dropout rate is 0.3. The λ is set to 0.1. Adam (Kingma and Ba, 2014) is used for optimization, with an initial learning rate of 0.001. The character embeddings used in this work are pre-trained on Chinese Wikipedia corpus by using word2vec toolkit (Mikolov et al., 2013). We use the same lexicon as Zhang and Yang (2018).

4.3 Compared with State-of-the-art Methods

4.3.1 Evaluation on MSRA

We compare our proposed model with previous methods on MSRA dataset. The results are listed in Table 2[1]. Zhang et al. (2006) leverage rich handcrafted features for Chinese NER. The model gives very competitive performance. Dong et al. (2016) incorporate radical features into neural LSTM+CRF model,

[1] * in Table 2, 3 and 4 denotes that a model exploits additional labeled data.

Models	P(%)	R(%)	F1(%)
Chen et al. (2006)	91.22	81.71	86.20
Zhou et al. (2006)	88.94	84.20	86.51
Zhang et al. (2006)*	92.20	90.18	91.18
Zhou et al. (2013)	91.86	88.75	90.28
Dong et al. (2016)	91.28	90.62	90.95
Zhang and Yang (2018)	93.57	92.79	93.18
Cao et al. (2018)	91.73	89.58	90.64
AMMNHT	**93.62**	**92.96**	**93.29**

Table 2: Experimental results on MSRA dataset.

Models	P(%)	R(%)	F1(%)
Che et al. (2013)*	77.71	72.51	75.02
Wang et al. (2013)*	76.43	72.32	74.32
Yang et al. (2016a)	65.59	71.84	68.57
Yang et al. (2016a)*	72.98	80.15	76.40
Zhang and Yang (2018)	76.35	71.56	73.88
AMMNHT	**76.51**	**71.70**	**74.03**

Table 3: Experimental results on OntoNotes 4 dataset. The first and second blocks list word-based methods and character-based method, respectively.

achieving the F1 score of 90.95%. Cao et al. (2018) achieve competitive performance via adversarial transfer learning method. We can observe that our proposed model gets significant improvements over previous state-of-the-art methods. For example, compared with the latest model (Cao et al., 2018) which uses additional CWS training data, our proposed method improves the F1 score from 90.64% to 93.29%. Moreover, compared with Zhang and Yang (2018), our model also greatly improves the performance. We also perform a t-test ($p < 0.01$), which indicates that our method outperforms all of the compared methods.

4.3.2 Evaluation on OntoNotes

We evaluate our proposed model on OntoNotes 4 dataset. Table 3 lists the results of our proposed model and previous state-of-the-art methods. In the first two blocks, we give the performance of word-based and character-based methods for Chinese NER, respectively. Based on the gold segmentation, Che et al. (2013) propose an integer linear program based inference algorithm with bilingual constraints for NER. The model gives a 75.02% F1 score. With gold word segmentation, the word-based models achieve better performance than the character-based model. This demonstrates that word boundaries information is useful for Chinese NER task. Compared with the character-based method (Zhang and Yang, 2018), our model improves the F1 score from 73.88% to 74.03%. Compared with the word-based method (Wang et al., 2013), our model also achieves better performance. The great improvements over previous state-of-the-art methods demonstrate the effectiveness of our proposed model.

4.3.3 Evaluation on WeiboNER

We compare our proposed model with the latest models on WeiboNER dataset. The experimental results are shown in Table 4, where NE, NM and Overall denote F1 scores for named entities, nominal entities and both, respectively. Peng and Dredze (2016) propose a model that jointly performs Chinese NER and CWS task, which achieves better results than Peng and Dredze (2015) for named entity, nominal mention and overall. Recently, Zhang and Yang (2018) propose a lattice LSTM model to exploit word sequence information. The model gives a 58.79% F1 score on overall performance. It can be observed that our proposed model achieves great improvements compared with previous methods. For example, compared

Models	NE	NM	Overall
Peng and Dredze (2015)	51.96	61.05	56.05
Peng and Dredze (2016)*	**55.28**	62.97	58.99
He and Sun (2016)	50.60	59.32	54.82
He and Sun (2017)*	54.50	62.17	58.23
Zhang and Yang (2018)	53.04	62.25	58.79
Cao et al. (2018)	54.34	57.35	58.70
AMMNHT	54.09	62.43	**59.04**

Table 4: F1 scores (%) on WeiboNER dataset.

Models	MSRA	OntoNotes	WeiboNER
BiLSTM+CRF	89.13	63.17	55.84
BiLSTM+CRF+AMMN	92.40	73.11	58.65
BiLSTM+HT	90.53	64.14	56.55
AMMNHT	**93.29**	**74.03**	**59.04**

Table 5: F1 score (%) of AMMNHT and its simplified models on MSRA, OntoNotes 4 and WeiboNER datasets, respectively.

with the lattice LSTM model, our proposed model improves the F1 score from 53.04% to 54.09% for named entity. It proves the effectiveness of our proposed model.

4.4 Ablation Experiment

To investigate the effectiveness of adaptive multi-pass memory network and hierarchical tagging mechanism, we conduct the ablation studies. The baseline and simplified models of the proposed model are detailed as follows: (1) **BiLSTM+CRF**: The model is exploited as the strong baseline in our experiment. (2) **BiLSTM+CRF+AMMN**: The model integrates lexical knowledge from a lexicon via adaptive multi-pass memory network. (3) **BiLSTM+HT**: The model exploits the BiLSTM to extract features and uses the hierarchical tagging layer to predict labels.

From the results listed in Table 5, we have several important observations as follows:

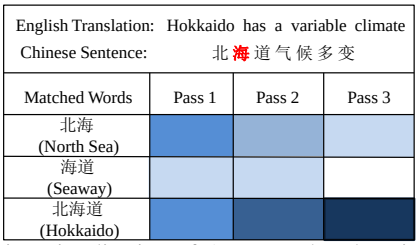
- **Effectiveness of Adaptive Multi-pass Memory Network.** We observe that the BiLSTM+CRF+AMMN model outperforms the BiLSTM+CRF on these three datasets. For example, compared with the baseline, it improves the F1 score from 89.13% to 92.40% on MSRA dataset. Compared the AMMNHT with BiLSTM+HT, we can find similar phenomenon. The great improvements demonstrate the effectiveness of the adaptive multi-pass memory network.

- **Effectiveness of Hierarchical Tagging Mechanism.** Compared with the BiLSTM+CRF, the BiLSTM+HT model improves the performance, achieving 1.40% improvements of F1 score on MSRA dataset. Moreover, the AMMNHT also outperforms the BiLSTM+CRF+AMMN. The great improvements indicate the hierarchical tagging mechanism is very effective for Chinese NER task.

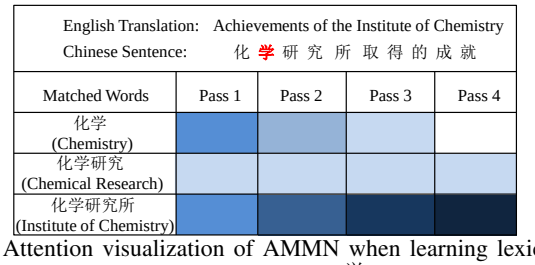
- **Effectiveness of Adaptive Multi-pass Memory Network and Hierarchical Tagging Mechanism.** We observe that the proposed model AMMNHT achieves better performance than its simplified models on the three datasets. For example, compared with BiLSTM+CRF, the AMMNHT model improves the F1 score from 89.13% to 93.29% on MSRA dataset. It indicates that simultaneously exploiting the adaptive multi-pass memory network and hierarchical tagging mechanism is also very effective.

4.5 Adaptive Multiple Passes Analysis

To better illustrate the influence of multiple passes and adaptive multi-pass memory network, we give the results of fixed multiple passes and adaptive multi-pass memory network in Table 6. The results



(a) Attention visualization of AMMN when learning lexical knowledge for the candidate character "海 (sea)".



(b) Attention visualization of AMMN when learning lexical knowledge for the candidate character '学 (subject)".

Figure 3: Two examples of attention weights in adaptive multi-pass memory network. The reasoning passes are 3 and 4, respectively. Darker colors mean that the attention weight is higher.

Pass	MSRA	OntoNotes	WeiboNER
1	92.64	72.87	58.52
2	92.96	73.50	58.83
3	93.14	73.77	58.74
4	93.12	73.85	58.34
5	93.03	73.46	58.13
Adaptive	**93.29**	**74.03**	**59.04**

Table 6: F1 score (%) of different passes from 1 to 5 and adaptive passes on the test sets. It shows suitable reasoning passes of memory network can boost the performance.

show that multiple passes operation performs better than one pass. The reason is that multiple passes reasoning can help to highlight the most appropriate matched words. The cases in Figure 3 show that the deep deliberation can recognize the correct lexical knowledge by enlarging the attention gap between correct matched words and incorrect ones. When the number of passes is too large, the performance stops increasing or even decreases due to over-fitting. In contrast to the fixed multiple passes memory network, the adaptive multi-pass memory network has 0.21% improvements of F1 score on the WeiboNER dataset. Furthermore, the two examples in Figure 3 show that adaptive multi-pass memory network can choose suitable reasoning passes according to the complexity of the input sentence, which also demonstrates the effectiveness of the adaptive multi-pass memory network.

5 Conclusion

In this paper, we propose an adaptive multi-pass memory network to incorporate lexical knowledge from a lexicon for Chinese NER task which can adaptively choose suitable reasoning passes according to the complexity of each sentence. Besides, we devise a hierarchical tagging layer to capture tag dependencies in the whole sentence. The adaptive memory network and hierarchical tagging mechanism can be easily applied to similar tasks involving multi-pass reasoning and decoding process, such as knowledge base question answering and machine translation. Experimental results on three widely used datasets demonstrate that our proposed model outperforms previous state-of-the-art methods.

Acknowledgments

This work is supported by the Natural Key R&D Program of China (No.2017YFB1002101), the National Natural Science Foundation of China (No.61533018, No.61922085 , No.61976211, No.61806201) and the Key Research Program of the Chinese Academy of Sciences (Grant NO. ZDBS-SSW-JSC006). This work isalso supported by Beijing Academy of Artificial Intelligence (BAAI2019QN0301), the CCF-Tencent Open Research Fund and independent research project of National Laboratory of Pattern Recognition.

References

Razvan Bunescu and Raymond Mooney. 2005. A shortest path dependency kernel for relation extraction. In *Proceedings of EMNLP*, pages 724–731.

Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2018. Adversarial transfer learning for chinese named entity recognition with self-attention mechanism. In *Proceedings of EMNLP*.

Wanxiang Che, Mengqiu Wang, Christopher D Manning, and Ting Liu. 2013. Named entity recognition with bilingual constraints. In *Proceedings of NAACL-HLT*, pages 52–62.

Aitao Chen, Fuchun Peng, Roy Shan, and Gordon Sun. 2006. Chinese named entity recognition with conditional probabilistic models. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 213–216.

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of ACL*, pages 167–176.

Chuanhai Dong, Jiajun Zhang, Chengqing Zong, Masanori Hattori, and Hui Di. 2016. Character-based lstm-crf with radical-level features for chinese named entity recognition. In *Natural Language Understanding and Intelligent Applications*, pages 239–250.

Tao Gui, Yicheng Zou, Qi Zhang, Minlong Peng, Jinlan Fu, Zhongyu Wei, and Xuan-Jing Huang. 2019. A lexicon-based graph neural network for chinese ner. In *EMNLP-IJCNLP*.

Hangfeng He and Xu Sun. 2016. F-score driven max margin neural network for named entity recognition in chinese social media. *arXiv preprint arXiv:1611.04234*.

Hangfeng He and Xu Sun. 2017. A unified model for cross-domain and semi-supervised named entity recognition in chinese social media. In *Proceedings of AAAI*.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Hideki Isozaki and Hideto Kazawa. 2002. Efficient support vector classifiers for named entity recognition. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of NAACL-HLT*, pages 260–270.

Gina-Anne Levow. 2006. The third international chinese language processing bakeoff: Word segmentation and named entity recognition. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 108–117.

Yanan Lu, Yue Zhang, and Dong-Hong Ji. 2016. Multi-prototype chinese character embedding. In *Proceedings of LREC*.

Fuli Luo, Tianyu Liu, Qiaolin Xia, Baobao Chang, and Zhifang Sui. 2018. Incorporating glosses into neural word sense disambiguation. In *Proceedings of ACL*, pages 2473–2482.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of ACL*, pages 1064–1074.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Nanyun Peng and Mark Dredze. 2015. Named entity recognition for chinese social media with jointly trained embeddings. In *Proceedings of EMNLP*, pages 548–554.

Nanyun Peng and Mark Dredze. 2016. Improving named entity recognition for chinese social media with word segmentation representation learning. In *Proceedings of ACL*, pages 149–155.

Matthew Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models. In *Proceedings of ACL*, pages 1756–1765.

Dianbo Sui, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2019. Leverage lexical knowledge for chinese named entity recognition via collaborative graph network. In *EMNLP-IJCNLP*.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Proceedings of NeurIPS*, pages 2440–2448.

Richard S Sutton, Andrew G Barto, et al. 1998. *Introduction to reinforcement learning*. MIT press Cambridge.

Mengqiu Wang, Wanxiang Che, and Christopher D Manning. 2013. Effective bilingual constraints for semi-supervised learning of named entity recognizers. In *Proceedings of AAAI*.

Ralph Weischedel, Sameer Pradhan, Lance Ramshaw, Martha Palmer, Nianwen Xue, Mitchell Marcus, Ann Taylor, Craig Greenberg, Eduard Hovy, Robert Belvin, et al. 2011. Ontonotes release 4.0. *LDC2011T03, Philadelphia, Penn.: Linguistic Data Consortium*.

Mohamed Yahya, Klaus Berberich, Shady Elbassuoni, and Gerhard Weikum. 2013. Robust question answering over the web of linked data. In *Proceedings of CIKM*, pages 1107–1116.

Jie Yang, Zhiyang Teng, Meishan Zhang, and Yue Zhang. 2016a. Combining discrete and neural features for sequence labeling. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 140–154.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016b. Hierarchical attention networks for document classification. In *Proceedings of NAACL-HLT*, pages 1480–1489.

Yue Zhang and Jie Yang. 2018. Chinese ner using lattice lstm. In *Proceedings of ACL*, pages 1554–1564.

Suxiang Zhang, Ying Qin, Juan Wen, and Xiaojie Wang. 2006. Word segmentation and named entity recognition for sighan bakeoff3. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 158–161.

Yuan Zhang, Hongshen Chen, Yihong Zhao, Qun Liu, and Dawei Yin. 2018. Learning tag dependencies for sequence tagging. In *Proceedings of IJCAI*, pages 4581–4587.

Junsheng Zhou, Liang He, Xinyu Dai, and Jiajun Chen. 2006. Chinese named entity recognition with a multi-phase model. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*.

Junsheng Zhou, Weiguang Qu, and Fen Zhang. 2013. Chinese named entity recognition via joint identification and categorization. *Chinese journal of electronics*.

A Practice of Tourism Knowledge Graph Construction based on Heterogeneous Information

Dinghe Xiao
Hainan Sino-intelligent-Info Technology Ltd.
xiaodinghe@e-zzx.com

Nannan Wang
Beijing University of Posts and Telecommunications
wangnannan@bupt.edu.cn

Jiangang Yu
Hainan Sino-intelligent-Info Technology Ltd.
yujiangang@e-zzx.com

Chunhong Zhang✉
Beijing University of Posts and Telecommunications
zhangch@bupt.edu.cn

Jiaqi Wu
Hainan Sino-intelligent-Info Technology Ltd.
wujiaqi@e-zzx.com

Abstract

The increasing amount of semi-structured and unstructured data on tourism websites brings a need for information extraction (IE) so as to construct a Tourism-domain Knowledge Graph (TKG), which is helpful to manage tourism information and develop downstream applications such as tourism search engine, recommendation and Q & A. However, the existing TKG is deficient, and there are few open methods to promote the construction and widespread application of TKG. In this paper, we present a systematic framework to build a TKG for Hainan, collecting data from popular tourism websites and structuring it into triples. The data is multi-source and heterogeneous, which raises a great challenge for processing it. So we develop two pipelines of processing methods for semi-structured data and unstructured data respectively. We refer to tourism InfoBox for semi-structured knowledge extraction and leverage deep learning algorithms to extract entities and relations from unstructured travel notes, which are colloquial and high-noise, and then we fuse the extracted knowledge from two sources. Finally, a TKG with 13 entity types and 46 relation types is established, which totally contains 34,079 entities and 441,371 triples. The systematic procedure proposed by this paper can construct a TKG from tourism websites, which can further applied to many scenarios and provide detailed reference for the construction of other domain-specific knowledge graphs.

1 Introduction

Tourism has become increasingly popular in people's daily life. Before people set out to travel, they often need to make clear the travel guides and matters needing attention for their destinations. Nowadays, with the development of the Internet, many tourism websites have appeared and provide a variety of travel information, such as attractions, tickets, bus routes, travel guides, etc. However, there may be some errors in the miscellaneous information on the tourism websites, and information on different tourism websites may be inconsistent. As shown in screenshots of Sina Micro-Blog users' blogs in Figure 1, there are still tourists who are worried about making travel strategies despite rich information on all kinds of tourism-related search engines. How to collect and integrate valuable tourism knowledge on websites is a very important issue.

Recently, Knowledge Graph (KG) has received much attention and research interest in industry and academia. The KG utilizes a set of subject-predicate-object triplets to represent the diverse entities and their relations in real-world scenes, which are respectively represented as nodes and edges in the graph. The KG is a graph-based large-scale knowledge representation and integration method, which has been applied in various scenarios such as enterprise (Miao et al., 2015), medical (Rotmensch et al., 2017) and industry (Zhao et al., 2019). Naturally, we consider applying KG in the field of Tourism to integrate and organize relevant knowledge, so as to provide tourists with easier tools to develop travel strategies.

At present, several General Knowledge Graphs (GKGs) have been built both in Chinese and English (Auer et al., 2007; Suchanek et al., 2007; Niu et al., 2011; Xu et al., 2017). The Domain-specific

Proceedings of the 19th China National Conference on Computational Linguistics, pages 939-949,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

Figure 1: Screenshots of Sina Micro-Blog users' blogs. In the blogs, people with tourism intentions complain that it is difficult to formulate travel strategies.

Knowledge Graph (DKG) in which the stored knowledge is limited to a certain field has also been implemented and put into use in many domains (Zhao et al., 2018). However, Tourism-domain Knowledge Graph (TKG) is still deficient, which undoubtedly hinders the development of intelligent tourism system. In this paper, we propose a systematic framework to construct a TKG under the background of Hainan Tourism. We combine the semi-structured knowledge crawled from the encyclopedia pages of tourism websites with the unstructured travel notes shared by tourists on the websites as the data source. Because of the lack of sufficient high-quality data and the difficulty of language processing, constructing a Chinese-based TKG still faces several challenges as follows:

Travel notes are colloquial and high-noise. The writing style of travel notes is often arbitrary, and tourists tend to add various pictures, emoticons and special characters to travel notes, which will introduce much noise for unstructured data.

The Lack of datasets dedicated to tourism. There is a serious lack of normative datasets in the tourism field, which are basis of model training.

Are the general algorithms suitable for tourism? Entity extraction and relation extraction are the key steps in knowledge graph construction. Most of the existing algorithms for these two tasks are tested on the general datasets, we need to verify whether these algorithms are suitable for the tourism field.

How to integrate data from different sources? Data from different sources inevitably have some overlaps and ambiguities, which should be eliminated in the KG.

Facing this challenges, we put forward corresponding methods to deal with them. In detail, the contributions of our work are highlighted as follows:

- A specific method of collecting and processing tourism-domain data is described, and labeled datasets for information extraction in the field of tourism is constructed;
- The most suitable models for our tourism data are identified, and a tourism-domain knowledge graph is finally constructed.
- Experience in constructing the TKG can provide detailed reference for the construction of other domain-specific knowledge graphs.

2 Related Work

In recent years, the KG has been applied in many fields to complete knowledge storage, query, recommendation and other functions. In the tourism scene, experts and scholars have also begun to explore the application value of knowledge graphs. DBtravel (Calleja et al., 2018) is an English tourism-oriented knowledge graph generated from the collaborative travel site Wikitravel. A Chinese TKG was also constructed by (Zhang et al., 2019), which extracted tourism-related knowledge from existing Chinese

general knowledge graph such as zhishi.me (Niu et al., 2011) and CN-DBpedia (Xu et al., 2017). Unlike their Chinese TKG, we extensively obtain data and extract knowledge from popular tourism websites. In this way, the completeness of our knowledge graph does not depend on the existing knowledge graph, but on the amount of data we acquire. To construct the TKG, we need to extract triples form all kinds of information resources. The conversion process from semi-structured data to structured data is more standardized and has fewer errors, but semi-structured data often cannot contain all the knowledge. With the development of Natural Language Processing (NLP), more and more knowledge graphs are constructed based on unstructured corpus, using named entity recognition (NER) and relation extraction (RE) technologies.

As a hot research direction in the field of NLP, many Chinese NER models have been proposed over the years. The purpose of NER task is to identify mentions of named entities from text and match them to pre-defined categories. As a classic branch of NER models, the dictionary-based methods recognize named entities by constructing a dictionary and matching text with it. For example, CMEL (Meng et al., 2014) built a synonym dictionary for Chinese entities from Microblog and adopts improved SVM to get textual similarity for entity disambiguation. Another line of related work is to apply traditional machine learning techniques to complete the NER task, just like the Conditional Random Fields(CRFs)-based NER System proposed by (Han et al., 2013). Recently, neural network-based (NN-based) models have shown great future prospects in improving the performance of NER systems, including bidirectional Long Short-Term Memory (LSTM) model (He et al., 2019), lattice-structured LSTM model (Zhang and Yang, 2018), convolution neural network (CNN)-based model (Gui et al., 2019) and so on. In our work, we adopt the most mainstream NN-based NER algorithm at present, which combines BiLSTM and CRF.

Relation extraction (RE) is also one of the most important tasks in NLP. On the premise of pre-defined relation categories, RE is often transformed into a relation classification task. Similar to entity extraction, the mainstream algorithms for RE in recent years have also focused on NN-based ones. Zeng et al. (2014) utilized CNNs to classify relations and made representative progress. However, because CNN can not extract contextual semantic information well, recurrent neural network (RNN) (Zhang and Wang, 2015), which is often used to process texts, is proposed for relation extraction. Since RNN is difficult to learn long-term dependencies, LSTM (Zhang et al., 2015) was introduced into the RE task. To capture the most important information in a sentence, Attention-Based Bidirectional Long Short-Term Memory Networks(Att-BLSTM) (Zhou et al., 2016) was come up and become a popular RE algorithm. The above supervise learning algorithms are time-consuming and costly to label data. In order to solve these problems, some distant supervision algorithms have also been developed (Zeng et al., 2015; Han and Sun, 2016; Ji et al., 2017). Because the TKG only contains knowledge in the field of tourism, the corpus for training is not large, so we do not consider using distant supervision algorithms.

3 Implementation

In this paper, we crawl semi-structured and unstructured data related to Hainan Tourism from popular travel websites, and extract the structured knowledge from these two types of data in two pipelines. Figure 2 shows the overview of our method.

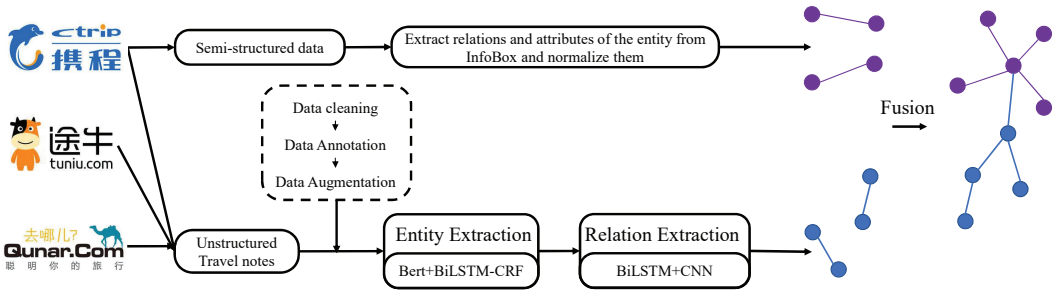


Figure 2: The overview of our method.

3.1 Data Preparation

Tourism is an intelligent application market with great potential. Tourism data on the Internet has a large quantity but not effectively used, and standardized tourism datasets are not yet available. In this section, we will describe our data preparation process in detail, which is mainly divided into four steps including data acquisition, data cleaning, data annotation and data augmentation, and the last three steps are mainly aimed at unstructured data that is noisy and irregular.

Data acquisition: This step aims to collect raw data in the field of tourism, which will be processed later to be used as input to the information extraction models. There are many popular Chinese tourism websites that cover numerous tourism-related knowledge on the Internet. We crawled semi-structured data on the Ctrip[0], where tourism-related entities (scenic areas, hotels, cities, etc.) have their corresponding descriptive pages. The Information Boxes (InfoBox) in these pages with clear structure contain a great number of named entities, relations and attributes, which can be used to fill the TKG. For example, the InfoBox of "Haikou Ublaya Inn" is shown in the Figure 3(a). Meanwhile, we crawled tourists' travel notes related to Hainan on the three major Chinese travel websites, Ctrip[1], Tuniu[2] and Qunar[3]. Travel notes are rich in content and easy to obtain, which may supplement the information not contained in semi-structured data, and Figure 3(b) shows an example of travel notes on the Tuniu.

(a) InfoBox Example. (b) Travel notes Example.

Figure 3: An example of (a) an InfoBox of "Haikou Ublaya Inn" and (b) travel notes related to Hainan on the Tuniu, which respectively correspond to the semi-structured data and unstructured data that we want to crawl on the travel websites.

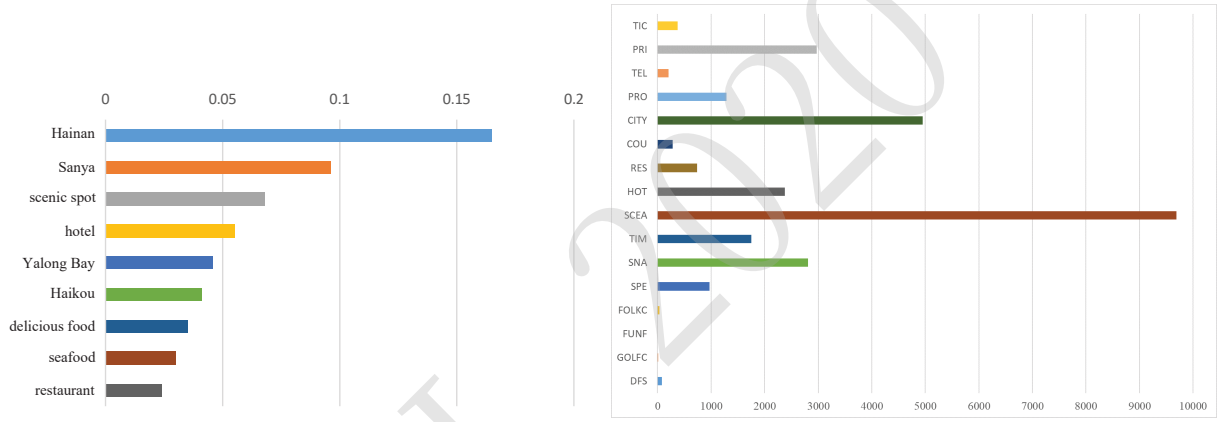
We have crawled 33177 pages corresponding to Hainan-related entities from the Ctrip. In addition, a total of 19,023 travel notes are obtained after crawling the above three popular websites. The combination of semi-structured data and unstructured data helps to provide a more complete source of information in the construction of TKG.

Data cleaning: For unstructured data, due to the colloquial and casual nature, the travel notes crawled from the travel websites usually contain some noise that should be cleaned up, including some inconsistent Traditional Chinese characters, emoticons, Uniform Resource Locator (URL) links and some special characters like #, &, $, {, }, etc. We mainly delete these redundant contents through regular expressions. In view of the fact that some paragraphs in travel notes are relatively longer than the ideal length required by the models for entity extraction and relation extraction, we further perform paragraph segmentation to reduce the pressure of model training.

Data Annotation: For unstructured text, we should label it to build datasets that meet the training requirements for subsequent entity recognition and relation recognition algorithms. Before annotating data, we must first define the types of entities and relations that need to be extracted in the field of tourism. In order to truly understand the issues that users are concerned about, we crawl the text about

[0]https://you.ctrip.com/place/100001.html
[1]https://you.ctrip.com/travels/
[2]https://trips.tuniu.com/
[3]https://travel.qunar.com/

the keyword ”Hainan” in the QA modules of Ctrip and Tuniu, mainly including some users' questions and the answers given by other users, and then the word frequency in the Q & A data is analyzed through TF-IDF (Term Frequency-Inverse Document Frequency) algorithm. The statistical results of word frequency in our work are shown in figure 4(a). The results show that high-frequency words are mainly concentrated on types such as hotel, scenic spot, city, food, restaurant, etc. Referring to the definition of entities and relations in CN-DBpedia (Xu et al., 2017), we define 16 entity types and 51 relation types that should be extracted from unstructured data based on the features of tourism-domain data. Entity types include DFS (Duty Free Shop), GOLFC (Golf Course), FUNF (Funfair), HOT (Hotel), FOLKC (Folk Custom), SPE (Specialty), SNA (Snacks), TIM (Time), TEL (Telephone), PRI (Price), TIC (Ticket), SCEA (Scenic Area), PRO (Province), CITY (City), COU (County) and RES (Restaurant). Because of the relatively large number of relation types, we give an example to illustrate the relation types. When choosing a restaurant, tourists need to figure out the location, price, business hours and phone number of the hotel, and the location must be specific. So we define seven relations for RES type, including res_locatedin_scea, res_locatedin_pro, res_locatedin_city, res_locatedin_cou, res_open_time, res_phonenumber, res_PRI, where res_locatedin_scea means that the restaurant is in a certain scenic area, and the explanation of the remaining relations is similar.



(a) Statistics of word frequency.

(b) Statistics of entities of 16 types.

Figure 4: (a) Word frequency statistics in the Q & A data, where high-frequency words need to be focused on; (b) Statistics of the number of 16 types of entities, it shows that the number of entities is unevenly distributed.

After defining the entity & relation types to be extracted, for a sentence in travel notes, we should first label entity mentions in it, and then label the relation between entity pairs according to semantics. We adopt BRAT (Stenetorp et al., 2012) as the main tool to label entities and relations in the text. There exist some problems when using BRAT to label entities and relations in the field of tourism. When labeling entities, 1) The travel notes are expressed by different people in a colloquial way, which makes it difficult to determine the boundary of the entities. We reasonably label the entity mentions with the boundary as large as possible, so as to make the entity mention more complete and specific; 2) In different contexts, entities with the same mention may belong to different types. So we label relations based on the semantics of the context. There are also some problems when labeling relations, 1) When multiple entities appear in a sentence, and there is more than one entity pair that has connections, we label as many entity-relation-entity combinations to obtain adequate relation annotated data; 2)A sentence may contain two entities, and there may be a connection between the two entities according to external knowledge, but the context of the sentence cannot reflect this connection. For this situation, we will not label the relation, so as not to have a negative impact on the subsequent training of the RE model.

After handling the above problems, 1902 travel notes are annotated. Because labeling relations needs to consider the context, which affects the speed of the labeling, we have not labeled all crawled travel notes, but only labeled the number enough to train the models. The details of the datasets will be shown

in Section 4.1.

Data Augmentation: The number of entities in travel notes is not evenly distributed in categories. We make statistics on the number of entities of each entity type contained in the annotated dataset, as shown in the figure 4(b). We can see that there are a large number of labeled entities in SCEA and CITY types, and the proportion of other types is relatively small. In order to reduce the training error brought by data imbalance, we use substitution method to expand the types with a small amount of data. We take the DFC entities with a small proportion as example. First, select some sentences containing the DFC entity from the dataset, and then replace the DFC entity mentions in each sentence with other different DFC mentions. Although such replacement destroys the authenticity of the original data, the training for models is appropriate. We use this technology to augment a total of more than 8,000 pieces of sentence.

3.2 Knowledge Extraction of Semi-structured Data

Since a page crawled on Ctrip tends to contain the description of the relevant attributes and relations of only one named entity, we extract not only entity mention but also the corresponding URL, and the URL can uniquely represent the entity. In this way, we successfully extract the uniquely identifiable entities from the semi-structured data, and there is no ambiguity between these entities. In addition, we extract attributes and relations of a entity mainly through the InfoBox. It is worth noting that there are many cases of inconsistent attributes and value conflicts in Semi-structured data. For example, attribute names can be inconsistent (telephone, contact number), and attribute values can be inconsistent(086-6888-8888 and 68888888), so for the extracted semi-structured knowledge, we further refer to CN-DBpedia (Xu et al., 2017) for attribute normalization and value normalization to further obtain well-organized knowledge, and then we finally obtain about 370,000 triples from semi-structured data.

3.3 Knowledge Extraction of Unstructured Data

In this section, we mainly utilize mainstream deep learning algorithms to extract entities and relations in unstructured text.

Entity extraction. For unstructured data from tourist travel notes, we take the method of Named Entity Recognition (NER) to extract entity mentions from the text. The main work of NER is sequence labeling, and Long Short-Term Memory (LSTM) networks have natural advantages in processing time series related tasks. The Conditional Random Field (CRF) model can effectively consider the mutual influence of output labels between characters. Therefore, the BiLSTM model and the CRF model are usually used together to become the mainstream model in the NER field.

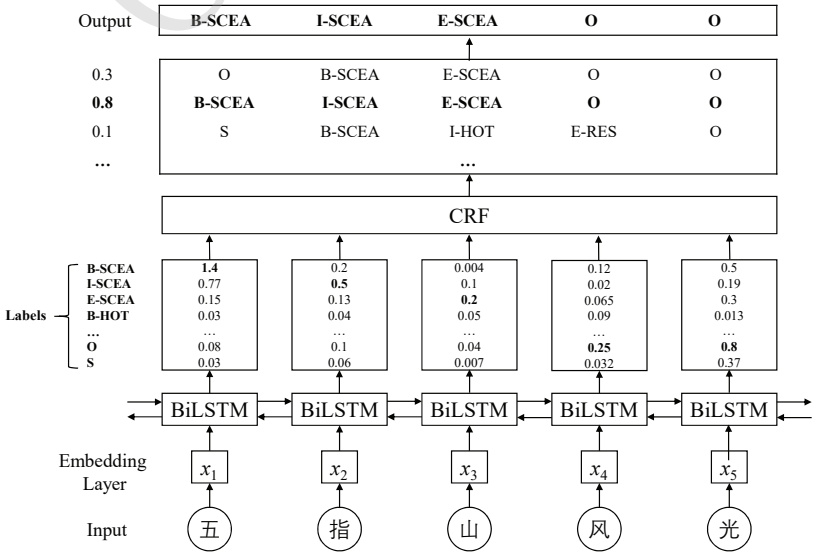


Figure 5: The baseline framework of entity extraction model based on BiLSTM-CRF.

The BiILSTM-CRF (Huang et al., 2015) model diagram is shown in Figure 5. The input in English is "Five-Finger Mountains' scenery", and the output means that Five-Finger Mountains belong to the entity type SCEA. Next, We make further improvements in the embedding layer. After the Google BERT (Devlin et al., 2018) model was proposed, the innovation of the pretrained language model has enabled many NLP tasks to achieve state-of-the-art performance, and large pretrained language models have become a hot tool. After BERT, other large pretrained language models like ALBERT (Lan et al., 2019) model have also been proposed. ALBERT is a simplified BERT version, and the number of parameters is much smaller than the traditional BERT architecture. In this paper, we utilize the pretrained BERT and ALBERT model to obtain the embedding matrix in embedding layer respectively, which is constant during the training process.

Relation extraction. Relation Extraction (RE) is an important task of natural language processing (NLP) and also a key link in knowledge graph construction. After RE, a triple (s, r, o) is usually obtained, where s represents the head entity, o represents the tail entity, and r represents the relation between them. In our travel data, the number of relations is limited, so we can choose to transform the RE into a relation classification task, and we treat each relation type as a class. Comprehensively considering the advantages and disadvantages of the mainstream relationship classification model and characteristics of tourism data, we choose to adopt supervised algorithm, BiLSTM+CNN (Zhang and Xiang, 2018), for RE task in our work, whose framework can be shown in Figure 6. CNN can extract local features of sentences, but it is not good at handling long dependencies among words, which can be made up by BiLSTM.

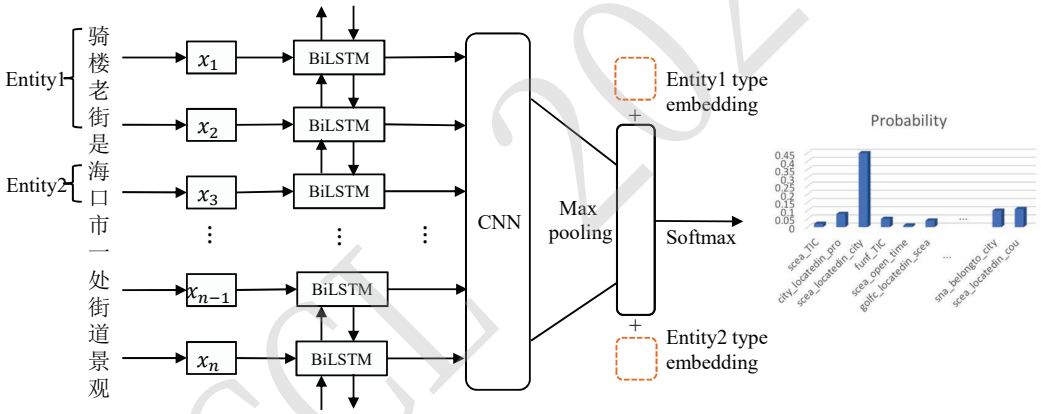


Figure 6: The framework of the relation extraction model based on BiLSTM+CNN. The input in English is "Qilou Old Street is a street view of Haikou City.", and from the bar graph we know that the output relation is scea_locatedin_city, then we can get the triple (Qilou Old Street, scea_locatedin_city, Haikou).

At the same time, considering that the entity category information may have an impact on the relation classification, the entity type information is introduced into the model (Lee et al., 2019). Specifically, each entity type is represented as distributed embedding. As is shown in Figure 6, after the CNN layer, we concatenate the entity type embedding of entity1 and entity2 with the output vector of Max pooling layer, and then feed it to the fully connected layer for subsequent label prediction.

Entity Alignment. There is often a situation where multiple mentions refer to the same entity. Entity alignment is to determine whether two entities with different mentions are the same entity by calculating and comparing their similarity. We observe the names of the entities that need to be aligned and find that the names of the two entities to be aligned are similar in most cases, just like "Nantian Ecological Grand View Garden" and "Nantian Grand View Garden". Therefore, basic distance measurement-based models are suitable enough for our entity alignment task, which is to calculate the distance between the names of the two entities. Common distance measurement algorithms include Jaccard coefficient, Euclidean distance, and editing distance. We weight and sum the distances measured under these three distance metrics, so as to discriminate whether entities with different names belong to the same entity. Although this method is simple, but it can solve most of the problems we encounter. Finally, we obtain about

220,000 triples from unstructured data.

In summary, we first construct independent knowledge graphs from two heterogeneous data sources respectively, and then we fuse the two sub-knowledge graphs to obtain a more complete knowledge graph, which is the Tourism-domain Knowledge Graph finally constructed.

4 Experiments

4.1 Datasets

In Section 3.1, we acquire, clean, annotate and augment the unstructured text crawled from popular travel websites, and obtain two labeled datasets suitable for Named Entity Recognition (NER) and Relation Extraction (RE) tasks. For labeled datasets, post-processing operations are needed to eliminate data that is meaningless for model training. Specifically, if there is no entity in a sentence, delete it directly. If the sentence contains only one entity, it will be cut to the proper length and only be used for NER training. Our datasets are both based on sentences, and a sentence is a piece of data. For NER dataset, we use train, valid, and test splits of 5490, 1178, and 591 sequence labeled sentences respectively. And train, valid, and test sets for RE task contain 6225, 1000 and 400 sentences respectively. Using the datasets we construct and divide, we next conduct comparative experiments to measure the model performance.

4.2 Model Training and Results

In order to obtain a named entity recognition model suitable for tourism-domain data, we compare several mainstream NER models including BERT(Cai, 2019), ALBERT(Lan et al., 2019), BiLSTM-CRF(Huang et al., 2015), BERT+BiLSTM-CRF(Dai et al., 2019), and BERT-CRF(Souza et al., 2019) on our NER dataset. For this task, we use Precision (P), Recall (R) and F1 score (F1) to evaluate the effect of NER model, which are standard information extraction metrics. The experimental results in Table 1 show that the BiLSTM-CRF algorithm based on the pretrained language model BERT has the best performance with F1-score 90.6%. BERT+BiLSTM-CRF practiced by Dai et al. (2019) is used to complete the task of Chinese electronic medical records named entity recognition, and BERT+BiLSTM-CRF achieves approximately 75% F1 score and performs better than other models like BiLSTM-CRF and BiGRU-CRF in their work, which is consistent with our results. Both in their and our practice, the effectiveness of combining pretrained models with mainstream models is reflected. Meanwhile, we can see that baselines other than BERT+BiLSTM-CRF that have good performance on the general standard datasets can also achieve comparative results in the application of actual projects.

The NER models share the same divided NER dataset and training environment, and all models are trained with 15 epochs.

	Model	P	R	F1
NER	BiLSTM-CRF(Huang et al., 2015)	**0.890**	0.876	0.883
	BERT-CRF(Souza et al., 2019)	0.862	0.904	0.882
	BERT(Cai, 2019)	0.822	0.867	0.839
	ALBERT(Lan et al., 2019)	0.837	0.829	0.828
	BERT+BiLSTM-CRF(Dai et al., 2019)	0.887	**0.926**	**0.906**
RE	BiLSTM+ATT(Zhou et al., 2016)	0.766	0.681	0.702
	CNN(Zeng et al., 2014)	0.803	0.651	0.701
	BiLSTM-CNN(Zhang and Xiang, 2018)	**0.941**	0.791	0.842
	BiLSTM-CNN(with types)	0.918	**0.914**	**0.909**

Table 1: Comparison of experimental results with NER baselines and RE baselines on our datasets.

Similar to entity extraction, we also compare three mainstream models in relation extraction task, including BiLSTM+ATT (Zhou et al., 2016), CNN (Zeng et al., 2014) and BiLSTM-CNN (Zhang and Xiang, 2018). The evaluation metrics applied in RE models are also P, R and F1. Among these models, as is shown in Table 1, BiLSTM-CNN shows the relatively better performance than BiLSTM+ATT and CNN on our RE dataset.

In order to further verify the validity of adding entity type embedding in RE, comparative experiments are carried out on the model BiLSTM-CNN. Table 1 shows that by introducing entity type information, the F1 score of BiLSTM-CNN is improved by 6.7%, which is the highest among our experimental models. The main reason may be that by introducing entity type information into the model, the scope of classification is narrowed, that is to say, entity type information restricts the classification to a certain extent, so as to significantly improve the effect of relation classification. The above RE models share the same divided RE dataset and training environment, and all models are trained with 64 epochs.

To sum up, based on the above analysis of the experimental results of each model, BERT+BiLSTM-CRF is selected as NER model and BiLSTM + CNN model with entity type information introduced is selected as the RE model in our work.

4.3 Knowledge Construction

We fuse the two sub-knowledge graphs obtained from semi-structured data and unstructured data to get the complete TKG. The final TKG with a total of 441,371 triples contains 13 entity types and 46 relation types. In Figure 7, a knowledge graph composed of partial triples is depicted. The central node is Sansha that belongs to CITY type, and we show a part of the nodes around it and the adjacent relations and attributes.

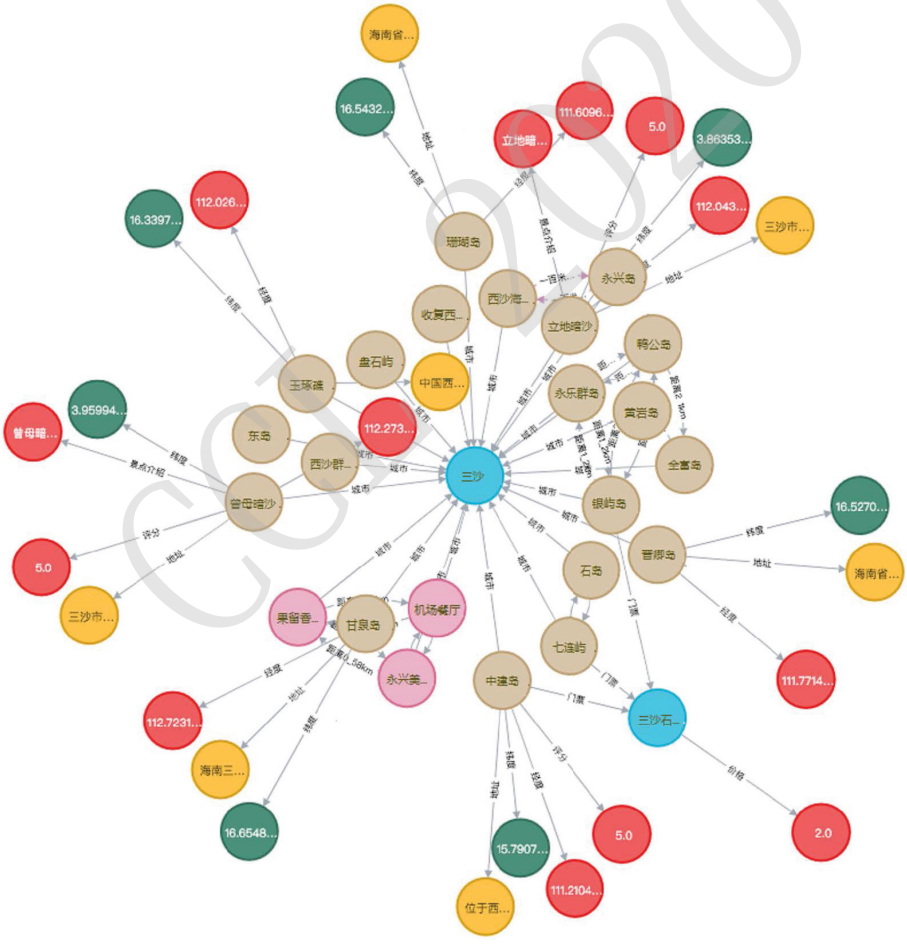


Figure 7: Partial triples in tourism knowledge graph, which shows the part of the tourism-domain knowledge graph with CITY Sansha as the central node.

5 CONCLUSIONS

With the development of tourism, information management and utilization in the field of tourism is a very important task. We proposed a systematic approach to construct the Chinese tourism knowledge graph,

using the information on the tourism websites. We leveraged semi-structured data and unstructured data to extract entities and relations synchronously, and they can be combined to obtain more complete sets of entities and relations than only one of them. Due to the lack of standardized datasets in the field of tourism, we first proposed a strategy for constructing datasets to facilitate the extraction of entities and relations from the complex network text data. In addition, we used several algorithms to complete the named entity recognition (NER) task and relation extraction (RE) task on the datasets we created, and compare the results. We found that BERT+BILSTM-CRF has the best performance for NER task and BiLSTM+CNN with entity type information introduced performs best on RE task.

We have implemented a relatively complete information extraction system on the tourism knowledge graph. In the future work, we want to solve the problem of how to update the knowledge in real time, because the knowledge on the tourism websites is always increasing and changing. In addition, we intend to explore some domain-adaptive techniques to make our model can be used widely.

Acknowledgements

This work is supported by the Science and Technology Department of Hainan Province, "Intelligent analysis platform of Hainan tourist's behavior and accurate service mining prediction" project(ZDKJ201808).

References

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer.

Qing Cai. 2019. Research on chinese naming recognition model based on bert embedding. In *2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS)*, pages 1–4. IEEE.

Pablo Calleja, Freddy Priyatna, Nandana Mihindukulasooriya, and Mariano Rico. 2018. Dbtravel: A tourism-oriented semantic graph. In *International Conference on Web Engineering*, pages 206–212. Springer.

Zhenjin Dai, Xutao Wang, Pin Ni, Yuming Li, Gangmin Li, and Xuming Bai. 2019. Named entity recognition using bert bilstm crf for chinese electronic health records. In *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–5. IEEE.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Tao Gui, Ruotian Ma, Qi Zhang, Lujun Zhao, Yu-Gang Jiang, and Xuanjing Huang. 2019. Cnn-based chinese ner with lexicon rethinking. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 4982–4988. AAAI Press.

Xianpei Han and Le Sun. 2016. Global distant supervision for relation extraction. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Aaron L-F Han, Derek F Wong, and Lidia S Chao. 2013. Chinese named entity recognition with conditional random fields in the light of chinese characteristics. In *Intelligent Information Systems Symposium*, pages 57–68. Springer.

Zhonghe He, Zhongcheng Zhou, Liang Gan, Jiuming Huang, and Yan Zeng. 2019. Chinese entity attributes extraction based on bidirectional lstm networks. *International Journal of Computational Science and Engineering*, 18(1):65–71.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Guoliang Ji, Kang Liu, Shizhu He, and Jun Zhao. 2017. Distant supervision for relation extraction with sentence-level attention and entity descriptions. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Joohong Lee, Sangwoo Seo, and Yong Suk Choi. 2019. Semantic relation classification via bidirectional lstm networks with entity-aware attention using latent entity typing. *Symmetry*, 11(6):785.

Zeyu Meng, Dong Yu, and Endong Xun. 2014. Chinese microblog entity linking system combining wikipedia and search engine retrieval results. In *Natural Language Processing and Chinese Computing*, pages 449–456. Springer.

Qingliang Miao, Yao Meng, and Bo Zhang. 2015. Chinese enterprise knowledge graph construction based on linked data. In *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, pages 153–154. IEEE.

Xing Niu, Xinruo Sun, Haofen Wang, Shu Rong, Guilin Qi, and Yong Yu. 2011. Zhishi. me-weaving chinese linking open data. In *International Semantic Web Conference*, pages 205–220. Springer.

Maya Rotmensch, Yoni Halpern, Abdulhakim Tlimat, Steven Horng, and David Sontag. 2017. Learning a health knowledge graph from electronic medical records. *Scientific reports*, 7(1):1–11.

Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. 2019. Portuguese named entity recognition using bert-crf. *arXiv preprint arXiv:1909.10649*.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. Brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107. Association for Computational Linguistics.

Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706.

Bo Xu, Yong Xu, Jiaqing Liang, Chenhao Xie, Bin Liang, Wanyun Cui, and Yanghua Xiao. 2017. Cn-dbpedia: A never-ending chinese knowledge extraction system. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 428–438. Springer.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, Jun Zhao, et al. 2014. Relation classification via convolutional deep neural network.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1753–1762.

Dongxu Zhang and Dong Wang. 2015. Relation classification via recurrent neural network. *arXiv preprint arXiv:1508.01006*.

Lei Zhang and Fusheng Xiang. 2018. Relation classification via bilstm-cnn. In *International Conference on Data Mining and Big Data*, pages 373–382. Springer.

Yue Zhang and Jie Yang. 2018. Chinese ner using lattice lstm. *arXiv preprint arXiv:1805.02023*.

Shu Zhang, Dequan Zheng, Xinchen Hu, and Ming Yang. 2015. Bidirectional long short-term memory networks for relation classification. In *Proceedings of the 29th Pacific Asia conference on language, information and computation*, pages 73–78.

Weizhen Zhang, Han Cao, Fei Hao, Lu Yang, Muhib Ahmad, and Yifei Li. 2019. The chinese knowledge graph on domain-tourism. In *Advanced Multimedia and Ubiquitous Engineering*, pages 20–27. Springer.

Zhanfang Zhao, Sung-Kook Han, and In-Mi So. 2018. Architecture of knowledge graph construction techniques. *International Journal of Pure and Applied Mathematics*, 118(19):1869–1883.

Mingxiong Zhao, Han Wang, Jin Guo, Di Liu, Cheng Xie, Qing Liu, and Zhibo Cheng. 2019. Construction of an industrial knowledge graph for unstructured chinese text learning. *Applied Sciences*, 9(13):2720.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212.

A Novel Joint Framework for Multiple Chinese Events Extraction

Nuo Xu [1,2], Haihua Xie [2], Dongyan Zhao [1]

[1] Wangxuan Institute of Computer Technology, Peking University, Beijing 100080, China

[2] State Key Laboratory of Digital Publishing Technology,
Peking University Founder Group Co., Ltd., Beijing 100871, China

xunuo2019@pku.edu.cn, xiehh@founder.com, zhaodongyan@pku.edu.cn

Abstract

Event extraction is an essential yet challenging task in information extraction. Previous approaches have paid little attention to the problem of roles overlap which is a common phenomenon in practice. To solve this problem, this paper defines event relation triple to explicitly represent relations among triggers, arguments and roles which are incorporated into the model to learn their inter-dependencies. A novel joint framework for multiple Chinese events extraction is proposed which jointly performs predictions for event triggers and arguments based on shared feature representations from pre-trained language model. Experimental comparison with state-of-the-art baselines on ACE 2005 dataset shows the superiority of the proposed method in both trigger classification and argument classification.

1 Introduction

Event extraction (EE) is of utility and challenge task in natural language processing (NLP). It aims to identify event triggers of specified types and their arguments in text. As defined in Automatic Content Extraction (ACE) program, the event extraction task is divided into two subtasks, i.e., trigger extraction (identifying and classifying event triggers) and argument extraction (identifying arguments and labeling their roles).

Chinese event extraction is a more difficult task because of language specific issue in Chinese (Chen and Ji, 2009). Since Chinese does not have delimiters between words, segmentation is usually a necessary step for further processing, leading to word-trigger mismatch problem (Lin et al., 2018). The approaches based on word-wise classification paradigm commonly suffer from this. For instance, two characters in one word "打死" (hit and die) trigger two different events: an "Attack" event triggered by "打" (hit) and a "Die" event triggered by "死" (die). It is hard to extract accurately when a trigger is part of a word or cross multiple words. To avoid this issue, we formulate Chinese event extraction as a character-based classification task. In addition, another interesting issue in event extraction which is rarely followed requires more efforts. It is the roles overlap problem that we concern in this paper, including the problems of either roles sharing the same argument or arguments overlapping on some words. There are

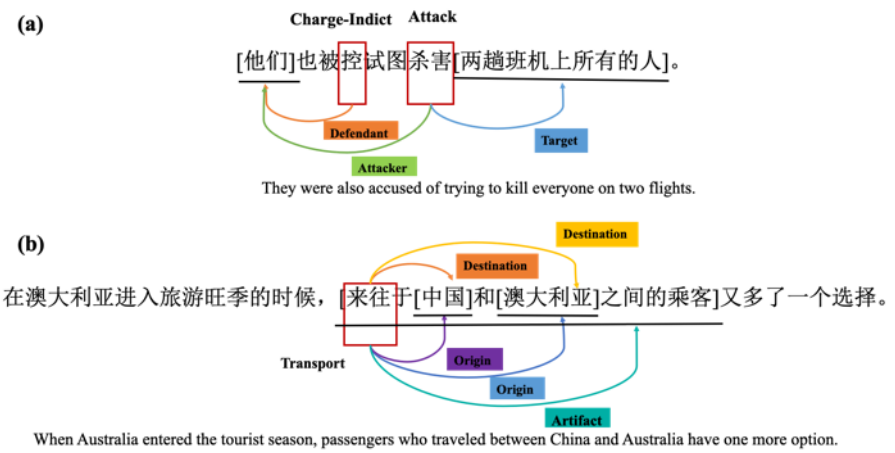


Figure 1: Examples of roles overlap problem

multiple events existing in the one sentence, which commonly causes the roles overlap problem and is easy to overlook (Yang et al., 2019). Fig. 1(a) shows example of roles sharing the same argument in ACE 2005 dataset. “控” (accuse) triggers a Charge-Indict event and “杀害” (kill) triggers an Attack event, while argument “他们” (them) plays the role “Defendant” as well as the role “Attacker” at the same time. Fig. 1(b) shows example of arguments overlapping on some words in ACE 2005 dataset. “来往” (traveled between) triggers a Transport event, while argument “中国” (China) plays not only the role “Origin” but “Destination” and argument “来往于中国和澳大利亚之间的乘客” (passengers who traveled between China and Australia) plays the role “Artifact”. We observe that the above two arguments overlap on word “中国” (China), which is more challenging for traditional methods to simultaneously identify these two arguments, especially for those being long noun phrases. Research shows that there exist about 10% events in ACE 2005 dataset (Doddington et al., 2004) having the roles overlap problem (Yang et al., 2019). Moreover, the results of event extraction could affect the effectiveness of many other NLP tasks, such as the construction of knowledge graph. If there exist roles overlap problems in events, the model identities accurately when it predicts any one argument or role, which leads to omission and incompleteness of information for knowledge graph construction and is obviously far from real-world applications. Therefore, the roles overlap problem is of great importance and needs to be seriously addressed.

It is thus appealing to design a single architecture to solve the problem. Although there exist prior studies that mention the roles overlap problem on ACE 2005 dataset, they share the limitations that include either depending on elaborate engineering features (i.e, hand-crafted features (He and Duan, 2019), dependency paths (Liu et al., 2018), etc.) or following the pipelined approach (Yang et al., 2019).

To overcome the issues of such prior works, in this paper, we propose a single framework to jointly extract triggers and arguments. Inspired by the effectiveness of pre-trained language models, we adopt bidirectional encoder representation from transformer (BERT) as the encoder to obtain the shared feature representations. Specifically, the relations among triggers (t),

arguments (a) and roles (r) are defined as event relation triples $< t, r, a >$ where r represents the dependencies of a on t in the event triggered by t. The event sentence of Fig. 1(b) could be represented by event relation triples as < 来往, Origin, 中国 >, < 来往, Destination, 中国 >, < 来往, Origin, 澳大利亚 >, < 来往, Destination, 澳大利亚 >, < 来往, Artifact, 来往于中国和澳大利亚之间的乘客 >. As is seen, event relation triples could explicitly describe relations among the three items. The key contribution of this paper is to design a novel joint extraction framework which jointly conducts trigger and argument extraction with incorporating the event relations defined. The task of argument classification is converted to relation extraction. Specially, to extract multiple events and relation triples, we utilize multiple sets of binary classifiers to determine the spans (each span includes a start and an end). By this approach, not only roles overlap problem but also word-trigger mismatch and word boundary problems in Chinese language are solved. Our framework avoids human involvements and elaborate engineering features in event extraction, but yields better performance over prior works.

This paper is organized as follows: Section 2 presents the related work for EE. Section 3 introduces our approach to tackle problems of roles overlap. Extensive experiments are conducted to evaluate the effectiveness of the proposed model on widely-used dataset ACE 2005 in Section 4. Besides, more rigorous evaluation criteria are adopted in experiments. Conclusions and future work are given in Section 5.

2 Related Work

EE is an important task which has attracted many attentions. There are two main paradigms for EE: a) the joint approach that predicts event triggers and arguments jointly, and b) the pipelined approach that first identifies trigger and then identifies arguments in separate stages (Nguyen et al., 2016). The advantages of such a joint system are twofold: (1) mitigating the error propagation from the upstream component (trigger extraction) to the downstream classifier (argument extraction), and (2) benefiting from the inter-dependencies among event triggers and argument roles (Nguyen and Nguyen, 2019). Traditional methods that rely heavily on hand-craft features are hard to transfer among languages and annotation standards (Chen and Ng, 2012; Liao and Grishman , 2010; Li et al., 2013). The neural network based methods that are able to learn features automatically (Chen et al., 2015; Feng et al., 2016; Nguyen et al., 2016; Nguyen and Grishman, 2016; Zeng et al., 2016) have achieved significant progress. Most of them have followed the pipelined approach. Some improvements have been made by jointly predicting triggers and arguments (Liu et al., 2018; Nguyen et al., 2016; Nguyen and Nguyen, 2019) and introducing more complicated architectures to capture larger scale of contexts. These methods have achieved promising results in EE.

Unfortunately, roles overlap problem has been put forward (He and Duan, 2019; Yang et al., 2019), but there are only few works in the literature to study this. He and Duan (2019) construct a multi-task learning with CRF enhanced model to jointly learn sub-events. However, their method relies on hand-crafted features and patterns, which makes them difficult to be integrated into recent neural models. The similar work to ours is Yang et al.(2019) that adopts

a two-stage event extraction by adding multiple sets of binary classifiers to solve roles overlap problem. But this work needs to detect triggers and arguments separately which suffers from error propagation. It does not employ shared feature representations as we do in this work.

In recent years, pre-trained language models are successful in capturing words semantic information dynamically by considering their context. McCann et al.(2017) pre-train a deep LSTM encoder from an attentional sequence-to-sequence model for machine translation (MT) to contextualize word vectors. ELMo (Embeddings from Language Models) improve 6 challenging NLP problems by learning the internal states of the stacked bidirectional LSTM (Long Short-Term Memory) (Peters et al., 2018). Open AI GPT (Generative Pre-Training) improves the state-of-the-art in 9 of 12 tasks (Radford et al., 2018). BERT obtains new state-of-the-art results on 11 NLP tasks (Devlin et al., 2018).

3 Extraction Model

This section describes our approach that is designed to extract events occurring in plain text. We now define the scope of our work. The task of argument extraction is defined as automatically extracting event relation triples defined. In our model, instead of treating entity mentions as being provided by human annotators, only event label types and argument role types are utilized as training data for both trigger and argument extraction.

We propose a pre-trained language model based joint multiple Chinese event extractor (JMCEE). Let $s = \{c_1, c_2, ..., c_n\}$ be annotated sentence s with n as the number of characters and c_i as the ith character. Given the set of event relation triples $E = \{< t, r, a >\}$ in s, the goal of our framework is to perform the task of trigger extraction T and argument extraction A jointly:

$$P(A, T|s) = P(A|T, s) \times P(T|s) = \prod_{(r,a)\in E|t} p((r, a)|t, s) \prod_{t\in E} p(l, t|s) \quad (1)$$

Here $(r, a) \in E|t$ denotes an argument and role pair (r, a) in the event triples E triggered by t and l denotes the event label type. Based on Eq. (1), we first predict all possible triggers and their label types in a sentence; then for each trigger, we integrate information of predicted trigger word to extract event relation triple $< t, r, a >$ by simultaneously predicting all possible roles and arguments, as illustrated in Fig. 2. We employ a pre-trained BERT encoder to learn the representation for each character in one sentence, then feed it into downstream modules. The input of our joint extractor follows the BERT, i.e. the sum of three types of embeddings, including WordPiece embedding, Position embedding and Segment embedding. Token [CLS] and [SEP] are placed at the start and end of the sentence. Multiple sets of binary classifiers are added on the top of the BERT encoder to implement predictions for multiple events and relation triples. For trigger extraction, we need to predict the start and end of event type l for $c_i \in s$ (l could be "Other" type to indicate that there is no word triggering any event) with each set of binary classifiers severing for an event type to determine the starts and ends of all triggers. For argument extraction, we need to extract event relation triple $< t, r, a >$ by predicting the start and end of role type r for c_i in sentence s based on predicted triggers (r is set to "Other"

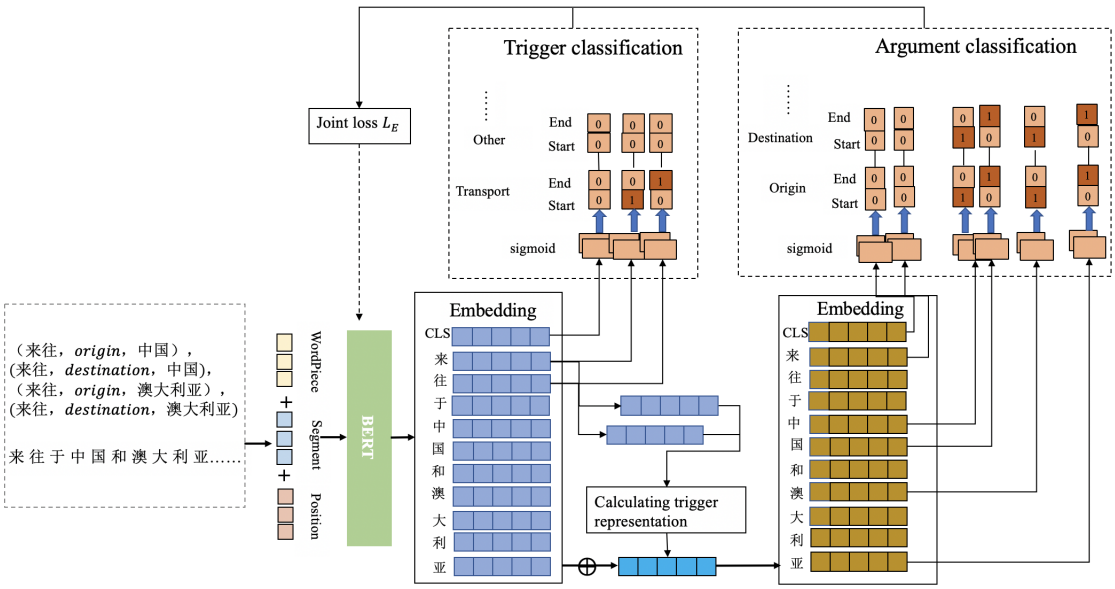


Figure 2: The framework of JMCEE, including the trigger extract component and the argument extract component. The extraction procedure of the event instance is shown.

if there is no word triggering any event as well) with each set of binary classifiers severing for a role to determine the starts and ends of all arguments that play it. The roles overlap problem could be solved since the prediction could belong to different arguments and roles. Besides, our JMCEE enables to identify those arguments being long noun phrases like "来往于中国和澳大利亚之间的乘客" (passengers who traveled between China and Australia), which tackles the word boundary problem often encountered in Chinese. Compared with sentence-level sequential modeling methods, our approach also avoids suffering low efficiency in capturing very long-range dependencies in previous works (Sha et al., 2018; Liu et al., 2018).

3.1 Trigger Extraction

Trigger extraction aims to predict whether a token is a start or an end of a trigger for type label l. A token c_i is predicted as the start of a trigger with probability for type label l through feeding it into a fully-connected layer with sigmoid activation function:

$$P^l_{Ts}(c_i) = \sigma(W^l_{Ts}\beta(c_i) + b^l_{Ts}) \tag{2}$$

while as the end with probability:

$$P^l_{Te}(c_i) = \sigma(W^l_{Te}\beta(c_i) + b^l_{Te}) \tag{3}$$

where we utilize subscript "s" to denote "start" and subscript "e" to denote "end". W_{Ts} and b_{Ts} are respectively the trainable weights and bias of binary classifier that targets to detect starts of triggers' labels, while W_{Te} and b_{Te} are respectively the trainable weights and bias of another binary classifier that targets to detect ends of triggers' labels. β is the BERT embedding. Set thresholds of detecting starts and ends as $\delta^l = \{\delta^l_s, \delta^l_e\}$, δ^l_s and δ^l_e are respectively the thresholds

of binary classifiers that targets to detect starts and ends of triggers' labels. If $P^l_{Ts}(c_i) > \delta^l_s$, token c_i is identified as the start of type label l. if $P^l_{Te}(c_i) > \delta^l_e$, token c_i is identified as the end of type label l.

3.2 Argument Extraction

Once the triggers and their type labels have been identified, we come to the argument extraction component. Argument classification is converted to event relation extraction for triple $< t, r, a >$. Note that when the sentence is identified as "Other" type, we simply skip the following operation for argument role extraction. To better learn the inter-dependencies among the multiple events appearing in one sentence, we randomly pick one of predicted triggers in a sentence during the training phase, while in the evaluation phase, all the predicted triggers are picked in turn to predict corresponding arguments and roles played in the triggering events. We integrate information of predicted trigger word to argument extraction component. In ACE corpus, more than 98.5% triggers contain no more than 3 characters, so we simply pick the embedding vectors of start $\beta_s(c_i)$ and end $\beta_e(c_j)$ of one predicted trigger word t, and then generate representation of trigger word $\beta(t)$ by averaging these two vectors.

$$\beta(t) = \frac{(\beta_s(c_i) + \beta_e(c_j))}{2} \tag{4}$$

When obtain representations of trigger words $\beta(t)$, we add original embedding generated by BERT and $\beta(t)$ together:

$$\beta^{'}(s) = \beta(s) + \beta(t) \tag{5}$$

After integrate information of predicted trigger word to BERT sentence encoding, feed $\beta^{'}(s)$ into a full-connected layer with sigmoid activation function. A token c_k is predicted as the start of an argument triggered by word t which plays role r with probability:

$$P_{As}(c_k, r|t) = \sigma(W^r_{As}\beta^{'}(c_k) + b^r_{As}) \tag{6}$$

while as the end triggered by word t with probability:

$$P_{Ae}(c_k, r|t) = \sigma(W^r_{Ae}\beta^{'}(c_k) + b^r_{Ae}) \tag{7}$$

where W_{Ae} and b_{As} are respectively the trainable weights and bias of binary classifier that targets to detect starts of arguments' roles, while W_{Ae} and b_{Ae} are respectively the trainable weights of the other binary classifier that detects ends of arguments' roles. Set thresholds of detecting starts and ends as $\varepsilon^r = \{\varepsilon^r_s, \varepsilon^r_e\}$, ε^r_s and ε^r_e are respectively the thresholds of binary classifiers that target to detect starts and ends of triggers' labels. If $P_{As}(c_k, r|t) > \varepsilon^r_s$,token c_k is identified as the start of argument role r. if $P_{Ae}(c_k, r|t) > \varepsilon^r_e$, token c_k is identified as the end of argument role r. Algorithm 1 is utilized to detect each token to determine triggers, types, arguments and roles.

3.3 Model Training

We train the joint model and define L_T as the loss function of all binary classifiers that are responsible for detecting triggers, shown as follows:

$$L_T = \frac{1}{m \times n}(\sum_{l=0}^{m}\sum_{i=0}^{n} -logP_{Ts}^{l}(c_i) + \sum_{l=0}^{m}\sum_{i=0}^{n} -logP_{Te}^{l}(c_i)) \tag{8}$$

L_T denotes the average of cross entropy of output probabilities of all binary classifiers which detect starts and ends of triggers on each type label. In the same way, we define L_A as the loss function of all binary classifiers that are responsible for detecting event relation triples:

$$L_A = \frac{1}{m \times n}(\sum_{r=0}^{m}\sum_{i=0}^{n} -logP_{As}(c_k, r|t) + \sum_{r=0}^{m}\sum_{i=0}^{n} -logP_{Ae}(c_k, r|t)) \tag{9}$$

Where m denotes the sum of event label types and argument role types. L_A denotes the average of cross entropy of output probabilities of all binary classifiers which detect starts and ends of arguments on each role. The final loss function $L_E = L_T + L_A$. We minimize the final loss function to optimize the parameters of the model.

Algorithm 1 trigger and argument identification

Input: P_{Ts}^{l},P_{Te}^{l},P_{As}, P_{Ae},predicted trigger matrix TP,predicted argument matrix AP, sentence s, label list L
Output: predicted trigger list L_T, length of L_T l, predicted argument list L_A

Take out matrix S_t of ids and labels of starts that satisfy $P_{Ts}^{l} > \delta_s^l$ from TP and matrix E_t of ids and labels of ends that satisfy $P_{Te}^{l} > \delta_e^l$ from AP
for each (id_s,l_s) in S_t **do**
  for each (id_e,l_e) in E_t **do**
    if $id_s < id_e \& l_s == l_e$ **then**
      $trigger \leftarrow s[id_s - 1, id_e]$
      $label \leftarrow L[l_e]$
      $Append[trigger, label] to L_T$
      $break$
    end if
  end for
end for
return L_t
if L_T **then**
  for $i = 0 \rightarrow l$ **do**
    Take out matrix S_{ai} of ids and labels of starts that satisfy $P_{As} > \varepsilon_s^r$ from AP and matrix E_{ai} of ids and labels of ends that satisfy $P_{Ae} > \varepsilon_e^r$ for ith trigger from AP
    for each (id_{si},r_{si}) in S_{ai} **do**
      for each (id_{ei},r_{ei}) in E_{ai} **do**
        if $id_{si} < id_{ei} \& r_{si} == r_{ei}$ **then**
          $argument \leftarrow s[id_{si} - 1, id_{ei}]$
          $role \leftarrow L[r_{ei}]$
          $Append[argument, role] to L_A$
          $break$
        end if
      end for
    end for
  end for
end if
return L_A

4 Experiments

We evaluate JMCEE framework on the ACE 2005 dataset that contains 633 Chinese documents. We follow the same setup as (Chen and Ji, 2009; Lin et al., 2018; Zeng et al., 2016), in which 549/20/64 documents are used for training/development/test set. The proposed model is compared with the following state-of-the-art methods:

1) DMCNN (Chen et al., 2015) adopts dynamic multi-pooling CNN to extract sentence-level features automatically.

2) Rich-C (Chen and Ng, 2012) is a joint-learning, knowledge-rich approach including character-based features and discourse consistency features, which is the feature-based state-of-art system.

3) C-BiLSTM (Zeng et al., 2016) designs a convolutional Bi-LSTM model which conduct Chinese event extraction from perspective of a character-level sequential labeling paradigm.

4) NPNs (Lin et al., 2018) performs event extraction in a character-wise paradigm, where a hybrid representation for each character is learned to capture both structural and semantic information from both characters and words.

ACE 2005 dataset annotates 33 event subtypes and 35 role classes. The tasks of event trigger classification and argument classification in this paper are combined into a 70-category task along with "None" word and "Other" type. In order to evaluate the effectiveness of our proposed model, we evaluate models by micro-averaged Precision (P), Recall (R) and F1-score followed the computation measures of Chen and Ji (2009). The following criteria are utilized to evaluate the performance of predicted results:

1) A trigger prediction is correct only if its span and type match with the golden labels.

2) An argument prediction is correct only if its span, role, related trigger and trigger type match with the golden labels.

It is worth noting that all the predicted roles for an argument are required to match with the golden labels, instead of just one of them. We take a further step to see the impacts of pipelined model and joint model. The pipelined model called MCEE which identifies triggers and arguments in two separate stages based our classification algorithm. The highest F-score parameters on the development set are picked and listed in Table 1.

Hyper-parameter	Trigger classification	Argument classification
character embedding	768	768
maximum length	510	510
batch size	8	8
learning rate of Adam	0.0005	0.0005
classification thresholds	[0.5,0.5,0.5,0.5]	[0.5,0.4,0.5,0.4]

Table 1: Hyper-parameters for experiments.

4.1 Overall Results

Table 2 shows the results of trigger extraction on ACE 2005. As is seen, our JMCEE framework achieves the best F1 scores for trigger classification among all the compared methods.

Note that the results of Rich-C could obtain more accurate estimation of model performance since it performed 10-fold cross-validation experiments. However, our JMCEE gains at least 8% F1-score improvements on trigger classification task on ACE 2005, which steadily outperforms all baselines. The improvement on the trigger extraction is quite significant, with a sharp increase of near 10% on the F1 score compared with these conventional methods.

Model	Trigger identification			Trigger classification		
	P	R	F1	P	R	F1
DMCNN	66.6	63.6	65.1	61.6	58.8	60.2
Rich-C	62.2	71.9	66.7	58.9	68.1	63.2
C-BiLSTM	65.6	66.7	66.1	60.0	60.9	60.4
NPNs	75.9	61.2	67.8	73.8	59.6	65.9
MCEE(BERT-Pipeline)	82.5	78.0	80.2	72.6	68.2	70.3
JMCEE(BERT-Joint)	**84.3**	**80.4**	**82.3**	**76.4**	**71.7**	**74.0**

Table 2: Comparison of different methods on Chinese trigger extraction on ACE 2005 test set. Bold denotes the best result.

Table 3 shows results of argument extraction. Compared with these baselines, our JMCEE is at least 3% higher over other models on F1-score on argument classification task. While the improvement in argument extraction is not so obvious comparing to trigger extraction. This is probably due to the rigorous evaluation metric we have taken and the difficulty of argument extraction. Note that by our approach we identify 89% overlap roles in test set. Moreover, results show that our joint model substantially outperforms the pipelined model whether on trigger classification or argument classification. It is seen that joint model enables to capture the dependencies and interactions between the two subtasks and communicate deeper information between them, and thus improves the overall performance.

Model	Argument identification			Argument classification		
	P	R	F1	P	R	F1
Rich-C	43.6	**57.3**	49.5	39.2	**51.6**	44.6
C-BiLSTM	53.0	52.2	52.6	47.3	46.6	46.9
MCEE(BERT-Pipeline)	59.5	40.4	48.1	51.9	37.5	43.6
JMCEE(BERT-Joint)	**66.3**	45.2	**53.7**	**53.7**	46.7	**50.0**

Table 3: Comparison of different methods on Chinese argument extraction on ACE 2005 test set. Bold denotes the best result.

4.2 The Effect of Classification Thresholds

The effectiveness of thresholds settings for the trigger and argument classification is studies in this subsection. Table 4 lists the results of thresholds settings of the starts and ends of both two tasks. Specially, we tune two set of thresholds of starts and ends of trigger and arguments through setting δ^l to be 0.5, 0.5 and setting ε^r ranging from 0.5 to 0.4. Then, set δ^l to be 0.5, 0.4 and set ε^r ranging from 0.5 to 0.4. By analyzing the results, we find that the best performance of JMCEE on trigger extraction is achieved with parameters 0.5, 0.5, 0.5, 0.5, while the best performance of JMCEE on argument extraction is achieved with parameters 0.5, 0.4, 0.5, 0.4.

It suggests that when the ends of thresholds of both trigger and argument classification are set to be 0.4 could identify more candidate triggers and arguments. More candidate triggers could contribute to identifying arguments as we incorporate inter-dependencies between event triggers and argument roles in our joint extraction architecture, while the increased triggers could bring more noise to trigger classification with decreasing on the F1 score.

δ_l		ε_r		Trigger classification			Argument classification		
Start	End	Start	End	P	R	F1	P	R	F1
0.5	0.5	0.5	0.5	**76.4**	**71.7**	**74.0**	53.4	43.7	48.0
0.5	0.5	0.5	0.4	71.2	68.9	70.0	50.3	44.9	47.5
0.5	0.5	0.4	0.5	74.1	69.6	71.8	52.6	45.7	48.9
0.5	0.4	0.5	0.5	74.6	69.2	71.8	49.5	44.2	46.7
0.5	0.4	0.5	0.4	73.8	71.4	72.6	**53.7**	**46.7**	**50.0**
0.5	0.4	0.4	0.5	72.0	70.7	71.3	47.8	47.5	47.7

Table 4: Results of thresholds settings for the start and end of trigger and argument classification. Bold denotes the best result

Overall, the experimental results are remarkable facts given that our framework achieves better performance without any external and manually-generated features. We consider this as a strong promise toward our proposed joint framework which could be used as a good starting point.

5 Conclusions

In this paper, we propose a simple yet effective joint Chinese multiple events extraction framework which jointly extracts triggers and arguments by adopting a pre-trained BERT encoder without elaborate engineering features. Our contribution in this work is as follows:

1) Event relation triple is defined and incorporated into our framework to learn inter-dependencies among event triggers, arguments and arguments roles, which solves the roles overlap problem.

2) Our framework performs event extraction in a character-wise paradigm by utilizing multiple sets of binary classifiers to determine the spans, which allows to extract multiple events and relation triples and avoids Chinese language specific issues such as word-trigger mismatch and word boundary problem.

Experiments have shown that our method outperforms conventional methods. We believe our proposed framework could be applied to many other NLP tasks for exploiting inner composition structure during extraction, such as Entity Relation Extraction. Our future work will focus on data generation to enrich training data and try to extend our framework to the open domain.

Acknowledgements

This work has been supported by National Key Research and Development Program(No.2019YFB1406302), China Postdoctoral Science Foundation(NO.2020M670057) and Beijing Postdoctoral Research Foundation(No.ZZ2019-92).

References

Chen Chen and Vincent Ng. 2012. *Joint Modeling for Chinese Event Extraction with Rich Linguistic Features*. Proceedings of COLING 2012, 529–544.

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng and Jun Zhao. 2015. *Event Extraction via Dynamic Multi-pooling Convolutional Neural Networks* . Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing,vol.1,167–176.

Zheng Chen and Heng Ji. 2009. *Language Specific Issue and Feature Exploration in Chinese Event Extraction*. Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, 209–212.

Jacob Devlin, Ming-W. Chang, Kenton Lee and Kristina Toutanova. 1972. *Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv preprint arXiv:1810.04805

George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel and Ralph Weischedel. 2004. *The Automatic Content Extraction (ACE) Program-tasks, Data, and Evaluation*. LREC,vol.2

Xiaocheng Feng, Bing Qin and Ting Liu. 2016. *A Language-independent Neural Network for Event Detection.*. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, vol.2, 66–71.

Ruifang He and Shaoyang Duan. 2019. Joint Chinese Event Extraction based Multi-task Learning. *Journal of Software*, 30(4):1015–1030.

Qi Li, Heng Ji and Liang Huang. 2013. *Joint Event Extraction via Structured Prediction with Global Features*. Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, vol.1, 73–82.

Shasha Liao and Ralph Grishman. 2010. *Using Document Level Cross-event Inference to Improve Event Extraction*. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, 789–797.

Hongyu Lin, Yaojie Lu, Xianpei Han and Le Sun. 2018. *Joint Chinese Event Extraction based Multi-task Learning*. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics,1565–1574.

Jian Liu, Yubo Chen, Kang Liu and Jun Zhao. 2018. *Event Detection via Gated Multilingual Attention Mechanism*. Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 4865–4872.

Xiao Liu, Zhunchen Luo and Heyan Huang. 2018. *Jointly Multiple Events Extraction via Attention-based Graph Information Aggregation*. arXiv preprint arXiv:1809.09078.

Bryan McCann, James Bradbury, Caiming Xiong and Richard Socher. 2017. Learned in Translation: Contextualized Word Vectors. *In Advances in Neural Information Processing Systems*, 6294–6305.

Trung-M. Nguyen and Thien-H. Nguyen. 2019. *One for all: Neural Joint Modeling of Entities and Events*. Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 6851–6858.

Thien Huu Nguyen, Kyunghyun Cho and Ralph Grishman. 2016. *Joint Event Extraction via Recurrent Neural Networks*. Proceedings of NAACL-HLT 2016, 300–309.

Thien Huu Nguyen and Ralph Grishman. 2016. *Modeling Skip-grams for Event Detection with Convolutional Neural Networks*. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 886–891.

Matthew E. Peters, Mark Neumann, Mohit Iyyer and Matt Gardner. 2018. *Deep Contextualized Word Representations*. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics, 2227–2237 .

Alec Radford, Karthik Narasimhan, Tim Salimans and Ilya Sutskever. 2018. *Improving Language Understanding by Generative Pre-training*. https://www.cs.ubc.ca/ amuham01/LING530/papers/radford2018improving.pdf.

Lei Sha, Feng Qian, Baobao Chang and Zhifang Sui. 2018. *Jointly Extracting Event Triggers and Arguments by Dependency-bridge RNN and Tensor-based Argument Interaction*. Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 5916–5923.

Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan and Dongsheng Li. 2019. *Exploring Pre-trained Language Models for Event Extraction and Generation*. Proceedings of the 57th Annual Meeting of the Association for Computa-tional Linguistics, 5284–5294.

Ying Zeng, Honghui Yang, Yansong Feng and Dongyan Zhao. 2016. *A Convolution Bilstm Neural Network Model for Chinese Event Extraction*. In: Lin CY., Xue N., Zhao D., Huang X., Feng Y. (eds) Natural Language Understanding and Intelligent Applications. ICCPOL 2016, NLPCC 2016. LNCS, vol. 10102, 275–287.

Entity Relative Position Representation based Multi-head Selection for Joint Entity and Relation Extraction

Tianyang Zhao[1]* and **Zhao Yan**[2] and **Yunbo Cao**[2] and **Zhoujun Li**[1]
[1]State Key Lab of Software Development Environment, Beihang University, Beijing, China
[2]Tencent Cloud Xiaowei, Beijing, China
{tyzhao,lizj}@buaa.edu.cn, {zhaoyan,yunbocao}@tencent.com

Abstract

Joint entity and relation extraction has received increasing interests recently, due to the capability of utilizing the interactions between both steps. Among existing studies, the Multi-Head Selection (MHS) framework is efficient in extracting entities and relations simultaneously. However, the method is weak for its limited performance. In this paper, we propose several effective insights to address this problem. First, we propose an entity-specific Relative Position Representation (eRPR) to allow the model to fully leverage the distance information between entities and context tokens. Second, we introduce an auxiliary Global Relation Classification (GRC) to enhance the learning of local contextual features. Moreover, we improve the semantic representation by adopting a pre-trained language model BERT as the feature encoder. Finally, these new keypoints are closely integrated with the multi-head selection framework and optimized jointly. Extensive experiments on two benchmark datasets demonstrate that our approach overwhelmingly outperforms previous works in terms of all evaluation metrics, achieving significant improvements for relation F1 by $+2.40\%$ on CoNLL04 and $+1.90\%$ on ACE05, respectively.

1 Introduction

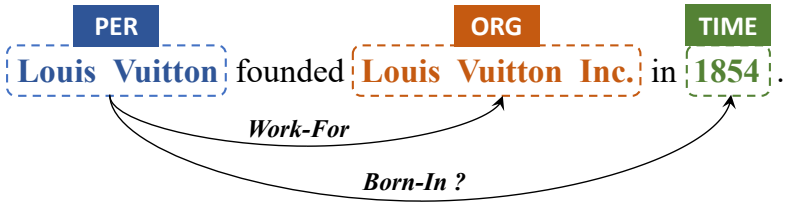
The entity-relation extraction task aims to recognize the entity spans from a sentence and detect the relations holds between two entities. Generally, it can be formed as extracting triplets (e_1, r, e_2), which denotes that the relation r holds between the head entity e_1 and the tail entity e_2, i.e., (*John Smith*, `Live-In`, *Atlanta*). It plays a vital role in the information extraction area and has attracted increasing attention in recent years.

Traditional pipelined methods divide the task into two phases, named entity recognition (NER) and relation extraction (RE) (Miwa et al., 2009; Chan and Roth, 2011; Lin et al., 2016). As such methods neglect the underlying correlations between the two phases and suffer from the error propagation issue, recent works propose to extract entities and relations jointly. These joint models fall into two paradigms. The first paradigm can be denoted as $(e_1, e_2) \rightarrow r$, which first recognizes all entities in the sentence, then classifies the relation depend on each extracted entity pairs. However, these methods require enumerating all possible entity pairs and the relation classification may be affected by the redundant ones. While another paradigm is referred as $e1 \rightarrow (r, e2)$, which detects head entities first and then predicts the corresponding relations and tail entities (Bekoulis et al., 2018; Li et al., 2019; Zhao et al., 2020). Comparing with the first paradigm, the second one can jointly identify entities and all the possible relations between them at once. A typical approach is the Multi-Head Selection (MHS) framework (Bekoulis et al., 2018). It first recognizes head entities using the BiLSTM-CRF structure and then performs tail entity extraction and relation extraction in one pass based on multiclass classification. The advantage of the MHS framework is obvious - it is efficient to work with the scenario, that one entity can involve several relational triplets, making this solution suitable for large scale practical applications. In this paper, we focus on the second paradigm of the joint models, especially on the MHS framework.

Despite the efficiency of the MHS framework, it is weak for the limited performance comparing with other complex models. Intuitively, the distance between entities and other context tokens provide impor-

* Work done during an internship at Tencent.

Proceedings of the 19th China National Conference on Computational Linguistics, pages 962-973,
Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China



Golden Relation:
(Louis Vuitton, ***Work-For***, Louis Vuitton Inc.)

Figure 1: An example to show the impact of entity-specific relative position.

tant evidence for entity and relation extraction. Meanwhile, the distance information of non-entity words is less important. As shown in the sentence of Fig. 1, the "Louis Vuitton" that is far from the word "Inc." is a person entity, while the one adjacent to "Inc." denotes an organization. Such entity-specific relative position can be a useful indicator to differentiate entity tokens and non-entity tokens and enhance interactions between entities. While the existing model pays equal attention to each context tokens and ignores the relative distance information of entities. As a result, the entity-specific features may become less obscure and mislead the relation selection. Second, the existing model predicts the relations and tail entities merely based on the local contextual features of the head entity, and the incomplete local information may confuse the predictor. While the semantic of the whole sentence always has a significant impact on relation prediction. For example, in Fig. 1, the relation between "Louis Vuitton" and "1854" may easily be mislabeled as "`Born-In`" without considering the meaning of the whole sentence. Therefore, the global semantics should also be taken into account.

To address the aforementioned limitations, we present several new key points to improve the existing multi-head selection framework. First, we propose an entity-specific Relative Position Representation (eRPR) to leverage the distance information between entities and their contextual tokens, which provides important positional information for each entity. Then, in order to better consider the sentence-level semantic during relation prediction, we add up an auxiliary Global Relational Classification (GRC) to guide the optimization of local context features. In addition, different from the original MHS structure, we adopt the pre-trained transformer-based encoder (BERT) to enhance the ability of semantic representations. Notably, the proposed method can address the entity and multiple-relation extraction simultaneously and without relying on any external parsing tools or hand-crafted features. We conduct extensive experiments on two widely-used datasets CoNLL04 and ACE05, and demonstrate the effectiveness of the proposed framework.

To summarize, the contributions of this paper are as follows:

- We propose an entity-specific relative position representation to allow the model aware of the distance information of entities, which provides the model with richer semantics and handles the issue of obscure entity features.
- We introduce a global relation classifier to integrate the essential sentence-level semantics with the token-level ones, which can remedy the problem caused by incompleted local information.
- Experiments on the CoNLL04 and ACE05 datasets demonstrate that the proposed framework significantly outperforms the previous work, achieving $+2.40\%$ and $+1.90\%$ improvements in F1-score on the two datasets.

2 Related Work

In this section, we introduce the related studies for this work, entity and relation extraction as well as the positional representation.

2.1 Entity and relation extracion

As a crucial content of information extraction, the entity-relation extraction task has always been widely concerned. Previous studies (Miwa et al., 2009; Chan and Roth, 2011; Lin et al., 2016) mainly focus

on pipelined structure, which divides the task into two independent phases, all entities are extracted first by an entity recognizer, and then relations between every entity pairs are predicted by a relation classifier. The pipelined methods suffer from error propagation issue and they ignore the interactions between the two phrases. To ease these problems, many joint models have been proposed to extract the relational triplets (e_1, r, e_2), simultaneously. According to different extraction order, the joint models can be categorized into two paradigms. The first paradigm first identifies all entities in the sentence, then traverses each pair of entities and determines their potential relation. Various models have achieved promising results by exploiting recurrent neural network (Miwa and Bansal, 2016; Luan et al., 2019), graph convolutional network (Sun et al., 2019; Fu et al., 2019) and transformer-based structure (Eberts and Ulges, 2019; Wang et al., 2019). Though effective, these models need to examine every possible entity pairs, which inevitably contains a lot of redundant pairs. In the second paradigm, the head entities are detected first and the corresponding relations and tail entities are extracted later. Bekoulis et al. (Bekoulis et al., 2018) present the multi-head selection framework to automatically extract multiple entities and relations at once. Huang et al. (Huang et al., 2019) improve the MHS framework by using NER pretraining and soft label embedding features. Recently, Li et al. (Li et al., 2019) cast the task as a question answering problem and identify entities based on a machine reading comprehension model. Different from the first one, the second paradigm is able to extract entities and all the relations between at once without enumerating every entity pair each time, which reduces redundant prediction and improves work efficiency.

Our work is inspired by the multi-head selection framework but enjoys new key points as follows. 1) We propose an entity-specific relative position representation to better encode the distance between entities and context tokens. 2) We incorporate the sentence-level information for relation classification to revise the learning of local features. 3) We enhance the original MHS framework with a pre-trained self-attentive encoder. Together these improvements contribute to the extraction performance remarkably.

2.2 Positional Representation

Generally, non-recurrent models do not contain the sequential order information of input tokens. Therefore, in order to fit for the sequential inputs, they need to design representations to encode positional information explicitly.

The approaches for positional representations can fall into three catagroies. The first one designs the position encodings as a deterministic function of position or learned parameters (Sukhbaatar et al., 2015; Gehring et al., 2017). These encodings are combined with input elements to expose position information to the model. For example, the convolutional neural networks inherently capture the relative positions within each convolutional kernels. The second catagroy is the absolute position representation. The Transformer structure (Vaswani et al., 2017) contains neither recurrence nor convolution, in order to inject the positional information to the model, it defines the sine and consine functions of different frequencies to encode absolute positions. However, such absolute positions cannot model the interaction information between any two input tokens explicitly. Therefore, the third catagroy extends the self-attention mechanism to consider the relative positions or distances between sequential elements. Such as the model by (Shaw et al., 2018) and Transformer-XL (Dai et al., 2019). Different from the relative positions metioned above, we propose the relative positions exspecially for entities. As such information is not necessary for non-entity tokens, and may introduce noise on the contrary.

3 Method

In this section, we briefly present the details of the relative position representation based multi-head selection framework. The concept of multi-head means that any head entity may be relevant to multiple relations and tail entities (Bekoulis et al., 2018).

Formally, denote $\mathcal{E}$ and $\mathcal{R}$ as the set of pre-defined entity types and relation categories, respectively. Given an input sentence with N tokens $\boldsymbol{s} = \{s_1, s_2, \ldots, s_N\}$, the entity-relation extraction task aims at extracting a set of named entities $\boldsymbol{e} = \{e_1, e_2, \ldots, e_M\}$ with specific types $\boldsymbol{y} = \{y_1, y_2, \ldots, y_M\}$, and predict the relation r_{ij} for each entity pair (e_i, e_j), where $y_i \in \mathcal{E}$ and $r_{ij} \in \mathcal{R}$. Triplets such as

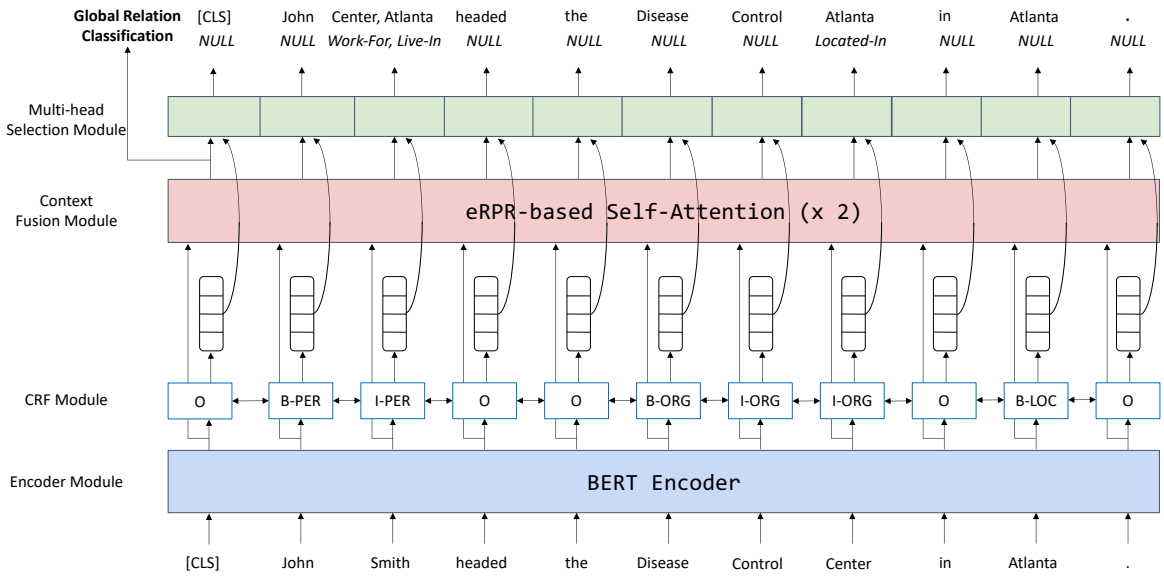


Figure 2: The overview of the relative position representation based multi-head selection framework. We take a sentence from CoNLL04 dataset as an example. In this sentence, the golden relational triplets are: (*John Smith*, `Live-In`, *Atlanta*), (*John Smith*, `Work-For`, *Disease Control Center*) and (*Disease Control Center*, `Located-In`, *Atlanta*). The *NULL* label denotes a case of no relation.

(e_i, r_{ij}, e_j) are formulated as the output, where e_i is the head entity and e_j is the tail entity, e.g., (*John Smith*, `Live-In`, *Atlanta*).

As illustrated in Fig. 2, our framework consists of four modules as follows: the encoder module, the CRF module, the context fusion module and the multi-head selection module. The token sequence is taken as the input of the framework and is fed into the BERT encoder to capture contextual representations. The CRF module is applied afterward to extract potential head entities (i.e., boundaries and types). Then, the hidden states of BERT and the entity information are feed into the context fusion module to encoder the entity position-based features. Finally, a multi-head selection module is employed to simultaneously extract tuples of relation and tail entity for the input token (e.g., (`Work-For`, *Center*) and (`Live-In`, *Atlanta*) for the head entity *Simth*). Additionally, we present the strategy of global relation classification. We will elaborate on each of the modules in the following subsections.

3.1 Encoder Module

The encoder module aims at mapping discrete tokens into distributed semantic representations. Bidirectioal Encoder Representations from Transformers (BERT) (Devlin et al., 2019) is a pre-trained language representations built on the bidirectional self-attentive models. It is known as its powerful feature representative ability and recently breaks through the leaderboards of a wide range of natural language processing tasks, such as named entity recognition, word segmentation and question answering. Different from the previous work (Bekoulis et al., 2018) which uses the BiLSTM as the feature encoder, we use the BERT instead to better represent contextual features.

As illustrated in Fig. 2, given a N-token sentence $\boldsymbol{s} = \{s_1, s_2, \ldots, s_N\}$, a special classification token ([CLS]) is introduced as the first token of the input sequence as $\{[\text{CLS}], s_1, s_2, \ldots, s_N\}$. The sequence is encoded by the multi-layer bidirectional attention structure. The output of the BERT layer is the contextual representation of each token as $\boldsymbol{h} = \{h_0, h_1, \ldots, h_N\}$ where $h_i \in \mathbb{R}^{d_h}$, where d_h denotes the dimension of the hidden state of BERT.

3.2 CRF Module

The conditional random field is a probabilistic method that jointly models interactions between entity labels, which is widely used in named Entity recognition task. Similarly, we employ a linear-chain CRF over the BERT layer to obtain the most possible entity label for each token, e.g., B-PER.

Given the BERT outputs $\boldsymbol{h} = \{h_0, h_1, \ldots, h_N\}$, the corresponding entity label sequence is denoted as $\boldsymbol{y} = \{y_0, y_1, \ldots, y_N\}$. Specifically, we uses the BIO (Begin, Inside, Non-Entity) tagging scheme. For example, B-PER denotes the beginning token of a person entity. The probability of using $\boldsymbol{y}$ as the label prediction for the input context is calculated as

$$p(\boldsymbol{y}|\boldsymbol{h}) = \frac{\prod_{i=1}^{N} \phi_i(y_{i-1}, y_i, \boldsymbol{h})}{\sum_{y' \in \mathcal{Y}(\boldsymbol{h})} \prod_{i=1}^{N} \phi_i(y'_{i-1}, y'_i, \boldsymbol{h})}. \tag{1}$$

Here, $\mathcal{Y}(\boldsymbol{h})$ is the set of all possible label predictions. And $\phi_i(y_{i-1}, y_i, \boldsymbol{h}) = \exp(\mathbf{W}_{\text{CRF}}^{y_i} h_i + \mathbf{b}_{\text{CRF}}^{y_{i-1} \to y_i})$, where $\mathbf{W}_{\text{CRF}} \in \mathbb{R}^{d_h \times d_l}$, $\mathbf{b}_{\text{CRF}} \in \mathbb{R}^{d_l \times d_l}$ with d_l denoting the size of the entity label set. $\mathbf{W}_{\text{CRF}}^{y_i}$ is the column corresponding to label y_i, and $\mathbf{b}_{\text{CRF}}^{y_{i-1} \to y_i}$ is the transition probability from label y_{i-1} to y_i.

During training, the NER loss function $\mathcal{L}_{\text{CRF}}$ is defined as the negative log-likelihood:

$$\mathcal{L}_{\text{NER}} = -\sum_{\boldsymbol{h}} \log p(\boldsymbol{y}|\boldsymbol{h}). \tag{2}$$

During decoding, the most possible label sequence y^* is the sequence with maximal likelihood of the prediction probability:

$$y^* = \arg\max_{\boldsymbol{y} \in \mathcal{Y}(\mathbf{h})} p(\boldsymbol{y}|\boldsymbol{h}). \tag{3}$$

The final labels can be efficiently addressed by the Viterbi algorithm.

3.3 Context Fusion Module

The context fusion module focuses on injecting the entity-specific relative position representation into the semantic feature of entities to capture the distance information between entities and other context tokens. The self-attention structure in BERT introduces sine and cosine functions of varying frequency to represent the absolute position representation (APR) of tokens as:

$$\begin{aligned} PE_{(pos,2i)} &= sin(pos/10000^{2i/d_{model}}) \\ PE_{(pos,2i+1)} &= cos(pos/10000^{2i/d_{model}}), \end{aligned} \tag{4}$$

where d_{model} stands for the hidden dimention of the model. However, such absolute position representation neglects the relative distance information between entities and other tokens, while such distance plays a crucial role in entity-relation prediction. Hence, we introduce an entity-specific relative position representation to efficiently encode the relative distance.

Formally, for the output states of BERT encoder $\boldsymbol{h} = \{h_0, h_1, \ldots, h_N\}$ where $h_i \in \mathbb{R}^{d_h}$, the relative position layer outputs a transformed sequence $\boldsymbol{p} = \{p_0, p_1, \ldots, p_N\}$ where $p_i \in d_p$ with d_p as the hidden dimention of self-attention structure.

Consider two input states h_i and h_j, where h_i denotes an entity and h_j denotes a contextual token, $i, j \in 0, 1, \ldots, N$. In order to inject the relative position information into x_i, we define $a_{ij}^K \in d_p$, $a_{ij}^V \in d_p$ as two different relative distances between h_i and h_j. Suppose that the impacts of tokens beyond a maximum distance on current token are negligible. Therefore, we clip the relative position within a maximum distance δ and only consider the position information of δ tokens on the left and δ tokens on the right. We define $\boldsymbol{\omega^K} = (\omega_{-\delta}^K, \ldots, \omega_{\delta}^K)$ and $\boldsymbol{\omega^V} = (\omega_{-\delta}^V, \ldots, \omega_{\delta}^V)$ as two relative position representations, where $\omega_i^K, \omega_i^V \in \mathbb{R}^{d_p}$ are initialized randomly and will be learned during training. Figure 3 illustrates an example of the relative position representations. Then, a_{ij}^K and a_{ij}^V are assigned as:

$$\begin{aligned} a_{ij}^K &= \omega_{\text{clip}(j-i,\delta)}^K \\ a_{ij}^V &= \omega_{\text{clip}(j-i,\delta)}^V \\ clip(x, \delta) &= \max(-\delta, \min(x, \delta)). \end{aligned} \tag{5}$$

Based on the relative position representations a_{ij}^K, a_{ij}^V, the attention matrix between h_i and h_j is calculated as:

$$\alpha_{ij} = \text{softmax}(\frac{(h_i W^Q)(h_j W^K + a_{ij}^K)^T}{\sqrt{d_p}}), \tag{6}$$

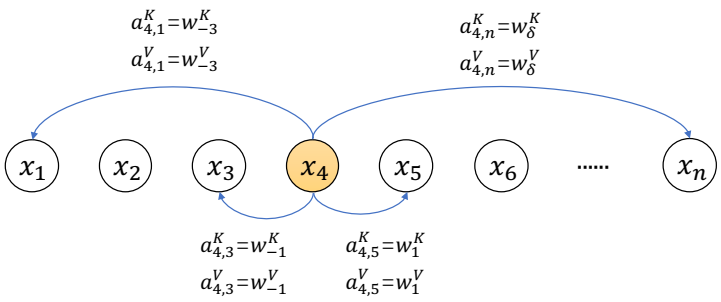


Figure 3: An example to illustrate the entity relative position representation. x_4 is considered as an entity, we show the eRPR between x_4 and the context tokens within the clipped distance δ. Assuming $3 <= \delta <= n-4$ in this example.

where $W^Q \in \mathbb{R}^{d_h \times d_p}$, $W^K \in \mathbb{R}^{d_h \times d_p}$ are parameter matrices for multi-head projections. The attentional output of h_i is the weighted sum of h_j which also consider the relative position:

$$p_i = \sum_{j=1}^{n} \alpha_{ij}(h_j W^V + a^V_{ij}). \tag{7}$$

Specifically, we only consider the relative position of named entities rather than every tokens in the sentence. So $\boldsymbol{\omega^K}$ and $\boldsymbol{\omega^V}$ are set as 0 for non-entity tokens. the This entity-only RPR approach comes with the following key advantages: 1) it encodes unique features for entities and thus can better differentiate entities from other plain tokens; 2) it provides entity-specific information and helps the relation and tail entity prediction.

3.4 Multi-head Selection Module

The multi-head selection module aims to predict the possible relations and tail entities simultaneously for each head entity (Bekoulis et al., 2018). Given a sequence of entity labels $\boldsymbol{y} = \{y_0, y_1, \ldots, y_N\}$ predicted by the CRF module, we map each label to a distributed label embedding as $\boldsymbol{l} = \{l_0, l_1, \ldots, l_N\}, l_i \in \mathbb{R}^{d_l}$, where d_l is the label embedding size. The mapping dictionary is randomly initialized and be fine-tuned during training. During training, we use the golden entity labels.

As shown in Fig. 2, the input to the multi-head selection layer are the concatentation of label embedding and the outputs of relative position layer as:

$$z_i = [l_i; p_i], i = 0, 1, \ldots, N. \tag{8}$$

For each input state z_i, we compute the score between z_i and z_j given a relation $r_k, r_k \in \mathcal{R}$ as:

$$g(z_i, z_j, r_k) = V^r f(U^r z_j + W^r z_i + b^r), \tag{9}$$

where $V^r \in \mathbb{R}^{d_r}$, $U^r, W^r \in \mathbb{R}^{d_r \times (d_h + d_l)}$, $b^r \in \mathbb{R}^{d_r}$, $f(\cdot)$ is the element-wise RELU function. The most probable tail entity s_j with the relation r_k corresponding to the head entity s_i is predicted as:

$$\Pr(\text{tail} = s_j, \text{relation} = r_k | \text{head} = s_i) = \sigma(g(z_i, z_j, r_k)), \tag{10}$$

where $\sigma(\cdot)$ denotes the sigmoid function.

During training, we optimize the cross-entropy loss $\mathcal{L}_{\text{MHS}}$ for the candidate tail entity s_{ij} and relation r_{ij} given the head entity s_i as:

$$\mathcal{L}_{\text{MHS}} = \sum_{i=0}^{N} \sum_{j=0}^{M} -\log \Pr(tail = s_j, relation = r_j | head = s_i), \tag{11}$$

where M is the number of golden relations for s_i. During testing, we select the tuple of the relation and tail entity $(\hat{r_k}, \hat{s_j})$ with a score exceeding the confidence threshold η. In this way, multiple tail entities and relations for the head entity s_i can be predicted simultaneously.

3.5 Global Relation Classification

Generally, detecting the relation between entites need to consider the theme of the sentence. The previous work only use the local context information for relation and entity prediction, which may lead to the deviation of global semantics. We introduce the global relation classification strategy to guide the training of local semantic features. As illustrated in Fig. 2, the first output of the relative position layer corresponding to the hidden state of [CLS] token p_0, which can be considered as the aggregate representation of the sentence. Therefore, we use the [CLS] token to predict the relations relevant to the whole sentence $\boldsymbol{s}$ as:

$$\Pr(\text{relation} = \boldsymbol{r}|\boldsymbol{s}) = \sigma(W^g p_0 + b^g), \tag{12}$$

where $\boldsymbol{r} \subseteq \mathcal{R}$, $W^g \in \mathbb{R}^{d_h \times |\mathcal{R}|}$, $b^r \in \mathbb{R}^{|R|}$, $\sigma(\cdot)$ is the sigmoid function.

During training, we minimize the binary cross-entropy loss for the global classification as:

$$\mathcal{L}_{\text{GRC}} = \sum_{i=0}^{T} \Pr(\text{relation} = \boldsymbol{r}|\boldsymbol{s}), \tag{13}$$

where T denotes the number of golden relations in the sentence.

3.6 Joint Training

To train the model jointly, we optimize the final combined objective function during training:

$$\mathcal{L} = \mathcal{L}_{\text{NER}} + \lambda \mathcal{L}_{\text{GRC}} + \mathcal{L}_{\text{MHS}}, \tag{14}$$

where $\mathcal{L}_{\text{NER}}$, $\mathcal{L}_{\text{GRC}}$, and $\mathcal{L}_{\text{MHS}}$ denote the loss function for head entity recognition, global relation classification and multi-head selection, respectively (Eq. 2, 13, 11), $\lambda \in [0, 1]$ is the weight controling the trade-off of the global relation classification. $\mathcal{L}$ is averaged over samples for each batch.

4 Experiment

In this section, we conduct extensive experiments to verify the effectiveness of our framework, and make detailed analyses to show its advantages.

4.1 Dataset

We evaluate the proposed method on two widely-used benchmarks for entity and relation extaction: CoNLL04 and ACE05.

- **CoNLL04** (Roth and Yih, 2004) defines 4 entity types including Location (`LOC`), Organization (`ORG`), Person (`PER`) and `Other` and 5 relation categories as `Located-In`, `OrgBased-In`, `Live-In`, `Kill` and `Work-For`. It consists of news articles from the Wall Street Journal and Associated Press. We use the data split by Gupta et al. (Gupta et al., 2016) (910 instances for training, 243 for validation and 288 for testing).

- **ACE05** (Doddington et al., 2004) provides 7 entity types: Location (`LOC`), Organization (`ORG`), Person (`PER`), Geopolitical Entity (`GPE`), Vehicle (`VEH`), Facility (`FAC`), Weapon (`WEA`) and 6 relation types: Organization affiliation (`ORG-AFF`), Person-Social (`PER-SOC`), Agent-Artifact (`ART`), `PART-WHOLE`, GPE affiliation (`GEN-AFF`), Physical (`PHYS`). It contains documents from different domains as newswire and online forums. We adopt the same data splits as the previous work (Miwa and Bansal, 2016) (351 documents for training, 80 for validation and 80 for testing).

4.2 Implemental Details

Following previous works, we use the standard precision (P), recall (R), and micro-F1 score (F1) as the evaluation metrics. A relation is correct if the arguments of triplet (e_1, r, e_2) are correct. Other

experimental settings are as follows. We initialize the BERT encoder layer using the pre-trained BERT-Base-Cased checkpoint [1] which has 12 layers, a hidden size of 768. We use Adam optimizer with an initial learning rate of 5×10^{-5}. During training, we do warm-up startup first and employ a linearly decrease with 0.05 as the decay rate. For the model structure, we adopt 2-layer eRPR-based self-attention after the BERT encoder layer. The self-attention layer has an identical structure as the layer in BERT. The relative position representations $\boldsymbol{\omega}^{\boldsymbol{K}}, \boldsymbol{\omega}^{\boldsymbol{V}}$ are initiaized randomly with a uniform distribution. The maximum relative distance is set as $\delta = 4$. The GRC loss weight is set as $\lambda = 1$. The size of entity label embedding is set as $d_l = 50$. The threshold for multi-head selection $\eta = 0.5$.

Specifically, we use the both the *relaxed* and *strict* evaluation settings for comparison. In the *relaxed* setting, assuming the entity boundaries are given, a multi-token entity is correct if at least one of its comprising token types is correct; a relation is correct if the two argument entities are correct and the relation type is correct. In the *strict* setting, we consider an entity is correct if the entity type and the boundaries are both correct; a relation is correct if the relation type and the argument entities are both correct.

4.3 Results and Analyses

Table 1: Performance comparision with baseline models on CoNLL04 and ACE05. eRPR denotes models adopt the self-attention with entity-specific relative position representation at the context fusion module. The ✓and ✗marks stand for whether or not the model builds on hand-crafted features or NLP tools. eRPR MHS is the proposed full model.

Model	Pre-calculated Features	Evaluation	Entity			Relation		
			P	R	F1	P	R	F1
CoNLL04								
Gupta et al. (2016)	✓	*relaxed*	92.50	92.10	92.40	78.50	63.00	69.90
Gupta et al. (2016)	✗	*relaxed*	88.50	88.90	88.80	64.60	53.10	58.30
Adel and Schütze (2017)	✗	*relaxed*	-	-	82.10	-	-	62.50
Bekoulis et al. (2018)	✗	*relaxed*	93.41	93.15	93.26	72.99	63.37	67.01
eRPR MHS	✗	*relaxed*	94.32	93.81	**94.06**	73.85	64.41	**68.81**
Miwa and Sasaki (2014)	✓	*strict*	81.20	80.20	80.70	76.00	50.90	61.00
Bekoulis et al. (2018)	✗	*strict*	83.75	84.06	83.90	63.75	60.43	62.04
eRPR MHS	✗	*strict*	86.85	85.62	**86.23**	64.20	64.69	**64.44**
ACE05								
Miwa and Bansal (2016)	✓	*strict*	80.80	82.90	81.80	48.70	48.10	48.40
Katiyar and Cardie (2017)	✗	*strict*	81.20	78.10	79.60	46.40	45.53	45.70
eRPR MHS	✗	*strict*	86.26	84.66	**85.45**	60.60	60.84	**60.72**

Comparison Baseline As shown in Table 1, we list the following baselines for comparison. Gupta et al. (2016) propose a table-filling based method that relies on hand-crafted features and external NLP tools. Adel and Schütze (2017) use a global normalized convolutional neural networks to extract entities and relations. Miwa and Bansal (2017) adopt a BiLSTM to extract entities and a Tree-LSTM to model the dependency relations between entities. Bekoulis et al. (2018) propose the multi-head selection structure, which adopts BiLSTM as the feature encoder and uses CRF for entity recognition and can extract the relational triplet simultaneously. The results on CoNLL04 and ACE05 are directly copied from the published paper.

Main Results Table 1 presents the performance comparisions on CoNLL04 and ACE05 datasets. eRPR MHS is the proposed full model, which uses the BERT at encoder module, and follows by two

[1] BERT checkpoints are available at https://github.com/google-research/bert

Table 2: Ablation study on CoNLL04 and ACE05. APR denotes models adopt the general self-attention with absolute position representation at the context fusion module. eRPR denotes models adopt the self-attention with entity-specific relative position representation at the context fusion module. The ✓mark refers to the model include the global relation classification. We use the *strict* evaluation setting here.

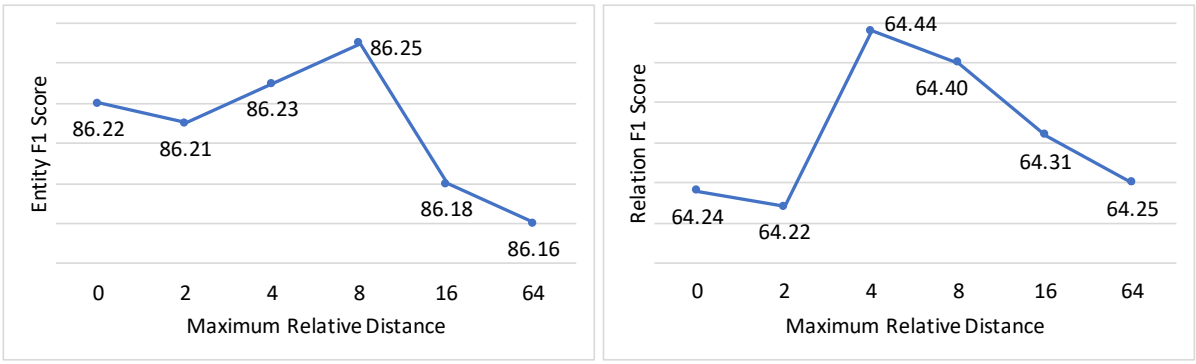
Model	Encoder	Context Fusion	GRC	Entity			Relation		
				P	R	F1	P	R	F1
CoNLL04									
1	BiLSTM	-	-	83.75	84.06	83.90	63.75	60.43	62.04
2	BERT	-	-	85.75	86.28	86.00	65.15	62.56	63.83
3	BERT	APR Layer ×2	-	86.32	85.68	86.00	64.53	63.40	63.96
4	BERT	eRPR Layer ×2	-	86.75	85.56	86.15	63.93	64.50	64.21
5	BERT	APR Layer ×2	✓	86.78	85.66	86.22	64.18	64.30	64.24
6	BERT	eRPR Layer ×2	✓	86.85	85.62	**86.23**	64.20	64.69	**64.44**
ACE05									
1	BiLSTM	-	-	84.88	84.10	84.49	57.40	60.32	58.82
2	BERT	-	-	85.70	84.25	84.96	59.92	60.06	59.99
3	BERT	APR Layer ×2	-	86.18	84.55	85.36	60.23	60.82	60.52
4	BERT	eRPR Layer ×2	-	86.24	84.60	85.41	60.57	60.76	60.66
5	BERT	APR Layer ×2	✓	86.22	84.57	85.39	60.46	60.76	60.61
6	BERT	eRPR Layer ×2	✓	86.26	84.66	**85.45**	60.60	60.84	**60.72**

eRPR self-attention layers and adopts the GRC strategy. As we can see, our eRPR MHS overwhelmingly outperforms all the baseline models in terms of all three evaluation metrics on the two datasets. by a large margin for both entity and relation extraction. Especially, comparing with the model by Bekoulis et al. (2018), our model achieves significant boosts by 2.40% and 1.90% for relation F1 on CoNLL04 and ACE05, respectively. These results show that, with our enhanced components, i.e., the eRPR layers, the global relation classification and the BERT encoder, the model performance can be significantly improved. Such improvements highlight the effectiveness of our proposed framework.

Ablation Study As shown in Table 2, we list variant models (Model 1-5) to each component in our framework. Model 1 stands for the original MHS framework proposed by Bekoulis et al. (2018). By comparison, we come to the following conclusions. 1) Replacing the BiLSTM with pre-trained BERT can improve the performance obviously (Model 2 v.s. Model 1). 2) Adding the context fusion module after the encoder module can enhance the semantic representation, leading to higher results (Model 3 v.s. Model 2). 3) Comparing Model 4 with the above variations, incorporating eRPR into the self-attention structure can significantly increase the precision of models and thus contribute to better overall F1 scores. For example, it increases the relation F1 from 63.96% to 64.21% on CoNLL04. We attribute it to that the eRPR injects distance information into entity features, which can provide useful information to the multi-head selection. 4) Comparing Model 5 and Model 4, the GRC strategy can further improve model performance. Therefore, global information is instructive for learning local features. Finally, combining all these components, we achieve significant improvements over the original MHS.

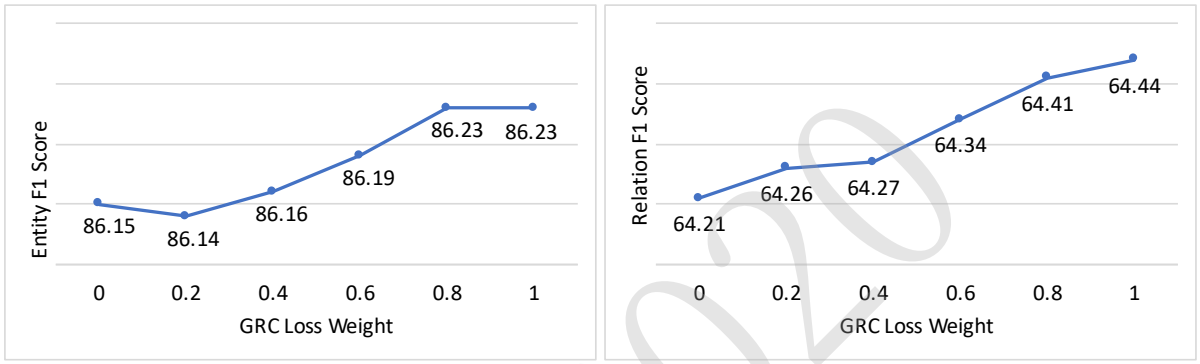
4.4 Effect of the Maximum Relative Distance

In this subsection, we evaluate the effect of varying the maximum relative distance δ. Following previous studies (Shaw et al., 2018), we conduct experiments on CoNLL04 with different maximum relative distance δ, increases exponentially from 0 to 64. Fig. 4 shows the experimental results. We observe that when $\delta = 8$, the entity F1 has the best result, and when $\delta = 4$, the relation F1 has the best result. Meanwhile, the larger value of δ (i.e., $\delta = 64$) is meaningless for both entity and relation extraction, which verifies that the impacts of tokens beyond a maximum distance can be negligible. Therefore, to ensure a better performance for relation extraction, we set $\delta = 4$ for all the experiements.



(a) Entity F1 Score (b) Relation F1 Score

Figure 4: Experimental results for varying the maximum relative distance δ.



(a) Entity F1 Score (b) Relation F1 Score

Figure 5: Experimental results for varying the GRC loss weight λ.

4.5 Effect of the GRC Loss Weight

In this subsection, we evaluate the effect of different GRC loss weight λ to the model performance. We keep the maximum relative distance δ as 4 and conduct the experiments on the CoNLL04 dataset with λ from 0 to 1 at the interval of 0.2. As shown in Fig. 5, the setting with $\lambda = 0$ denotes the GRC is not used in the framework and its performance is much lower than settings with larger λ In addition, with the growth of λ, both entity and relation F1 scores are increased continuously. As such, we keep $\lambda = 1$ for all the above experiments. These comparison results further demonstrate the effectiveness of GRC. Therefore, the sentence-level information can be utilized fruitfully for multi-head selection and helps improve the overall performance.

5 Conclusion

In this paper, we propose a relative position representation based multi-head selection framework for joint entity and relation extraction. Different with the existing multi-head selection method, we introduce the relative position representation to capture the distance information of entities. We then propose a global relation classification to guide the learning of local features. Additionaly, BERT is incorperated in the framework for sematic representation. Experimental results on CoNLL04 and ACE05 datasets show that our framework siginificantly outperforms all the baseline models for both entity and relation extraction.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (Grant Nos.U1636211, 61672081,61370126),the Beijing Advanced Innovation Center for Imaging Technology(Grant No.BAICIT-2016001), and the Fund of the State Key Laboratory of Software Development Environment (Grant No.SKLSDE-2019ZX-17).

References

Heike Adel and Hinrich Schütze. 2017. Global normalization of convolutional neural networks for joint entity and relation classification. In *EMNLP*, pages 1723–1729.

Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. 2018. Joint entity recognition and relation extraction as a multi-head selection problem. *Expert Systems with Applications*, 114:34–45.

Yee Seng Chan and Dan Roth. 2011. Exploiting syntactico-semantic structures for relation extraction. In *ACL*, pages 551–560. Association for Computational Linguistics.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *ACL*, pages 2978–2988.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186.

George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie M Strassel, and Ralph M Weischedel. 2004. The automatic content extraction (ace) program-tasks, data, and evaluation. In *Lrec*.

Markus Eberts and Adrian Ulges. 2019. Span-based joint entity and relation extraction with transformer pre-training. *arXiv preprint arXiv:1909.07755*.

Tsu-Jui Fu, Peng-Hsuan Li, and Wei-Yun Ma. 2019. Graphrel: Modeling text as relational graphs for joint entity and relation extraction. In *ACL*, pages 1409–1418.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *ICML*, pages 1243–1252.

Pankaj Gupta, Hinrich Schütze, and Bernt Andrassy. 2016. Table filling multi-task recurrent neural network for joint entity and relation extraction. In *COLING*, pages 2537–2547.

Weipeng Huang, Xingyi Cheng, Taifeng Wang, and Wei Chu. 2019. Bert-based multi-head selection for joint entity-relation extraction. In *NLPCC*, pages 713–723. Springer.

Arzoo Katiyar and Claire Cardie. 2017. Going out on a limb: Joint extraction of entity mentions and relations without dependency trees. In *ACL*, pages 917–928.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019. Entity-relation extraction as multi-turn question answering. *arXiv preprint arXiv:1905.05529*.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *ACL*, pages 2124–2133.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. In *NAACL*, pages 3036–3046.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. *arXiv preprint arXiv:1601.00770*.

Makoto Miwa and Yutaka Sasaki. 2014. Modeling joint entity and relation extraction with table representation. In *EMNLP*, pages 1858–1869.

Makoto Miwa, Rune Sætre, Yusuke Miyao, and Jun'ichi Tsujii. 2009. A rich feature vector for protein-protein interaction extraction from multiple corpora. In *EMNLP*, pages 121–130. Association for Computational Linguistics.

Dan Roth and Wen-tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. Technical report, ILLINOIS UNIV AT URBANA-CHAMPAIGN DEPT OF COMPUTER SCIENCE.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *NAACL*, pages 464–468.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448.

Changzhi Sun, Yeyun Gong, Yuanbin Wu, Ming Gong, Daxin Jiang, Man Lan, Shiliang Sun, and Nan Duan. 2019. Joint type inference on entities and relations via graph convolutional networks. In *ACL*, pages 1361–1370.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Haoyu Wang, Ming Tan, Mo Yu, Shiyu Chang, Dakuo Wang, Kun Xu, Xiaoxiao Guo, and Saloni Potdar. 2019. Extracting multiple-relations in one-pass with pre-trained transformers. *arXiv preprint arXiv:1902.01030*.

Tianyang Zhao, Zhao Yan, Yunbo Cao, and Zhoujun Li. 2020. Asking effective and diverse questions: A machine reading comprehension based framework for joint entity-relation extraction. In *IJCAI*.

A Mixed Learning Objective for Neural Machine Translation

Wenjie Lu, Leiying Zhou, Gongshen Liu* and Quanhai Zhang*
School of Electronic Information and Electrical Engineering,
Shanghai Jiao Tong University, Shanghai, China
{jonsey,zhouleiying,lgshen,qhzhang}@sjtu.edu.cn

Abstract

Evaluation discrepancy and overcorrection phenomenon are two common problems in neural machine translation (NMT). NMT models are generally trained with word-level learning objective, but evaluated by sentence-level metrics. Moreover, the cross-entropy loss function discourages model to generate synonymous predictions and overcorrect them to ground truth words. To address these two drawbacks, we adopt multi-task learning and propose a mixed learning objective (MLO) which combines the strength of word-level and sentence-level evaluation without modifying model structure. At word-level, it calculates semantic similarity between predicted and ground truth words. At sentence-level, it computes probabilistic n-gram matching scores of generated translations. We also combine a loss-sensitive scheduled sampling decoding strategy with MLO to explore its extensibility. Experimental results on IWSLT 2016 German-English and WMT 2019 English-Chinese datasets demonstrate that our methodology can significantly promote translation quality. The ablation study shows that both word-level and sentence-level learning objective can improve BLEU scores. Furthermore, MLO is consistent with state-of-the-art scheduled sampling methods and can achieve further promotion.

1 Introduction

In recent years, tremendous progresses have been made in the field of neural machine translation (NMT) (Sutskever et al., 2014; Luong et al., 2015). A typical NMT model can be formulated as an encoder-decoder-attention architecture (Forcada and Ñeco, 1997; Bahdanau et al., 2015) with maximum likelihood estimation (MLE) objective. Given sufficient parallel corpora, NMT models can achieve promising performance.

Despite much success, NMT models suffer from two major drawbacks. First, there exists a discrepancy between training objectives and evaluation metrics. Most NMT models are trained with MLE objective under the teacher forcing algorithm (Williams and Zipser, 1989), i.e., models calculate and accumulate cross-entropy loss between predicted and ground truth sentences word by word. A lower cross-entropy value means the predictions are closer to ground truth at word level. Model parameters are updated through backpropagation to minimize the value of loss function. However, translation quality is measured by sentence-level metrics such as BLEU (Shterionov et al., 2017), ROUGE (Lin, 2004), etc. This way of word-level optimization mismatches sentence-level evaluation metrics, which may mislead the promotion of translation performance. Second, the MLE training objective brings about overcorrection phenomenon (Zhang et al., 2019). To be specific, models are trained to learn absolutely correct translations and overcorrect synonymous words and phrases. Once the model predicts a word different from the ground truth word, the cross-entropy loss will immediately punish it and lead the model to the correct direction. As for synonymous phrases, it may result in translating wrong phrases while reducing the diversity of translation.

In this paper, we present a novel approach to solve the above problems. Instead of training NMT models with word-level cross-entropy loss, we propose to train models with a mixed learning objective

Proceedings of the 19th China National Conference on Computational Linguistics, pages 974-983,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

(MLO), which can combine the strength of word-level and sentence-level training. At word level, MLO estimates semantic similarity between the predicted and the ground truth words. Synonymous words will be encouraged rather than overcorrected. At sequence level, MLO calculates probabilistic n-gram matching score between the predicted and the ground truth sentences. The differentiable property of MLO enables NMT models to be trained flexibly without modifying structure. Most important of all, it can relieve the problem of evaluation discrepancy and overcorrection phenomenon.

The major contributions of this paper are summarized as follows:

- We present a novel mixed learning objective for training NMT models, aiming at alleviating evaluation discrepancy and overcorrection phenomenon. The mixed learning objective can encourage word-level semantic similarity and balance sequence-level n-gram precision of the translation.
- We explore the extensibility of mixed learning objective and adopt a novel loss-sensitive scheduled sampling instead of teacher forcing algorithm. The proposed objective is more consistent with state-of-the-art scheduled sampling methods and can achieve better performance.
- We demonstrate the effectiveness of our approach on IWSLT 2016 German-English and WMT 2019 English-Chinese datasets, and achieve significant improvements. Moreover, the mixed learning objective can be flexibly applied by various model structures and algorithms.

2 Related Work

2.1 Evaluation discrepancy

To tackle the problem of discrepancy between word-level MLE objective and sentence-level evaluation metrics, some researches utilize techniques like generative adversarial network (GAN) (Goodfellow et al., 2014) or reinforcement learning (RL) (Sutton et al., 1998). Borrowed idea from DAD (Venkatraman et al., 2015) and beam search (Sutskever et al., 2014; Rush et al., 2015), Ranzato et al. (2015) proposed Mixed Incremental Cross-Entropy Reinforce (MIXER) to directly optimized model parameters with respect to the metric used at inference time. Further, Shen et al. (2016) presented minimum risk training (MRT) to minimize the expected loss (i.e., risk) on the training data. Wieting et al. (2019) proposed to train NMT models with semantic similarity based on MRT. Wiseman and Rush (2016) introduced beam-search optimization schedule for model to learn global sequence scores. Moreover, Lin et al. (2017) proposed RankGAN which can analyze and rank sentences by giving a reference group, and thus achieve high-quality language descriptions.

2.2 Overcorrection Phenomenon

As for overcorrection phenomenon, especially synonymous phrases, one solution is to utilize the model's previous predictions as input in training. The generation inconsistency between training and inference which called exposure bias (Zhang et al., 2019) causes models to overcorrect from synonymous translations and generate wrong phrases. Bengio et al. (2015) firstly proposed a scheduled sampling strategy based on an algorithm called Data As Demonstrator (DAD) (Venkatraman et al., 2015). At every decoding step, a dynamic probability p is used to decide whether to sample from ground truth or the previous word predicted by the model itself. Inspired by their method, Zhang et al. (2019) came up with sampling from ground truth and inferred sentences word by word through force decoding.

3 Methodology

3.1 Model Overview

Without loss of generality, we utilize a common RNN attention model (Bahdanau et al., 2015) as baseline to demonstrate our approach. Suppose that the source sentence $X = (x_1, x_2, ..., x_{T_x})$ and the target sentence $Y = (y_1, y_1, ..., y_{T_y})$. The RNN model encodes the source sentence as follows:

$$h_t = \phi(h_{t-1}, x_t) \tag{1}$$

where h_0 is an initial vector and ϕ is a nonlinear function. Then context vector $c_i, i = 1, 2, ..., T_y$ is calculated by:

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} \cdot h_j \tag{2}$$

where α_{ij} is the attention weight between c_i and h_j.

When the decoder receives the context c_t, it calculates the hidden layer vector s_t by:

$$s_t = f(s_{t-1}, y_{t-1}, c_t) \tag{3}$$

where s_0 is an initial vector, f is a nonlinear function of hidden layers, y_{t-1} is the historical output at time $t-1$ in inference and ground truth word in training, and y_0 is the end flag of source sentence X.

According to the hidden layer state s_t, the probability of inferring the word y_t can be computed by:

$$P(y_t) = softmax(W_o \cdot p(y_t \mid y_1, ..., y_{t-1}, x)) \tag{4}$$

$$p(y_t \mid y_1, ..., y_{t-1}, x) = g(y_{t-1}, s_t, c_t) \tag{5}$$

where g is a nonlinear function and W_o is a mapping matrix.

Finally, given a set of sequence pairs $(X_i, Y_i), i = 1, 2, ..., N$ in the parallel corpora, the training objective is to maximize the likelihood as follows:

$$\hat{\theta}_{MLE} = argmax\{L(\theta)\} \tag{6}$$

where $L(\theta)$ is the loss function computed by:

$$L(\theta) = \sum_{i=1}^{N} logP(Y_i \mid X_i, \theta) = \sum_{i=1}^{N} \sum_{t=1}^{T_y} logP(y_t) \tag{7}$$

3.2 Word-level Semantic Similarity Objective

The original cross-entropy loss measures the probability of predicting right translation for each word, which means it only cares about how to generate ground truth words with maximum likelihood. This may cause two problems. First, generating any other words is discouraged. Although synonymous translations are right in the subjective sense, they will be punished and corrected to ground truth words. Second, suppose that the word with maximum probability is not ground truth word, and the model will choose it as predicted translation. The calculation of cross-entropy loss does not take into consideration of what exactly that word is, which is important for evaluating the model.

Therefore, we design the word-level learning objective in order to measure the semantic similarity between the generated translations and ground truth sentences. There have been lots of complex researches on semantic similarity (Pradhan et al., 2015; Kenter and De Rijke, 2015). In order not to include additional models, we adopt the cosine similarity method for measurement. Mathematically speaking, cosine similarity calculates the semantic similarity between two non-zero vectors, which is suitable for word embeddings.

Given the predicted translation $Y^* = (y_1^*, y_2^*, \ldots, y_{T_{Y^*}}^*)$, the semantic similarity between sentence Y and Y^* can be calculated by:

$$Sim(Y, Y^*) = \sum_{i=1}^{T_y} \frac{emb(y_i) \cdot emb(y_i^*)}{\|emb(y_i)\| \times \|emb(y_i^*)\|} \tag{8}$$

where $emb(\cdot)$ refers to the word embedding of each word.

Therefore, we can calculate semantic similarity between every translation and corresponding ground truth sentence. During training, the word-level training objective is defined as followings:

$$L_{word} = -\sum_{j=1}^{N} Sim(Y_j, Y_j^*) \tag{9}$$

3.3 Sentence-level Probabilistic N-gram Objective

The word-level semantic similarity objective helps to foster translation diversity and relieve the problem of overcorrection, which can improve word-level translation accuracy. As for another important standard fluency in machine translation, we design a sentence-level probabilistic n-gram objective which is consistent with evaluation metrics.

The calculation of n-gram matching is widely used in machine translation evaluation metrics. Take BLEU for example, firstly n-grams in source sentence Y and Y^* are extracted and counted, denoted as $C(n-gram)$. Next, n-gram matches between Y and Y^* are computed and denoted as $C_{clip}(n-gram)$. The precision score can be calculated by their ratio.

However, the non-differentiable property of BLEU makes it unable to be adopted as loss function. Therefore, inspired by Shao et al. (2018), we modified the calculation of n-gram matches as follows. Supposing that $(g_1, g_2, ..., g_n)$ is an n-gram sequence in Y, then its occurrences can be computed by:

$$\widetilde{C}_Y(n-gram) = \sum_{i=0}^{T_Y-n} \prod_{j=1}^{n} 1\{g_j = y_{i+j}\} \cdot P(y_{i+j}) \tag{10}$$

where $1\{\cdot\}$ denotes an indicator function and $P(\cdot)$ is calculated by equation (5). Then, the clip n-gram matches between two sentences and the precision score of translation Y can be computed as follows:

$$C_{clip}(n-gram) = min\{\widetilde{C}_Y(n-gram), C_{Y^*}(n-gram)\} \tag{11}$$

$$\widetilde{p_n} = \frac{\sum\limits_{n-gram \in Y} C_{clip}(n-gram)}{\sum\limits_{n-gram' \in Y} \widetilde{C}_Y(n-gram')} \tag{12}$$

Finally, to punish very long or short translations, BLEU is modified based on $\widetilde{p_n}$ and defined as follows:

$$\widetilde{BLEU}(Y, Y^*) = BP \cdot exp(\sum_{n=1}^{N} w_n log\widetilde{p_n}) \tag{13}$$

where BP is brevity penalty, w_n is positive weights and N is the maximum length of n-gram.

Therefore, we can calculate probabilistic n-gram matching score between every translation and corresponding ground truth sentence. During training, the sentence-level training objective is defined as followings:

$$L_{sent} = -\sum_{j=1}^{N} \widetilde{BLEU}(Y_j, Y_j^*) \tag{14}$$

3.4 Mixed Learning Objective

In order to alleviate the problem of evaluation discrepancy and overcorrection phenomenon, we propose the mixed learning objective. At word level, it can calculate semantic similarity for training evaluation and promote translation diversity. At sentence level, it can compute probabilistic n-gram precision of predicted sentence and promote translation fluency. The mixed learning objective is defined as follows:

$$L_{total} = L_{ce} + \alpha_{word} \cdot L_{word} + \alpha_{sent} \cdot L_{sent} \tag{15}$$

where L_{ce} refers to cross-entropy loss, α_{word} is the weight of word-level loss function and α_{sent} is the weight of sentence-level loss function.

Similar to Zhang et al. (2019), we adopt the Gumbel-Max technique (Gumbel, 1954; Maddison et al., 2014) for generating more robust outputs. To be specific, the Gumbel noise is defined as follows:

$$G = -log(-logU) \tag{16}$$

where $U \sim Unif[0, 1]$.

Then equation (5) is modified to:

$$P(y_t) = softmax(\frac{W_o \cdot p(y_t \mid y_1, ..., y_{t-1}, x) + G}{\tau}) \tag{17}$$

where τ is a temperature parameter controlling the generated distribution.

During training, we adopt a scheduled sampling strategy instead of teacher forcing algorithms. At every decoding step, a probability p is used to decide whether to sample from ground truth or inferred words. Specifically, assuming w_t is the input at each decoding step t and y'_{t-1} is the word obtained from inferred words, then $Pr(w_t = y_{t-1}) = p$ and $Pr(w_t = y'_{t-1}) = 1 - p$. We hope the probability p to decay from 1 to 0, so that the training process can gradually learned to deal with simulated inference situation.

Borrowing idea from the decay schedule in learning rate, sample probability can be defined as an inverse sigmoid curve with variable training epochs. Considering that a loss function intuitively reflects how well the model is trained, we define loss-sensitive sample probability as follows:

$$p = \frac{k}{k + exp(\frac{e}{k})} \cdot \sigma(L) \tag{18}$$

where k is a hyper-parameter, e is the current index of epoch, L is the average loss function value of epoch e, and σ is a non-linear function. For practice, we choose tanh function.

4 Experiments

4.1 Experimental Setup

We conduct our experiments comparable with previous work by using the following two datasets:

German-English. The German-English dataset is chosen from IWSLT 2016 (Cettolo et al., 2012). We use official testset2013 as validation set. The training and validation data consists of 196,884 and 992 sentences respectively. As for evaluation, we use the testset dataset from 2010 to 2014 and tokenized BLEU scores as computed by the multi-bleu.perl script[0].

English-Chinese. The English-Chinese dataset is chosen from the casia2015 parallel corpus in WMT 2019 shared task. It consists of approximately 1.05M sentences. We use official newsdev2017 as validation set and evaluate on the newstest dataset from 2017 to 2019.

For all training data, we perform tokenization and truecasing using standard Moses tools. For Chinese corpora, we use jieba[1] for segmentation. Then, we employ byte pair encoding (BPE) (Sennrich et al., 2016) with 50,000 operations to alleviate Out-of-Vocabulary problem. To accelerate training and save cost, we discard sentences with more than 50 tokens. The dimension of word embeddings is set to 512.

We first pretrain the baseline model by MLE. Then, we replace the cross-entropy loss function with MLO. The model is trained with a batch size of 60. We use Adam (Kingma and Ba, 2014) optimizer to tune the parameters. Besides, we use dropout regularization with a drop probability 0.5. During decoding, the beam size is set to 3. The hyper-parameter of sample probability k and temperature τ are set to 12 and 0.5 respectively. The weights of word-level and sentence-level loss function are set to 0.8 and 150 respectively.

4.2 Baseline Systems

We compare our method with existing common NMT systems including Transformer(Vaswani et al., 2017), Evolved Transformer (So et al., 2019) and DTMT (Meng and Zhang, 2019). Moreover, to explore the extensibility of MLO, we experiment on several state-of-the-art scheduled sampling works. These baseline systems are included as follows:

RNNsearch. A vanilla attention-based recurrent neural network which consists of 2-layer bidirectional GRU units (Cho et al., 2014). The dimension of hidden layer is 512.

[0] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl

[1] https://github.com/fxsjy/jieba

SS-NMT. A word-level scheduled sampling method (Bengio et al., 2015) which utilizes an inverse sigmoid decay schedule to sample from previous predicted word and ground truth word.

OR-NMT. A sentence-level scheduled sampling method (Zhang et al., 2019) which utilizes inverse sigmoid decay schedule to sample from predicted sentence and ground truth sentence. Predicted sentence is generated by beam search and force decoding.

4.3 Main Results

Table 1: Results of the proposed method on German-English dataset (BLEU).

Systems	testset2010	testset2011	testset2012	testset2014	average
Transformer	25.17	30.03	26.20	24.24	26.41
Evolved Transformer	26.33	31.45	27.28	25.36	27.61
DTMT	26.51	31.66	27.64	26.02	27.96
RNNsearch	24.46	28.06	24.92	22.94	25.10
+ SS-NMT	26.46	30.14	26.60	24.31	26.88
+ OR-NMT	27.37	30.72	27.54	25.20	27.71
+ MLO	25.84	29.85	26.32	23.73	26.44
+ SS-NMT + MLO	26.78	30.34	26.99	24.81	27.23
+ OR-NMT + MLO	**27.44**	**31.89**	**27.65**	**25.92**	**28.22**

Table 2: Results of the proposed method on English-Chinese dataset (BLEU).

Systems	newstest2017	newstest2018	newstest2019	average
Transformer	26.37	25.09	25.76	25.74
Evolved Transformer	27.84	25.98	27.25	27.02
DTMT	28.07	26.10	27.34	27.17
RNNsearch	24.92	24.17	24.20	24.63
+ SS-NMT	25.89	25.12	25.43	25.48
+ OR-NMT	28.03	26.10	26.66	26.93
+ MLO	25.83	24.74	25.32	25.29
+ SS-NMT + MLO	26.60	25.42	25.63	25.88
+ OR-NMT + MLO	**28.18**	**26.63**	**27.13**	**27.31**

TABLE 1 and TABLE 2 reports the results of the proposed method in comparison to other NMT systems on German-English and English-Chinese datasets respectively. As it can be seen, training with sentence-level scheduled sampling and mixed learning objective (OR-NMT + MLO) obtains the best published results on all testsets.

On German-English dataset, our full system can outperform RNNsearch by +3.11 BLEU averagely. On English-Chinese dataset, our full system can have an improvement of +2.68 BLEU on three testsets.

To validate the effectiveness of mixed learning objective, we carry out ablation study to evaluate the performance of word-level and sentence-level learning objective respectively. The mixed learning objective is proposed to encourage word-level semantic similarity and balance sequence-level n-gram precision of the translation. Meanwhile, it aims at relieving the problem of evaluation discrepancy and overcorrection. We will display and analyze the effect of mixed learning objective in detail in Section 4.4.

Another point of focus lies in the extensibility of mixed learning objective. As shown in TABLE 1 and TABLE 2, combining scheduled sampling strategy with mixed learning objective can achieve better translation performance. We will discuss the effect of scheduled sampling from two aspects in Section

Table 3: BLEU scores on German-English dataset.

Systems	testset2010	testset2011	testset2012	testset2014	average
RNNsearch	24.46	28.06	24.92	22.94	25.10
+ L_{word}	25.18	28.57	25.74	23.83	25.83
+ L_{sent}	25.40	28.72	26.01	24.07	26.05
+ MLO	25.84	29.85	26.32	23.73	26.44

Table 4: BLEU scores on English-Chinese dataset.

Systems	newstest2017	newstest2018	newstest2019	average
RNNsearch	24.92	24.17	24.20	24.63
+ L_{word}	25.26	24.45	24.67	24.79
+ L_{sent}	25.59	24.51	25.10	25.06
+ MLO	25.83	24.74	25.32	25.29

4.5. Besides, the loss-sensitive sample probability is defined to sense the speed of converge and make adjustment on sample probability. We will analyse its effect on scheduled sampling methods to explore how to achieve better performance.

4.4 Effect of Mixed Learning Objective

Aiming to alleviate evaluation discrepancy and overcorrection phenomenon, we propose the mixed learning objective which can promote word-level semantic similarity and sequence-level n-gram precision. To explore the effect of mixed learning objective, we conduct experiments on word-level and sentence-level learning objective respectively without scheduled sampling strategy on RNNsearch under the same conditions.

The experimental results are listed in TABLE 3 and TABLE 4. As it can be seen, only using word-level or sentence-level learning objective rather than cross-entropy loss can help achieve higher BLEU scores on two datasets. To be specific, word-level learning objective can get a promotion of +0.16 $\sim$ +0.73 BLEU averagely over RNNsearch on German-English and English-Chinese datasets. Sentence-level learning objective can outperform RNNsearch by +0.43 $\sim$ +0.95 BLEU score on two datasets averagely.

For the experimental results, we make some simple analysis. The word-level learning objective takes into account semantic similarity between predicted and ground truth words, so that it can avoid forcing model to generate the only one correct translation. The promotion in BLEU scores verifies that discouraging and punishing other synonymous words is disadvantageous for NMT models. Therefore, the word-level learning objective can to some extent solve this problem and encourage translation diversity.

The sentence-level learning objective calculates probabilistic n-gram matching scores between predicted and ground truth sentence, which is coordinate with general evaluation metrics. On the one hand, it contributes to alleviate the problem of evaluation discrepancy without importing additional complex model. On the other hand, the objective can naturally promote translation performance on BLEU scores.

Furthermore, MLO which combines word-level and sentence-level learning objective can obtain best translation performance in BLEU scores. Specifically, MLO can outperform RNNsearch by +0.66 $\sim$ +1.34 BLEU score averagely on German-English and English-Chinese datasets.

4.5 Effect of Loss-sensitive Scheduled Sampling

To validate the extensibility of MLO, we conduct various experiments which combine MLO with state-of-the-art scheduled sampling methods. Moreover, we defined a novel loss-sensitive sample probability for model to be flexibly adapted to scheduled sampling strategy. Under the same experimental settings, we conduct experiments on German-English and English-Chinese datasets to validate the effectiveness of loss-sensitive scheduled sampling and analyze in two aspects.

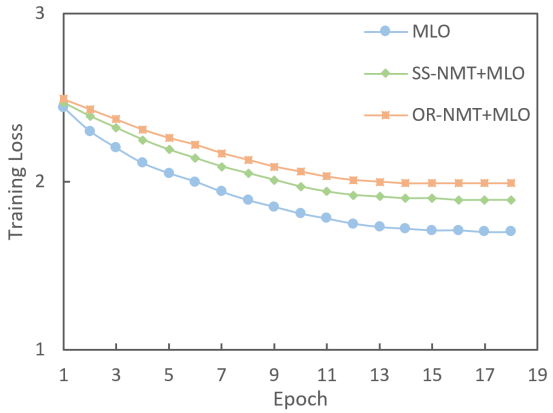


Figure 1: The training loss curves of three baseline systems on the IWSLT 2016 German-English translation task.

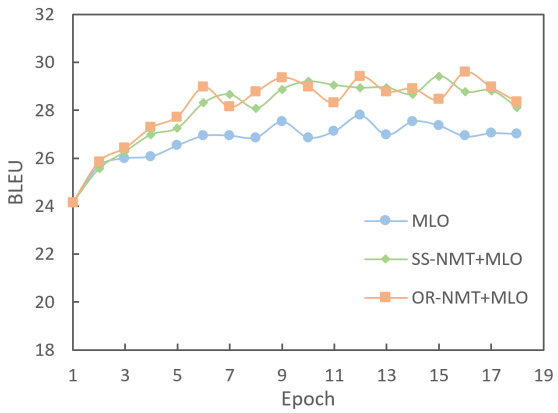


Figure 2: Trends of BLEU scores of three baseline systems on the validation set on the IWSLT 2016 German-English translation task.

Fig. 1 gives the training loss curves of MLO, SS-NMT+MLO and OR-NMT+MLO during training. As the training epoch increases, MLO continues to decrease at the lowest value and gradually tends to be flat. Due to different sampling strategies, SS-NMT and OR-NMT gradually converge to a certain training loss value. Moreover, Fig. 2 gives the BLEU score curves of three methods. It can be seen that SS-NMT+MLO and OR-NMT+MLO can achieve better BLEU scores compared to RNNsearch on validation set. We can also conclude from TABLE 1 and TABLE 2 that SS-NMT+MLO can achieve a promotion of +0.35 $\sim$ +0.4 BLEU scores over SS-NMT and OR-NMT+MLO can outperform OR-NMT by +0.38 $\sim$ +0.51 BLEU scores.

Since the starting point of scheduled sampling is to solve the problem of exposure bias and overcorrection phenomenon, the original cross-entropy loss function may be hard to score the inference results and guide the training process. However, the MLO is proposed for alleviating these problems as well. Therefore, the idea of combining MLO with scheduled sampling is natural and proved to be effective.

Besides the mutual promotion of MLO and scheduled sampling, the last thing we want to point out is the necessity and effectiveness of loss-sensitive sample probability. We define $\sigma(L) = 1$ as non-sensitive sample probability and perform parallel tests. By observing their decay curves during training, we find that loss-sensitive sample probability is more flexible and helpful in adjusting a proper probability for different training scenes. Since $tanh(L) < 1$, the loss-sensitive probability is calculated to be lower than non-sensitive probability. From the perspective of feeding as input inferred rather than ground truth words, we make it harder for model to learn and correct mistakes. Meanwhile, the experimental results show promotion on translation quality.

5 Conclusion

In this paper, we propose a mixed learning objective for NMT so as to alleviate the problem of evaluation discrepancy and overcorrection phenomenon. At word-level, the objective measures semantic similarity between the generated and ground truth words. At sentence-level, the objective calculates probabilistic n-gram matching scores of the translations. Moreover, we combine loss-sensitive scheduled sampling methods with mixed learning objective for mutual promotion. Experimental results show that our proposed method can achieve significant improvement on BLEU scores compared to previous works.

Acknowledgment

This research work has been funded by the National Natural Science Foundation of China (Grant No.61772337), the National Key Research and Development Program of China NO. 2018YFC0830803.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In Yoshua Bengio and Yann LeCun, editors, *ICLR*.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *NIPS*, pages 1171–1179.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. Wit^3: Web inventory of transcribed and translated talks. In *EAMT*, pages 261–268, Trento, Italy.

Kyunghyun Cho, Bart van Merriënboer Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *EMNLP*, pages 1724–1734.

Mikel L Forcada and Ramón P Ñeco. 1997. Recursive hetero-associative memories for translation. In *IWANN*, pages 453–462. Springer.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *NIPS*, pages 2672–2680.

Emil Julius Gumbel. 1954. Statistical theory of extreme values and some practical applications. *NBS Applied Mathematics Series*, 33.

Tom Kenter and Maarten De Rijke. 2015. Short text similarity with word embeddings. In *CIKM*, pages 1411–1420.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kevin Lin, Dianqi Li, Xiaodong He, Zhengyou Zhang, and Ming-Ting Sun. 2017. Adversarial ranking for language generation. In *NIPS*, pages 3155–3165.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*, pages 1412–1421.

Christopher Maddison, Daniel Tarlow, and Tom Minka. 2014. A* sampling. *NIPS*, 10.

Fandong Meng and Jinchao Zhang. 2019. Dtmt: A novel deep transition architecture for neural machine translation. In *AAAI*, volume 33, pages 224–231.

Nitesh Pradhan, Manasi Gyanchandani, and Rajesh Wadhvani. 2015. A review on text similarity technique used in ir and its application. *International Journal of Computer Applications*, 120(9).

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *EMNLP*, pages 379–389.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *ACL*, pages 1715–1725.

Chenze Shao, Yang Feng, and Xilin Chen. 2018. Greedy search with probabilistic n-gram matching for neural machine translation. *arXiv preprint arXiv:1809.03132*.

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *ACL*, pages 1683–1692.

Dimitar Shterionov, Pat Nagle, Laura Casanellas, Riccardo Superbo, and Tony O'Dowd. 2017. Empirical evaluation of nmt and pbsmt quality for large-scale translation production. In *EAMT: User Track*.

David So, Quoc Le, and Chen Liang. 2019. The evolved transformer. In *ICML*, pages 5877–5886.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112.

Richard S Sutton, Andrew G Barto, et al. 1998. *Introduction to reinforcement learning*, volume 2. MIT press Cambridge.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.

Arun Venkatraman, Martial Hebert, and J Andrew Bagnell. 2015. Improving multi-step prediction of learned time series models. In *AAAI*, pages 3024–3030.

John Wieting, Taylor Berg-Kirkpatrick, Kevin Gimpel, and Graham Neubig. 2019. Beyond bleu: Training neural machine translation with semantic similarity. In *ACL*, pages 4344–4355.

Ronald J Williams and David Zipser. 1989. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280.

Sam Wiseman and Alexander M Rush. 2016. Sequence-to-sequence learning as beam-search optimization. In *EMNLP 2016*, pages 1296–1306.

Wen Zhang, Yang Feng, Fandong Meng, Di You, and Qun Liu. 2019. Bridging the gap between training and inference for neural machine translation. In *ACL*, pages 4334–4343.

Multi-Reward based Reinforcement Learning for Neural Machine Translation

Shuo Sun[†] , **Hongxu Hou**[*], **Nier Wu**[†] , **Ziyue Guo**[†] ,**Chaowei Zhang**[†]

[†,*]College of Computer Science-college of Software, Inner Mongolia University,China
[*]cshhx@imu.edu.cn
[†]{sunshuo07,wunier04,guoziyue08,zhangchaowei08}@126.com

Abstract

Reinforcement learning (RL) has made remarkable progress in neural machine translation (NMT). However, it exists the problems with uneven sampling distribution, sparse rewards and high variance in training phase. Therefore, we propose a multi-reward reinforcement learning training strategy to decouple action selection and value estimation. Meanwhile, our method combines with language model rewards to jointly optimize model parameters. In addition, we add Gumbel noise in sampling to obtain more effective semantic information. To verify the robustness of our method, we not only conducted experiments on large corpora, but also performed on low-resource languages. Experimental results show that our work is superior to the baselines in WMT14 English-German, LDC2014 Chinese-English and CWMT2018 Mongolian-Chinese tasks, which fully certificates the effectiveness of our method.

1 Introduction

Neural machine translation (NMT) (Bahdanau et al., 2015; Wu et al., 2018a; Yang et al., 2018) has drawn universal attention without the demand of numerous manual work. In training phase, generic NMT models employ maximum likelihood estimation (MLE) (Harris and Mandelbaum, 1985), which is the token-level objective function. However, it is inconsistent with sequence-level evaluation metrics such as BLEU (Papineni et al., 2002). Reinforcement learning (RL) are leveraged for sequence generation tasks including NMT to optimize sequence-level objectives, such as Actor-Critic (Bahdanau et al., 2017) and Minimum Risk Training (MRT) (Shen et al., 2016). In machine translation community, Wu el at. (Wu et al., 2018a) provide the first comprehensive study of different aspects of RL training, they set a single reward to mitigate the inconsistency, and combine MLE with RL to stabilize the training process. Nevertheless, NMT based on reinforcement learning (RL) is unable to guarantee that the machine-translated sentences are as natural, sufficient and accurate as reference. To obtain smoother translation results, generative adversarial network (GAN) and deep reinforcement learning (DRL) (Wu et al., 2018b) are employed to NMT. And (Yang et al., 2018) utilizes sentence-level BLEU Q as a reinforcement target based on the work of (Wu et al., 2018b) to enhance the capability of the generator.

Although this nova machine translation learning paradigm based on GAN and DRL reveals excellent manifestation, there are still some limitations: (1) when calculating rewards, the overestimation of Q value will give rise to a suboptimal strategy update. (2) during training phase, it exists the problems with uneven sampling distribution, sparse rewards and high variance. What's more, the generator uses Monte Carlo to simulate the entire sentence, but it usually requires more calculation steps, resulting in too many parameters. (3) traditional NMT usually utilizes deterministic algorithms such as Beam Search or Greedy Decoding when predicting the next token. These methods lacks randomness, which may cause the potential best solution to be discarded.

In this paper, we propose some measures to address the above problems. Foremost, we adopt a novel multi-reward reinforcement learning method. That is, we weighted sum the actual reward of the discriminator, the language model reward and the sentence BLEU to obtain the total reward. Among them,

Proceedings of the 19th China National Conference on Computational Linguistics, pages 984-993,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

we adopt reward shaping to alleviate the sparse reward when calculating sentence rewards. Nextly, our method employ Temporal-Difference Learning (TD) (Sutton, 1988) to simulate the entire sentence. It effectively speeds up training and relieves the problem of error accumulation. Finally, we adopt Gumbel-Top-K Stochastic Beam Search (Kool et al., 2019) to predict the next token. The method trains model more efficiently by adding the noise obeying Gumbel distribution to control random sampled noise. Experiments on the datasets of the English-German, Chinese-English and Mongolian-Chinese translation tasks reveal our approach outperforms the best published results. In summary, we mainly made the following contributions:

- It is the first time that duel reward has been applied to neural machine translation. This method is applicable to arbitrary end-to-end NMT system.
- Our generator to optimize reward by using Gumbel-Top-K Stochastic Beam Search to sample different samples and Temporal-Difference Learning to simulate sentences.
- In English-German and Chinese-English translation tasks, we tested two different NMT models: RNNSearch and Transformer. Experimental results reveal that our proposed method performs well.

2 Background & Related Work

Common NMT models are based on an encoder decoder architecture. The encoder reads and encodes the source language sequence $X = (x_1, ..., x_n)$ into the context vector representation, and the decoder generates the corresponding target language sequence $\hat{Y} = (\hat{y}_1, ..., \hat{y}_m)$. Given H training sentence pairs $\{x^i, y^i\}_{i=1}^{H}$, at each timestep t, NMT is trained by maximum likelihood estimation(MLE) and generates the target words $\hat{y}_t$ by maximum the probability of translation conditioned on the source sentence X. The training goal is to maximize:

$$L_{MLE} = \sum_{i=1}^{H} logp(\hat{y}^i|x^i) = \sum_{i=1}^{H}\sum_{t=1}^{m} logp(\hat{y}_t^i|\hat{y}_1^i...\hat{y}_{t-1}^i, x^i) \tag{1}$$

where m is the length of sentence $\hat{y}^i$.

According (Williams, 1992), reinforcement learning enables NMT to optimize evaluation during training and usually estimates the overall expectation by sampling $\hat{y}$ with policy $p(\hat{y}|x)$. The training objective of RL is to maximize the expected reward:

$$L_{RL} = \sum_{i=1}^{H} R\left(\hat{y}^i, y^i\right), \hat{y}^i \sim p\left(\hat{y}|x^i\right), \forall_i \in [H]. \tag{2}$$

where $R(\hat{y}, y)$ is the final reward calculated by BLEU after generating the complete sentence $\hat{y}$. To increase stationarity, we combine the two simple linearly:

$$L_{COM} = \mu \times L_{MLE} + (1 - \mu) \times L_{RL} \tag{3}$$

where μ is the hyperparameter to control the balance between MLE and RL. L_{COM} is the strategy to stabilize RL training progress.

(Yang et al., 2018) proposed the BLEU reinforced conditional sequence generative adversarial net (BR-CSGAN) on the basis of reinforcement learning. The specific process is that generator G generates the target sentence based on the source sentence, and discriminator D detects whether the given sentence is groundtruth. During training status, G attempts to deceive discriminator D into believing that the generated sentence is groundtruth. The D strives to improve its anti-spoofing ability to distinguish machine-translated sentences from groundtruth. When G and D reach the Nash balance, the training results achieve the optimal state, and utilize BLEU to guide the learning of the generator.

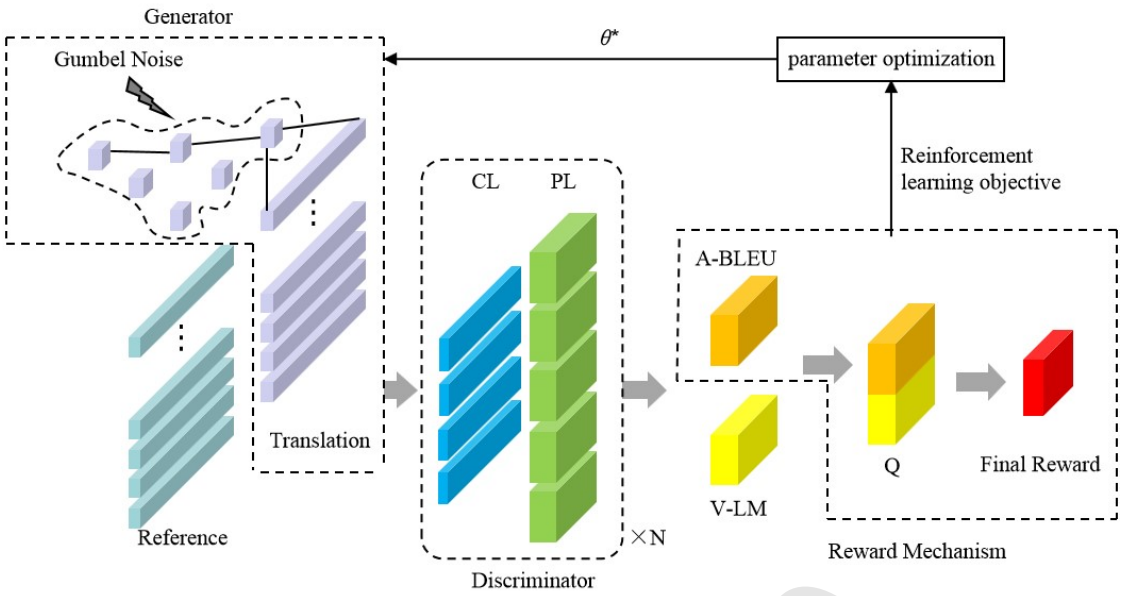


Figure 1: The Illustration of the proposed multi-reward generative adversarial net(referred to as MR-GAN). The discriminator D is trained over the reference sentence pairs translated by the human and the generated sentence pairs (sampled with Gumbel noise) by G. Extract feature information via convolution (CL) and pooling (PL) operations, and adopt global features to jointly calculate rewards. Lastly, the generator G is trained by policy gradient where the final reward R is provided by D, V and A.

3 Approach

In this section, we describe the multi-reward of reinforcement learning evaluation paradigm based on GAN model. The overall architecture is shown in Figure 1. We introduce the generator G, discriminator D, sampling with Gumbel-Top-K Stochastic Beam Search, calculating final reward and training the entire model in detail.

3.1 GAN-based NMT

The Generative Adversarial Net comprises of two adversarial sub models, a generator and a discriminator. The generator G is similar to the NMT model. Based on the source language sentence X, G aims to generate a target sentence $\hat{Y}$ which is indistinguishable from the reference Y. We take two different architectures for the generator, the traditional RNNSearch (Bahdanau et al., 2015) and the state-of-the-art Transformer (Vaswani et al., 2017).

We utilize CNN (Yin et al., 2016) that performs better in classification tasks to construct the discriminator D. It aims to identify machine-generated sentences from a set of sentences containing machine translation $\hat{Y}$ and reference Y. To be specific, the generator's output $\hat{Y}$ or reference Y is spliced with the source language sentence X to form a two-dimensional matrix, and the similarity between Y and X is measured by a convolution network. The optimization goal of the discriminator is minimize the coss-entropy loss of the binary classification:

$$L = -\left[a * log(p) + (1-a) * log(1-p)\right], p = \delta\left(V\left[r_X; r_{\hat{Y}}\right]\right) \tag{4}$$

where p is the probability that the target-language sentence is being real. r_X is the sentence representation of source language, which consists of extracting different features through different numbers of kernels with different window sizes. Similarly, $r_{\hat{Y}}$ is the target language sentence representation extracted from the target matrix $\hat{Y}_{1:T} = \hat{y}_1; \hat{y}_2; ...; \hat{y}_T$. V indicates the matrix which is used to merge r_X and $r_{\hat{Y}}$ into a low dimensional vector space. δ denote as the logistic function and a is a variable, which is correctly 1, otherwise 0.

3.2 Sampling

General NMT adopt Beam Search to generate the next token to reduce search space and speed up decoding. However, in many training methods such as RL or MRT, it is necessary to randomly collect multiple different samples from the model to calculate the sentence-level loss when decoding, but traditional methods can only produce similar results and loss of randomness. For this purpose, this work adopts an efficient and stable sampling method based on Gumbel-Top-K Stochastic Beam Search (Kool et al., 2019) to predict next token.

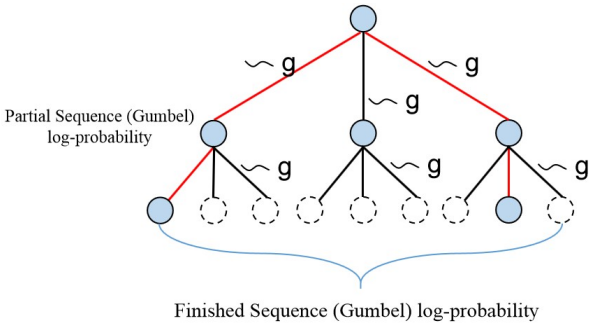


Figure 2: Gumbel-Top-K Stochastic Beam Search. $\sim g$ indicate add Gumbel noise when sampling.

This algorithm uses Top-Down sampling (Maddison et al., 2014) and performs Beam Search on the log probability of random perturbations. The structure is shown in Figure 2 with $k = 2$. We first perturb the log probability of the root node, then perturb and correct the log probability of all candidate sequences, and only keep the two nodes with the highest log perturbation probability to expand. Finally we get two samples with more randomness as well as each sample is subject to the original distribution.

Specifically, for a category distribution I with N categories $I \sim Categorical\left(\frac{exp\phi_i}{\sum_{j\in N} exp\phi_i}\right)$, where ϕ_i is the log-probability of the i-th category and $i \in N$. If we take the logarithm of each category of I and add the noise g that obeys the Gumbel distribution, then take the Top-K from this slightly disturbed sentence with the Top-K probability (ie.largest K categories after logarithmic calculation). The equations are as follows:

$$G \sim Gumbel\,(\phi) = \phi - log\,(-logU) = Gumbel\,(0) + \phi \tag{5}$$

$$I_{1,\ldots,k} = argtopK_{i\in N} Gumbel\,(\phi_i) \tag{6}$$

where $U \sim Uniform\,(0,1)$ and $G_i \sim Gumbel\,(0)$. It can be guaranteed that these K categories are subject to I and are different simultaneously, meanwhile the noise is controlled by the Gumbel distribution. With this method, we can train the model more efficiently and alleviate the problems of overtranslation and undertranslation in NMT.

3.3 Multi-Reward of Reinforcement Learning

As shown in Figure 1. Distinct with (Yang et al., 2018), which directly apply smoothed sentence-level BLEU as the specific objective Q for the generator. We aim to alleviate the overestimation of the reward, meanwhile, consider the fluency of machine translation and loyalty to the real translation. Therefore, our method is inspired by (Wang et al., 2016), given the generated sentence $\hat{Y}_{1:t}$ and the reference Y, the objective Q calculates a reward $Q(\hat{Y}_{1:t}, Y)$, which measures the fluency and loyalty of the generated sentence $\hat{Y}_{1:t}$, the equation is computed as:

$$Q(\hat{Y}_{1:t}, Y) = \lambda V\left(\hat{Y}_{1:t}\right) + (1-\lambda)\, A\left(\hat{Y}_{1:t}, Y\right) \tag{7}$$

where we set the independently generated language model reward as the value function $V\left(\hat{Y}_{1:t}\right)$, and the sentence reward as the advantage function $A\left(\hat{Y}_{1:t}, Y\right)$. λ is a hyper-parameter.

Value function For the sake of receiving smoother translation, we utilize the language model scores to participate in the calculation of rewards in reinforcement learning so that the NMT can consider the contextual and positional information of the corpus when translating. $V\left(\hat{Y}_{1:t}\right)$ represents the fluency score of sequence $\hat{y}_{1:t}$ including the current word, which guide the NMT to translate a sufficiently accurate and smooth result.

Typical language models have problems such as zero probability or statistical inadequacies. Good and Turing (Gale and Sampson, 1995) proposed a new probabilistic formula to ease the "unsmoothness" problem. As shown in Figure 3. They solved the zero-probability problem by down-regulating the frequency of words below the threshold and giving the out-of-vocabulary (OOV) a small non-zero value, where the sum of the down-regulated frequencies equals the probability of the OOV. The equation for 3-gram is as follow:

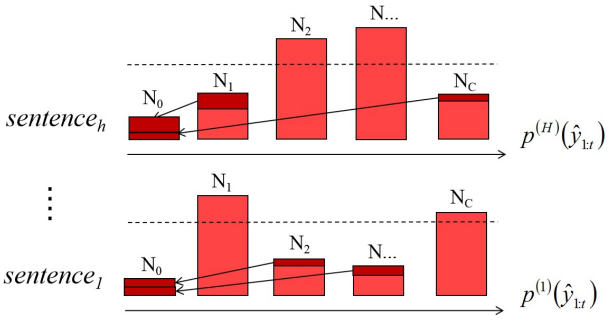


Figure 3: Good-Turing smoothing algorithm. "...." represent threshold, decrease the frequency of words which number of occurrences is lower than the threshold, and give the sum of the resulting frequencies to the words that do not appear.

$$P_{GT}(\omega_i|\omega_{i-2},\omega_{i-1}) = \begin{cases} f(\omega_i|\omega_{i-2},\omega_{i-1}), c(\omega_{i-2},\omega_{i-1},\omega_i) \geq U \\ f_{gt}(\omega_i|\omega_{i-2},\omega_{i-1}), 0 < c(\omega_{i-2},\omega_{i-1},\omega_i) < U \\ Q(\omega_{i-2},\omega_{i-1}) \cdot P(\omega_i|\omega_{i-1}), otherwise \end{cases} \tag{8}$$

where we set $U = 9$, which is a threshold, and the function $f_{gt}(.)$ represents the relative frequency after Good-Turing estimation. Therefore, the probability is normalized to get $V\left(\hat{Y}_{1:t}\right)$, the equation is as follow:

$$V\left(\hat{Y}_{1:t}\right) = \sum_{s=1}^{S} c_s P_{GT}^s \tag{9}$$

where c_s represents the number of words that occur s times, and P_{GT}^s represents the probability of s occurrences obtained from Good-Turing smooth algorithm.

Advantage function From Equation(2), the reward $R\left(\hat{y}, y\right)$ is only obtained after generate a complete sentence $\hat{y}$, it indicate only one reward is available for all actions(sample $\hat{y}_1 ... \hat{y}_T$). Consequently, RL training is inefficient due to the sparsity of rewards, and the model updates each token in the training sentence with the same reward without distinction. Following (Wu et al., 2018a), we employ reward shaping to overcome the shortcoming. The current reward with reward shaping is defined as:

$$r_t\left(\hat{y}_t, y\right) = R\left(\hat{y}_{1...t}, y\right) - R\left(\hat{y}_{1...t-1}, y\right) \tag{10}$$

where $R\left(\hat{y}_{1...t}, y\right)$ is defined as the BLEU score of $\hat{y}_{1...t}$ respect to y. Reinforce algorithm has high variance because it use a single sample $\hat{y}$ to estimate the expectation. To improve the stability of the algorithm, we add an estimate of the average reward at each step t, and then subtract it from future cumulative reward. The cumulative reward are obtained from (11):

$$R\left(\hat{y}, y\right) = \sum_{i=1}^{m} r_t\left(\hat{y}, y\right), R\left(\hat{y}, y\right) - \hat{r}_t \tag{11}$$

Combined with reward shaping, at each step t the Advantage function is computed as:

$$A\left(\hat{Y}_{1:t}, Y\right) = \sum_{T=t}^{m} r_T\left(\hat{y}_T, y\right) - \hat{r}_t \tag{12}$$

Final Reward According to the objective of the generator model (policy) $G_{\theta^*}(\hat{y}_t|\hat{Y}_{1:t-1})$ (Yu et al., 2017), to estimate $R_{D,V,A}^{G_{\theta^*}}$,which is the action-value function of a target sentence. Following Equation

(6), we consider the actual estimated probability of the discriminator D, the language model scores V and the sentence reward A as the final reward that update and optimize the generator G:

$$R_{D,V,A}^{G_{\theta^*}}\left(\hat{Y}_{1:T-1}, X, \hat{y}_T, Y\right) = \alpha\left(D\left(X, \hat{Y}_{1:T}\right) - b\left(X, \hat{Y}_{1:T}\right)\right) + \beta V\left(\hat{Y}_{1:t}\right) + \gamma A\left(\hat{Y}_{1:t}, Y\right) \quad (13)$$

where $b\left(X, \hat{Y}\right)$ represents the baseline value for reducing the variance estimation of rewards. We set $b\left(X, \hat{Y}\right) = 0.5$ based on experience. $\hat{Y}_{1:T}$ represents the generated target sentence and Y indicates the reference. α,β,γ are hyper-parameters.

However, D only provides a reward value for a entire generated target sequence. If $\hat{Y}_{1:T}$ is not the completed target sequence, the value of $D\left(X, \hat{Y}_{1:T}\right)$ is meaningless. Therefore, we cannot obtain the action-value of the intermediate state directly. Due to the large variance and parameters of Monte Carlo search, our work utilize Temporal-Difference (TD)[0] to sample the last $T-t$ tokens, it does not stop until the end of the sentence is sampled or the sampled sentence attains the maximum length. We implement the H TD emulation process as:

$$(\hat{Y}_{1:T_1}^1, ..., \hat{Y}_{1:T_H}^H) = TD^{G_{\theta^*}}\left(\left(\hat{Y}_{1:t}, X\right), H\right) \quad (14)$$

where $\left(\hat{Y}_{1:t}, X\right) = (\hat{y}_1 ... \hat{y}_t, X)$ is the current state, and $\hat{Y}_{t+1:T_H}^H$ is sampling based on G_{θ^*}. The discriminator rewards the sampled sentences separately and the discriminator output is calculated as the average of the H rewards. Therefore, for a target sentence of length T, we calculate the reward for $\hat{y}_t$ as:

$$R_{D,V,A}^{G_{\theta^*}}\left(\hat{Y}_{1:t-1}, X, \hat{y}_T, Y\right) = \begin{cases} \frac{1}{H}\sum_{j=1}^{H} \alpha\left(D\left(X, \hat{Y}_{1:T_h}^h\right) - b\left(X, \hat{Y}_{1:T_h}^h\right)\right) + \beta V\left(\hat{Y}_{1:T_h}\right) + \gamma A\left(\hat{Y}_{1:T_h}, Y\right) & t < T \\ \alpha D((X, \hat{Y}_{1:t}) - b(X, \hat{Y}_{1:t})) + \beta V\left(\hat{Y}_{1:t}\right) + \gamma A\left(\hat{Y}_{1:t}, Y\right) & t = T \end{cases} \quad (15)$$

3.4 Training

The training goal is to train G from the initial state to achieve maximum expectations end rewards. The objective equation is as follows:

$$J(\theta^*) = \sum_{\hat{Y}_{1:T}} G_{\theta^*}\left(\hat{Y}_{1:T}|X\right) \cdot R_{D,V,A}^{G_{\theta^*}}\left(\hat{Y}_{1:T-1}, X, \hat{y}_T, Y\right) \quad (16)$$

where R is Equation (16). Using sentence overall rewards to dynamically update the discriminator and then the generator.

$$min - E_{X,\hat{Y}\in P_{data}}\left[logD\left(X, \hat{Y}\right)\right] - E_{X,\hat{Y}\in G}\left[log\left(1 - D\left(X, \hat{Y}\right)\right)\right] \quad (17)$$

After completing the above operations, we adopt gradient descent to retrain the generator:

$$\bigtriangledown J(\theta^*) = \frac{1}{T}\sum_{t=1}^{T} E_{\hat{y}_t\in G_{\theta^*}}\left[R_{D,V,A}^{G_{\theta^*}}\left(\hat{Y}_{1:t-1}, X, \hat{y}_T, Y\right) \cdot \bigtriangledown_{\theta^*} logp\left(\hat{y}_t|\hat{Y}_{1:t-1}, X\right)\right] \quad (18)$$

4 Experiment and Analysis

We evaluate Chinese-English (Zh-En), English-German (En-De) and Mongolian-Chinese(Mo-Zh) tasks to verify the effectiveness of our MR-GAN.

[0]Monte Carlo search is updated after sampled the complete sentence $\hat{y}$. It causes too many parameters and slower update speed when sentence length is longer. Temporal-Difference (TD) algorithm is an iterative way of calculating value function, which is updated once per sampling, accelerates the convergence speed and reduces variance.

Model	Zh-En				En-De
	MT14	MT15	MT16	AVE	Newstest2014
Representative end-to-end NMT systems					
RNNSearch (Bahdanau et al., 2015)	33.76	34.08	33.98	33.94	21.20
RNNSearch+BR-CSGAN (Yang et al., 2018)	35.47	35.71	36.14	35.77	22.89
Transformer (Vaswani et al., 2017)	41.82	41.67	41.92	41.80	27.30
Transformer+RL (Wu et al., 2018a)	41.96	42.13	41.97	42.02	27.25
Transformer+BR-CSGAN (Yang et al., 2018)	42.46	42.54	42.83	42.61	27.92
Our work					
RNNSearch+MR-GAN	36.93	37.04	36.89	**36.95**	**24.61**
Transformer+MR-GAN	43.23	43.66	43.98	**43.62**	**28.69**

Table 1: BLEU scores of different NMT systems on Chinese-English and English-German.

4.1 Datasets and preprocessing

For En-De translation, we conduct our experiments on WMT14 En-De dataset, which contains 4.5 million bilingual pairs. Sentences are encoded using byte-pair encoding(BPE) (Sennrich et al., 2016). Newstest2012/2013 are chosen for development set, Newstest2014 as the test set. For the Zh-En translation, LDC2014 corpus as training set with a total of 1.6 million bilingual pairs. Both the source and target sentences are encoded with BPE. MT2013 is used as a development set and MT2014/2015/2016 as a test set. For Mo-Zh translation, the dataset adopts 261643 sentence pair Mongolian-Chinese bilingual aligned corpus provided by CWMT2018, we utilize 220000 sentence pairs as training set, 20822 as validation set, and the rest as test set. We perform word segmentation processing on the Chinese. On the Mongolian end, due to its own natural separator, so we encode it with BPE.

4.2 Setting

For Transformer-Big, following (Vaswani et al., 2017), we set $dropout = 0.1$ and set the dimension of the word embedding as 1024. We employ the Gumbel-Top-K Stochastic Beam Search to sample the target token with beam size $K = 4$. A single model obtained by averaging the last 20 checkpoints and we use adaptive methods to adjust the learning rate. For RNNSearch (Bahdanau et al., 2015), it is an RNN-based encoder decoder framework with attention mechanism. We set the hidden layer nodes and word embedding dimensions of the encoder and decoder to 512 and $dropout = 0$. The learning rate and checkpoint settings are consistent with Transformer-Big.

For D, CNN consists of one input layer, three convolution + pool layer pairs, one MLP layer and softmax layer. When the model is down-sampling, we use a 3×3 convolution window to perform convolution calculations on the internal corpus, and the output size is 2×2 pooling window. In addition, we set the feature map and MLP hidden layer size as 20. The word embedding dimension and the number of nodes are consistent with G.

Considering the computational complexity of model and the hardware environment of experiment, we adopt ELMO[1] (Peters et al., 2018) to construct and train the language model, which fully consider contextual information in semantic learning. Furthermore, we adopt BLEU (Papineni et al., 2002) to evaluate these tasks. All models are implemented in T2T tool and trained on two Titan XP GPUs. We stop training when the model does not improve on the tenth evaluation of the development set.

4.3 The pre-training of model

When the generator and discriminator achieve the synchronization and coordination effect, the performance of the model will be optimal. Therefore, we need to pre-train the model. The first step is pre-train

[1] http://allennlp.org/elmo/
ELMO, which fully consider contextual information has shown certain potential in semantic learning. It has strong modeling capabilities, meanwhile, the parameters and complexity are relatively small, which is convenient for model construction and training.

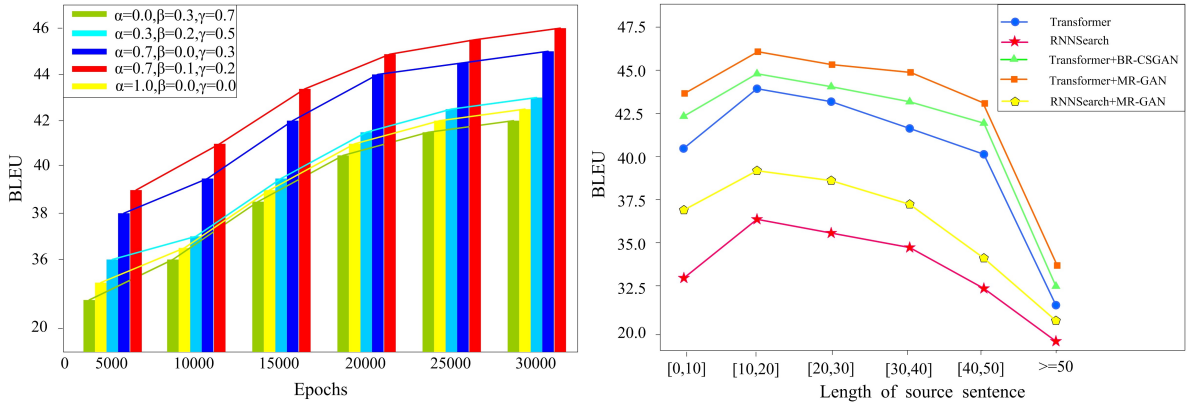


Figure 4: (a):Training line charts with different hyper-parameters weights. (b):BLEU scores on test set of LDC2014 Zh→En over different length of source sentences.

the generator G on bilingual training set until the best translation performance is achieved and we employ the traditional maximum likelihood estimation during the process. Then, generate the sentences(machine translations) by using the generator to decode the training data. The next step is pre-trian the discriminator on the combination of true bilingual data and machine transaltion data until the classification accuracy achieves at ξ. Finally, according to the study of Yang et al. (Yang et al., 2018), the method of jointly training the generator and discriminator and using the policy gradient to train the generator will lead to unstableness. Therefore, following (Yang et al., 2018), we adopt the teacher forcing approach to solve this problem. The parameter setting is exactly similar to (Yang et al., 2018), but the difference is that we employ the Temporal-Difference instead of Monte Carlo.

4.4 Main results and Analysis

For RNNSearch, it is optimized with the mini-batch of 64 examples. For Transformer, each training batch contains a set of sentence pairs contains approximately 25000 source tokens and 25000 target tokens. Table 1 shows the comparison between existing NMT system and our work. It can be seen that on Transformer, our approach outperforms the beat performance model and achieves improvement up to +1.01 BLEU points averagely on Chinese-English test sets an +0.77 BLEU points on English-German test set. It is profit from the novel method we have adopted to calculate rewards. Compared with traditional reinforcement learning, the scope of reward calculation is wider and making translation results more accurate and fluent. Furthermore, our method adds Gumbel noise when sampling, which makes the sampling more random and alleviates the problem of overtranslation and undertranslation. Experiments on the RNNSearch model shows the same trends, our approach still achieves 36.95 and 24.61 BLEU points on Chinese-English and English-German translations respectively.

4.5 Effect of Hyper-parameters and sentence length

We conduct a set of typical experiments using Transformer on the Chinese-English task to verify the influence of hyper-parameters (Equation 15) on experimental results. As shown in Figure $4(a)$, the worst result is obtained when $\alpha = 0$. The effect of the model continues to improve as the value of α increases. In the case of $\alpha = 0.7$, $\beta = 0.1$. and $\gamma = 0.2$, it achieves the best performance in several groups of experiments, and when $\alpha = 0.7$, $\beta = 0$. and $\gamma = 0.3$, the effect is not satisfactory. It indicates that the integration of language model rewards to evaluate the fluency of translation can effectively improve the quality of the model. When $\alpha = 1.0$, the effect is very poor but better than $\alpha = 0$, which explains that the multiple rewards proposed in this paper are effectively.

To verify the performance of this method on long sentences, following (Bahdanau et al., 2015), we divided the development set data and test set data of the Chinese-English task according to the sentence length. Figure $4(b)$ shows the BLEU scores for different sentence lengths. No matter on RNNSearch

ID	V-LM	A-BLEU	G-N	ZH-EN	EN-DE
1	×	×	×	42.61	27.92
2	×	√	×	42.82	27.97
3	√	√	×	43.07	28.23
4	√	×	√	43.21	28.42
5	√	√	√	**43.62**	**28.69**

(a)

Model	MO-ZH	Promote
Transformer	34.56	—
Transformer+RL	35.02	0.46
Transformer+BR-CSGAN	35.63	1.07
Transformer+Our method	**36.82**	**2.26**

(b)

Figure 5: (a): Ablation study on Zh→En and En→De tasks. "○" means utilize this module and "×" means not utilize. "G-N" indicate sample with Gumbel noise and Line 1 represent the result of BR-CSGAN. (b): BLEU scores on test set of CWMT2018 MO→ZH over different length of source sentences.

or Transformer, compared with baseline and the best performing BR-CSGANS (Yang et al., 2018), our work have outstanding behaviors continuously. It is due to our method not only calculates the single-step reward, but also adds a smoothing restriction, which makes our method perform better on both long and short sentences.

4.6 Ablation Study

Figure $5(a)$ shows the results of ablation study. Line 1 represent the result of BR-CSGAN and line 2 represent that reward shaping is used to calculate BLEU on the basis of BR-CSGAN. It is clear that language model reward plays a critical role since removing it impairs translation performance (line 3). As shown in line 4, sampling with Gumbel noise is also an essential part of our approach. The sentence reward with each token is also shown to be benefit for improving performance (line 2) but seem to have relatively smaller contributions than the above two parts.

4.7 Result of Mongolian-Chinese

To verify the robustness of the proposed method, we conducted a low-resource language Mongolian-Chinese experiment on Transformer. The experimental results are shown in Figure $5(b)$. Compared with the traditional Transformer, our approach improves 2.26 BLEU scores, meanwhile, it also increases 1.09 on the current best performance BR-CSGAN. It is fully proved that our method is also helpful for low-resource translation tasks.

5 Conclusion

In this paper, we propose a novel multi-reward reinforcement learning training paradigm to guide the optimization of model parameters, which makes the reward calculation more extensive. In addition, we employ Gumbel method instead of traditional beam search to selectively sample more random datas in the target space, and combining TD to calculate real-time reward. We validate the effectiveness of our method on the RNNSearch and the Transformer. A large number of experiments clearly show that our approach achieves significant improvements.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron C. Courville, and Yoshua Bengio. 2017. An actor-critic algorithm for sequence prediction. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

William A. Gale and Geoffrey Sampson. 1995. Good-turing frequency estimation without tears. *J. Quant. Linguistics*, 2(3):217–237.

Carl M. Harris and Jay Mandelbaum. 1985. A note on convergence requirements for nonlinear maximum-likelihood estimation of parameters from mixture models. *Computers & OR*, 12(2):237–240.

Wouter Kool, Herke van Hoof, and Max Welling. 2019. Stochastic beams and where to find them: The gumbel-top-k trick for sampling sequences without replacement. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 3499–3508.

Chris J. Maddison, Daniel Tarlow, and Tom Minka. 2014. A* sampling. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3086–3094.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Richard S. Sutton. 1988. Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

Ziyu Wang, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, and Nando de Freitas. 2016. Dueling network architectures for deep reinforcement learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, pages 1995–2003.

Ronald J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256.

Lijun Wu, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2018a. A study of reinforcement learning for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3612–3621.

Lijun Wu, Yingce Xia, Fei Tian, Li Zhao, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2018b. Adversarial neural machine translation. In *Proceedings of The 10th Asian Conference on Machine Learning, ACML 2018, Beijing, China, November 14-16, 2018*, pages 534–549.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2018. Improving neural machine translation with conditional sequence generative adversarial nets. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1346–1355.

Wenpeng Yin, Hinrich Schütze, Bing Xiang, and Bowen Zhou. 2016. ABCNN: attention-based convolutional neural network for modeling sentence pairs. *TACL*, 4:259–272.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 2852–2858.

Low-Resource Text Classification via Cross-lingual Language Model Fine-tuning

Xiuhong Li
Xinjiang University
xjulxh@xju.edu.cn

Zhe Li
Xinjiang University
lizhe@stu.xju.edu.cn

Jiabao Sheng
Xinjiang University
jiabao@stu.xju.edu.cn

Wushour Slamu
Xinjiang University
wushour@xju.edu.cn

Abstract

Text classification tends to be difficult when data are inadequate considering the amount of manually labeled text corpora. For low-resource agglutinative languages including Uyghur, Kazakh, and Kyrgyz (UKK languages), in which words are manufactured via stems concatenated with several suffixes and stems are used as the representation of text content, this feature allows infinite derivatives vocabulary that leads to high uncertainty of writing forms and huge redundant features. There are major challenges of low-resource agglutinative text classification the lack of labeled data in a target domain and morphologic diversity of derivations in language structures. It is an effective solution which fine-tuning a pre-trained language model to provide meaningful and favorable-to-use feature extractors for downstream text classification tasks. To this end, we propose a low-resource agglutinative language model fine-tuning $AggltiFiT$, specifically, we build a low-noise fine-tuning dataset by morphological analysis and stem extraction, then fine-tune the cross-lingual pre-training model on this dataset. Moreover, we propose an attention-based fine-tuning strategy that better selects relevant semantic and syntactic information from the pre-trained language model and uses those features on downstream text classification tasks. We evaluate our methods on nine Uyghur, Kazakh, and Kyrgyz classification datasets, where they have significantly better performance compared with several strong baselines.

1 Introduction

Text classification is the backbone of most natural language processing tasks such as sentiment analysis, classification of news topics, and intent recognition. Although deep learning models have reached the most advanced level on many Natural Language Processing(NLP) tasks, these models are trained from scratch, which makes them require larger datasets. Still, many low-resource languages lack rich annotated resources that support various tasks in text classification. For UKK languages, words are derived from stem affixes, so there is a huge vocabulary. Stems represent of text content and affixes provide semantic and grammatical functions. Diversity of morphological structure leads to transcribe speech as they pronounce while writing and suffer from high uncertainty of writing forms on these languages which causes the personalized spelling of words especially less frequent words and terms Ablimit et al. (2017). Data collected from the Internet are noisy and uncertain in terms of coding and spelling Ablimit et al. (2016). The main problems in NLP tasks for UKK languages are uncertainty in terms of spelling and coding and annotated datasets inadequate poses a big challenge for classifying short and noisy text data.

Data augmentation can effectively solve the problem of insufficient marker corpus in low-resource language datasets. Şahin and Steedman (2019) present two simple text augmentation techniques using "crops" sentences by removing dependency links, and "rotates" sentences by moving the tree fragments around the root. However, this may not be sufficient for several other tasks such as cross-language text classification due to irregularities across UKK languages in these kinds of scenarios. Pre-trained language models such as $BERT$ Devlin et al. (2018) or XLM Conneau and Lample (2019) have become

Proceedings of the 19th China National Conference on Computational Linguistics, pages 994-1005,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

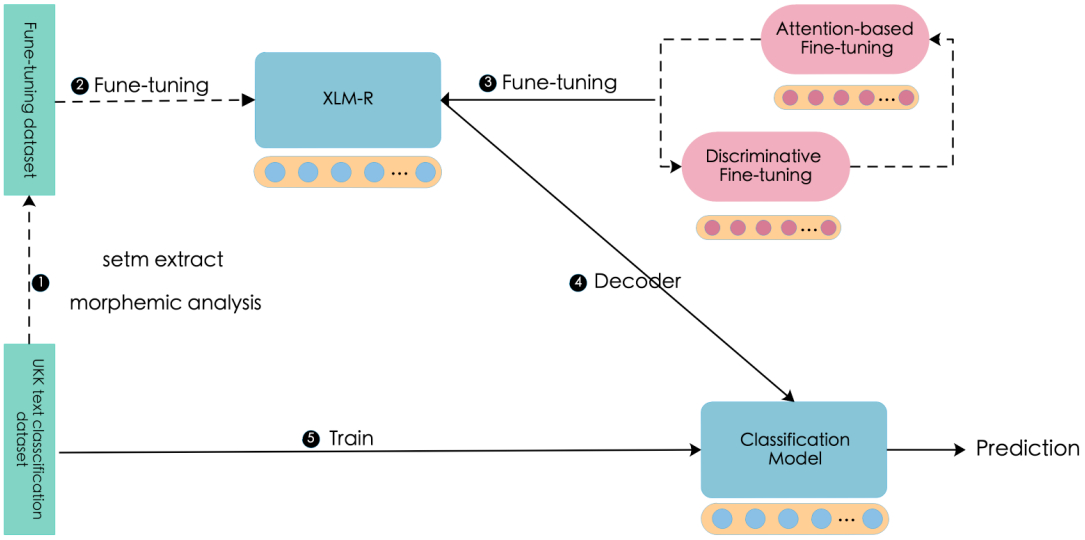


Figure 1: High-level illustration of AgglutiFiT

an effective way in NLP and yields state-of-the-art results on many downstream tasks. These models require only unmarked data for training, so they are especially useful when there is very little market data. Fully exploring fine-tuning can go a long way toward solving this problem Xu et al. (2020). Sun et al. (2019) conduct an empirical study on fine-tuning, although these methods achieve better performance, they did not perform well on UKK low-resource agglutinative languages due to the morphologic diversity of derivations.

The significant challenge of using language model fine-tuning on low-resource agglutinative languages is how to capture feature information. To apprehend rich semantic patterns from plain text, Zhang et al. (2019a) incorporating knowledge graphs (KGs), which provide rich structured knowledge facts for better language understanding. Zhang et al. (2019b) propose to incorporate explicit contextual semantics from pre-trained semantic role labeling (SemBERT) which can provide rich semantics for language representation to promote natural language understanding. UKK languages are a kind of morphologically rich agglutinative languages, in which words are formed by a root (stem) followed by suffixes. These methods are difficult to capture the semantic information of UKK languages. As the stems are the notionally independent word particles with a practical meaning, and affixes provide grammatical functions in UKK languages, morpheme segmentation can enable us to separate stems and remove syntactic suffixes as stop words, and reduce noise and capture rich feature in UKK languages texts in the classification task.

In this paper, as depict in Figure-1, we propose a low-resource agglutinative language model fine-tuning model: $AggltiFiT$ that is capable of addressing these issues. First, we use $XLM-R$ pre-train a language model on a large cross-lingual corpus. Then we build a fine-tuning dataset by stem extraction and morphological analysis as the target task dataset to fine-tune the cross-lingual pre-training model. Moreover, we introduce an attention-based fine-tuning strategy that selects relevant semantic and syntactic information from the pre-trained language model and uses discriminative fine-tuning to capture different types of information on different layers. To evaluate our model, we collect and annotate nine corpora for text classification of UKK low-resource agglutinative language, including topic classification, sentiment analysis, intention classification. The experimental results show $AggltiFiT$ can significantly improve the performance with a small number of labeled examples.

The contributions of this paper are summarized as follows:

- We collect three low-resource agglutinative languages including Uyghur, Kazakh, and Kyrgyz nine datasets, each of languages datasets contains topic classification, sentiment analysis, and intention classification three common text classification tasks.

- We propose a fine-tuning strategy on low-resource agglutinative language that builds a low-noise

fine-tuning dataset by stem extraction and morphological analysis to fine-tune the cross-lingual pre-training model.

- We propose an attention-based fine-tuning method that better select relevant semantic and syntactic information from the pre-trained language model and uses discriminative fine-tuning capture different types of information different layers.

2 Related work

In the field of natural language processing, low-resource text processing tasks receives increasing attention. We briefly reviewed three related directions: data augmentation, language model pre-training, and fine-tuning.

Data Augmentation Data Augmentation is that solves the problem of insufficient data by creating composite examples that are generated from but not identical to the original document. Wei and Zou (2019) present EDA, easy data augmentation techniques to improve the performance of text classification task. For a given sentence in the training set, EDA randomly chooses and performs one of the following operations: synonym replacement, random insertion, random swap, random deletion. UKK languages have few synonyms for a certain word, so the substitution of synonyms cannot add much data. Its words are formed by a root (stem) followed by suffixes, and as the powerful suffixes can reflect semantically and syntactically, random insertion, random swap, random deletion may change the meaning of a sentence and cause the original tags to become invalid. In the text classification, training documents are translated into another language by using an external system and then converted back to the original language to generate composite training examples, this technology known as $backtranslation$. Shleifer (2019) work experiments with $backtranslation$ as data augmentation strategies for text classification. The translation service quality of Uyghur is not good, and Kazakh and Kyrgyz do not have mature and robust translation service, so it is difficult to use the three languages in $backtranslation$. Şahin and Steedman (2019) propose an easily adaptable, multilingual text augmentation technique based on dependency trees. It augments the training sets of these low-resource languages which are known to have extensive morphological case-marking systems and relatively free word order including Uralic, Turkic, Slavic, and Baltic language families.

Cross-lingual Pre-trained Language Model Recently, Pre-training language models such as BERT Devlin et al. (2019) and GPT-2 Radford et al. (2019) have achieved enormous success in various tasks of natural language processing such as text classification, machine translation, question answering, summarization, etc. The early work in the field of cross-language understanding has proven the effectiveness of cross-language pre-trained models on cross-language understanding. The multilingual $BERT$ model is pre-trained on Wikipedia in 104 languages using a shared vocabulary of word blocks. LASER Artetxe and Schwenk (2019) is trained on parallel data of 93 languages and those languages share BPE vocabulary. Conneau and Lample (2019) also use parallel data to pre-train $BERT$. These models can achieve zero distance migration, but the effect is poor compared with the monolingual model. The $XLM-R$ Conneau et al. (2019) uses filtered common-crawled data over 2TB to demonstrate that using a large-scale multilingual pre-training model can significantly improve the performance of cross-language migration tasks.

Fine-tuning When we adapt the pre-training model to NLP tasks in a target domain, a proper fine-tuning strategy is desired. Howard and Ruder (2018) proposes the universal language model fine-tuning ($ULMFiT$) with several novel fine-tuning techniques. ULMFiT consists of three steps, namely general-domain LM pre-training, target task LM fine-tuning, and target task classifier fine-tuning. Eisenschlos et al. (2019) combines the $ULMFiT$ with the quasi-recurrent neural network ($QRNN$) Bradbury et al. (2018) and subword tokenization Kudo (2018) to propose multi-lingual language model fine-tuning ($MultiFit$) to enable practitioners to train and fine-tune language models efficiently. The $MultiFiT$ language model consists of one subword embedding layer, four $QRNN$ layers, one aggregation layer, and two linear layers. Moreover, a bootstrapping method Ruder and Plank (2018) is applied to reduce

the complexity of training. Although those approaches are general enough and have achieved state-of-the-art results on various classification datasets, the method is considered can not solve the problem of morphologic diversity of derivations in language structures on low-resource agglutinative language. Tao et al. (2019) proposes an attention-based fine-tuning algorithm. With this algorithm, the customers can use the given language model and fine-tune the target model by their own data, but that does not capture different levels of syntactic and semantic information on different layers of a neural network. In this paper, we use a new fine-tuning strategy that provides a feature extractor to extract features and use these features for downstream text classification tasks.

3 Methodology

In this section, we will explain our methodology, which is also shown in Figure-1. Our training consists of four stages. We first pre-train a language model on a large scale cross-lingual text corpus. Then the pre-trained model is fine-tuned by the fine-tuning dataset on unsupervised language modeling tasks. The fine-tuning dataset is constructed by means of stem extraction and morpheme analysis on the downstream classification datasets. Moreover, we use an attention-based fine-tuning to build our classification model and uses discriminative fine-tuning to capture different types of information on different layers. Finally, train the classifier using target task datasets.

3.1 LM fine-tuning based on UKK characteristics

When we apply the pre-training model to text classification tasks in a target domain, a proper fine-tuning strategy is desired. In this paper, we employ two fine-tuning methods as below.

3.1.1 Fine-tuning datasets based on morphemic analysis

UKK languages are agglutinative languages, meaning that words are formed by a stem augmented by an unlimited number of suffixes. The stem is an independent semantic unit while the suffixes are auxiliary functional units. Both stems and suffixes are called morphemes. Morphemes are the smallest functional units in agglutinative languages. Because of this agglutinative nature, the number of words of these languages can be almost infinite, and most of the words appear very rarely in the text corpus. Modeling based on a smaller unit like morpheme can provide stronger statistics hence robust models. The total number of suffixes in each of UKK languages is around 120. New suffixes may be created, but this is the typical case.

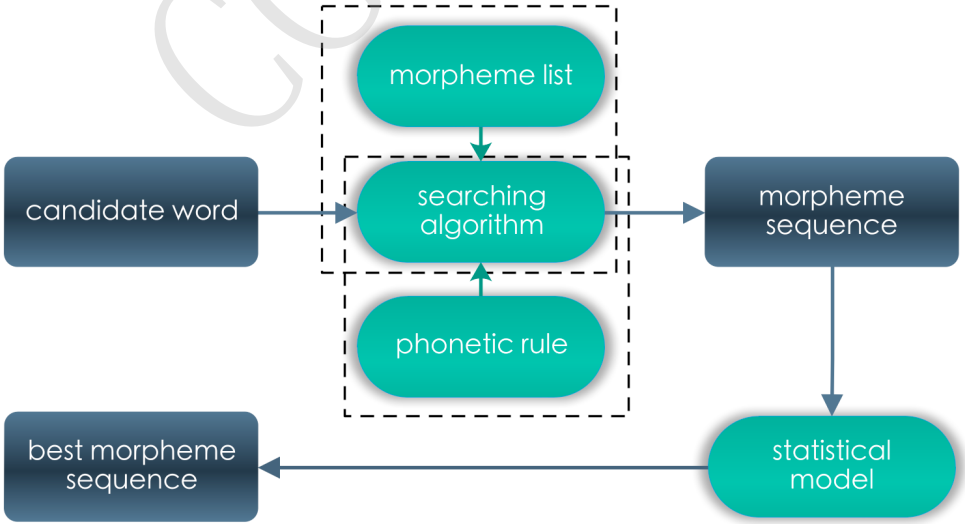


Figure 2: Morpheme segmentation flow chart

As shown in Figure-2, we use a semi-supervised morpheme segmenter based on the suffix set Ablimit et al. (2017). For a candidate word, this tool designs an iterative searching algorithm to produce all possible segmentation results by matching the stem-set and the suffix set. The phonemes on the boundaries

change their surface forms according to the phonetic harmony rules when the morphemes are merged into a word. Morphemes will harmonize each other, and appeal to the pronunciation of each other. When the pronunciation is precisely represented, the phonetic harmony can be clearly observed in the text. An independent statistical model can be adopted to pick the best result from N-best results in the UKK text classification task.

We adopt this tool to train a statistical model using word-morpheme parallel training corpus, extraction and greatly improved the UKK text classification task. which included 10,000 Uyghur sentences, 5000 Kazakhh sentences, and 5000 Kyrgyz sentences. We selected 80% of them as the training corpus. The remainder is used as the testing corpus to execute morpheme segmentation and stem extraction experiments. We can collect necessary terms compose a less noise fine-tuning datasets by extracting stems in the UKK languages classification task. Then fine-tuning with XLM-R on this fine-tuning datasets for better performance. For example in Table-1, a stem can grasp the features of other words, and the feature will be greatly reduced.

Stem	Words	Affixes
ئىش work	ئىش+چى = ئىشچى worker	چى
	ئىش+خانا = ئىشخانا office	خانا
	ئىش+ تات = ئىشتات position	تات
ئوقۇ read	ئوقۇ+ ش = ئوقۇش go to school	ش
	ئوقۇ+غۇچى = ئوقۇغۇچى student	غۇچى
	ئوقۇ+ ت = ئوقۇت teach	ت

Table 1: Examples of Uyghur word variants.

3.1.2 Discriminative Fine-tuning

Different layers of a neural network can capture different levels of syntactic and semantic information Yosinski et al. (2014; Howard and Ruder (2018). Naturally, the lower layers of the $XLM-R$ model may contain more general information. Therefore, we can fine-tune them with assorted learning rates. Following Howard and Ruder (2018), we use the discriminative fine-tuning method. We separate the parameters θ into $\{\theta^1, ..., \theta^L\}$, where θ^l contains the parameters of the l-th layer. Then the parameters are updated as follows:

$$\theta_t^l = \theta_{t-1}^l - \eta^l \cdot \nabla_{\theta^l} J(\theta), \tag{1}$$

where η^l represents the learning rate of the $l-th$ layer and t denotes the update step. Following Sun et al. (2019), we set the base learning rate to η_L and use $\eta^{k-1} = \xi \cdot \eta_k$, where ξ is a decay factor and less than or equal to 1. When $\xi < 1$, the lower layer has a slower learning rate than the higher layer. When ξ = 1, all layers have the same learning rate, which is equivalent to the regular stochastic gradient descent (SGD).

3.1.3 Attention-based Fine-tuning

For classification tasks, we adopt an attention-based encoder-decoder structure. As the encoder, our pre-trained model learns the contextualized features from inputs of the dataset. Then the hidden states over time steps, denoted as $H = h_1, h_2, ..., h_T$, can be viewed as the representation of the data to be classified, which are also the input of the attention layer. Since we do not have any additional information from the decoder, we use the self-attention to extract the relevant aspects from the input states. Specifically, the alignment is computed as

$$u_t = \tanh(W_u h_t + bu) \tag{2}$$

for $t = 1, 2, ..., T$, where W_u and b_u are the weight matrix and bias term to be learned. Then the alignment scores are given by the following soft-max function:

$$\alpha_t = \frac{\exp(W_\alpha u_t)}{\sum_{i=1}^{T} \exp(W_\alpha u_t)} \tag{3}$$

The final context vector, which is also the input of the classifier, is computed by

$$c = \sum_{i=1}^{T} \alpha_t u_t \tag{4}$$

3.2 Text Classifier

For the classifier, we add two linear blocks with batch normalization and dropout, and ReLU activations for the intermediate layer and a Softmax activation for the output layer that calculates a probability distribution over target classes. Consider the output of the last linear block is S_o. Further, denote by $C = c_1, c_2, ..., c_M = XxY$ the target classification data, where $c_i = (x_i, y_i)$, x_i is the input sequence of tokens and y_i is the corresponding label. The classification loss we use to train the model can be computed by:

$$L_2(C) = \sum_{(x,y)\in C} \log p(y|x) \tag{5}$$

where

$$p(y|x) = p(y|x_1, x_2, ..., x_m) := softmax(W_{s_o}) \tag{6}$$

4 Datasets

4.1 Data Collection

We construct nine low-resource agglutinative language datasets including Uyghur, Kazakh, and Kyrgyz, these datasets cover common text classification tasks: topic classification, sentiment analysis, and intention classification. We use the web crawler technology to collect our text data, and download from the Uyghur, Kazakh and Kyrgyz's official websites as well as other main websites.[1]

4.2 Corpus Statistics

In this section, we introduce the detailed information of the corpus. We divided them into morpheme sequences and used morpheme segmentation tools to extract word stems. The method of subword extraction based on stem affix has achieved a good performance on the reduction of feature space. As a result, the vocabulary of morpheme is greatly reduced to about 30%, as shown in Table 2, Table 3 and Table 4. In addition, when the types and numbers of corpora increase, the accumulation of morphemes is only one-third of the accumulation of words.

Topic Classification The corpus for the Uyghur language cover 9 topics: law, finance, sports, culture, health, tourism, education, science, and entertainment. Each category has 1,200 texts, resulting in a total of 10,800 texts. We name this corpus as `ug-topic`. The corpus for the Kazakh language cover 8 topics: law, finance, sports, culture, tourism, education, science, and entertainment. Each of them contains 1,200 texts, so there are 9,600 texts totally. We name this corpus as `kz-topic`. The corpus for the Kyrgyz language cover 7 topics: law, finance, sports, culture, tourism, education. Each category contains 1,200 texts (totally 8,400 texts). We name this corpus as `ky-topics`. The details are shown in Table-2.

Sentiment Analysis We constructed 3 sentiment analysis datasets for three-category classification, namely positive, negative, and neutral. Each language is related to 900 texts and each category contains 300 texts. We name these datasets as `ug-sen`, `kz-sen` and `ky-sen` as shown in Table-3.

[1] `www.uyghur.people.com.cn`, `uy.ts.cn`, `Kazakhh.ts.cn`, `www.hawar.cn`, Sina Weibo, Baidu Tieba and WeChat.

Intention Classification We construct 3 datasets of five-class user intent identification: news, life, travel, entertainment, and sports. Each language contains 200 texts. We name these datasets as `ug-intent`, `kz-intent` and `ky-intent` as shown in Table-4.

Corpus	of Class	Average text length	Word Vocabulary	Morpheme Vocabulary	Morpheme-Word Vocabulary Ratio (%)
ug-topic	9	148.3	79,126	23,364	29.5%
kz-topic	8	130.9	68,334	20,600	30.1%
ky-topic	7	145.7	58,137	18,487	31.7%

Table 2: Statistics of the topic classification dataset.

Corpus	of Class	Average text length	Word Vocabulary	Morpheme Vocabulary	Morpheme-Word Vocabulary Ratio (%)
ug-sen	3	23.6	8,791	2,794	31.1%
kz-sen	3	20.7	7,933	2,403	30.3%
ky-sen	3	21.3	7,385	2,274	30.8%

Table 3: Statistics of the sentiment analysis datasets.

Corpus	of Class	Average text length	Word Vocabulary	Morpheme Vocabulary	Morpheme-Word Vocabulary Ratio (%)
ug-intent	5	18.9	12,651	3,997	31.6%
kz-intent	5	16.0	10,368	3,182	30.7%
ky-intent	5	15.4	11,343	3,720	32.8%

Table 4: Statistics of the intention classification datasets.

4.3 Corpus Examples

In this section, we present some examples of various language categorization tasks. Different from Kazakhstan and Kyrgyzstan, in China, the Kazakh language used by the Kazakh people and the Kyrgyz language borrowed from the Arabic alphabet. The red keywords indicate the words that have the same meaning. The blue keywords represent their meaning in English.

5 Experiment

5.1 Datasets and Tasks

We evaluate our method on nine agglutinative language datasets which we construct of three common text classification tasks: topic classification, sentiment analysis, and intention classification. We use 75% of the data as the training set, 10% as the validation set, and 15% as the test set.

5.2 Baselines

We compare our method with the cross-lingual classification model $ULMFiT$ Howard and Ruder (2018), which introduces key techniques for fine-tuning language models, and $SemBERT$ Zhang et al. (2019b), which is capable of explicitly absorbing contextual semantics over a BERT backbone. Moreover, we compare against the cross-lingual embedding model, namely $LASER$ Artetxe and Schwenk (2019), which uses a large parallel corpus. We also compare against $BWEs$ Hangya et al. (2018), a cross-lingual domain adaptation method for classification text. For cross-lingual pre-training language models, the $XLM-R$ model used in this paper is loaded from the torch.Hub. $XLM-R$ shows the possibility of training one model for many languages while not sacrificing per-language performance. It is trained on $2.5TB$ of CommonCrawl data, in 100 languages and uses a large vocabulary size of

Topic	Law	Uyghur	دۆلەتنى قانۇن بويىچە ئىدارە قىلىشتا چىڭ تۇرۇش
		Kazakh	مەملەكەتتى زاڭمەن باسقارۋعا تاباندى بولۋ
		Kyrgyz	ماملەكەتتى زاكون بويۇنچا جونگو سالۇۇ
		English	Ensuring every dimension of governance is law-based
	Finance	Uyghur	ئامېرىكا ئىقتىسادىغا تەسىر كۆرسىتەمدۇ؟ COVID-19
		Kazakh	جاڭا ءتيپتى وكپە ايدارشا ۆيرۋسى امەريكا ەكونوميكاسىنا ىقپال ەتەمە؟
		Kyrgyz	جاڭىچا تاجىسامان ۋىرۇس امەرىكا ىقتىسادىناتاسىر كوتسوتوبۇ
		English	Will the COVID-19 pandemic affect the US economy?
	Sports	Uyghur	كوبى بىر ئۇلۇغ ۋاسكېتبول تەنھەرىكەتچىسى.
		Kazakh	كوبە ۇلى باسكەتبول سپورتشىسى
		Kyrgyz	گوبى دەگەن بئر ۇلۇۇ ۋاسكىتبول چەبەرى
		English	Kobe is a great basketball player.
Sentiment	Positive	Uyghur	شىنجاڭنىڭ مەنزىرىسى سۈرەتتەك گۈزەل
		Kazakh	شينجياڭنىڭ كورىنىسى سۋرەتتەي كوركەم
		Kyrgyz	شئنجاڭدىن كورۉنۉشتورۉ سۉروتتوي كوركوم
		English	Xinjiang is a picturesque landscape
	Neutral	Uyghur	بىز ئىلمىي ماقالە يېزىۋاتىمىز.
		Kazakh	ءبىز عىلمي ماقالا جازىپ جاتىرمىز
		Kyrgyz	بئز ماقالاجازىپ جاتابىز
		English	We are writing a paper
	Negative	Uyghur	سىز نېمىشقا بويسۇنمايسىز؟
		Kazakh	سەن نەگە بويسىنبايسىڭ؟
		Kyrgyz	سىز نەگە مويۇن سۇنبايسىز
		English	Why are you disobedient?

Table 5: Example from the UKK datasets

250K. For the $ULMFiT$ and $BWEs$ model, we use English as the source language. $XLM-R$ and $ULMFiT$ are fine-tuned on target task datasets rather than the fine-tuning datasets that we built.

5.3 Hyperparameters

In our experiment, we use the $XLM-R_{Base}$ model, which uses a $BERT_{Base}$ architecture Vaswani et al. (2017) with a hidden size of 768, 12 Transformer blocks and 12 self-attention heads. We fine-tune the $XLM-R_{Base}$ model on 4 Tesla K80 GPUs and set the batch size to 24 to ensure that the GPU memory is fully utilized. The dropout probability is always 0.1. We use Adam with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. Following Sun et al. (2019), we use the discriminative fine-tuning method Howard and Ruder (2018), where the base learning rate is $2e-5$, and the warm-up proportion is 0.1. We empirically set the max number of the epoch to 20 and save the best model on the validation set for testing.

5.4 Results and Analysis

In this section, we demonstrate the effectiveness of our low-resource agglutinative language fine-tuning model. Our approach significantly outperforms the previous work on cross-lingual classification. Separately, the best results in the metric are bold, respectively.

As given in Table-6, Table-7, and Table-8, We show results for topic classification, sentiment analysis, and intention classification. Our $AgglutiFiT$ outperform their cross-lingual and domain adaptation method. Pre-training is most beneficial for tasks with low-resource datasets and enables generalization even with 100 labeled examples when fine-tuning with fine-tuning dataset, our approach has a greater performance boost.

Compared with $ULMFiT$, we perform better on all three tasks, although $ULMFiT$ introduces techniques that are key for fine-tuning a language model including discriminative fine-tuning and target task classifier fine-tuning. The reason can be partly explained as we adopt a less noisy datasets in the fine-

Model	ug-topic	kz-topic	ky-topic
ULMFiT	92.99%	92.93%	92.34%
LASER	83.19%	82.32%	82.13%
SemBERT	91.53%	90.12%	90.24%
BWEs	59.24%	59.12%	58.89%
AggglutiFiT	**96.45%**	**95.39%**	**94.89%**

Table 6: Results on topic classification accuracy.

Model	ug-sen	kz-sen	ky-sen
ULMFiT	90.49%	90.39%	90.38%
LASER	74.32%	73.99%	72.13%
SemBERT	86.37%	88.47%	86.94%
BWEs	56.59%	56.39%	56.03%
AggglutiFiT	**92.81%**	**92.89%**	**92.23%**

Table 7: Results on sentiment analysis accuracy.

tuning phase and attention-based fine-tuning which makes it possible to obtain a closer distribution of data in the general domain to the target domain. $LASER$ obtain strong results in multilingual similarity search for low-resource languages, but we work better than $LASER$ contribute to we use attention-based fine-tuning and different learning rates at a different layer, which allows us to capture more syntactic and semantic information at each layer, moreover, $LASER$ has no learn joint multilingual sentence representations for UKK languages. Experimental results on methods $SemBERT$ are lower than $AggglutiFiT$ on account of lack of the necessary semantic role labels to embedding in the parallel lead to does not capture more accurate semantic information. $BWEs$ is significantly lower than other models, we conjecture is that the source language of method $BWEs$ is English, which is quite different from the UKK languages in data distribution, more importantly, the datasets of UKK languages are too inadequacy to create good $BWEs$. Our three task experiments also show that using more high-quality datasets to fine-tune the results would be better.

5.5 Ablation Study

To evaluate the contributions of key factors in our method, we perform an ablation study as shown in Figure-3. We run experiments on nine corpora that are representative of different tasks, genres, and sizes.

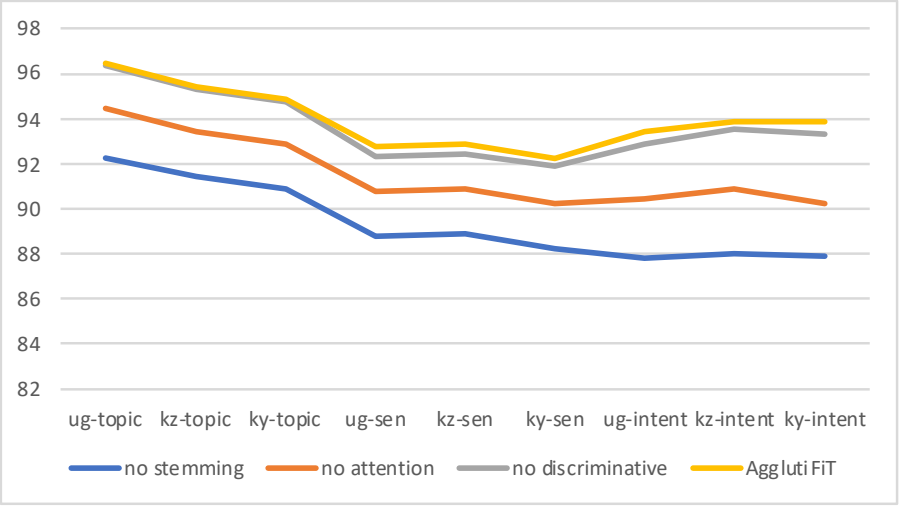


Figure 3: Explore the influence of important factors on accuracy

Model	ug-intent	kz-intent	ky-intent
ULMFiT	90.97%	91.23%	91.13%
LASER	77.21%	77.89%	77.33%
SemBERT	89.79%	87.28%	89.13%
BWEs	57.50%	57.48%	57.39%
AggglutiFiT	**93.47%**	**93.81%**	**93.28%**

Table 8: Results on intention classification accuracy.

The effect of morphemic Analysis In order to gauge the impact of fine-tuning datasets quality, we compare the fine-tuning on the constructed fine-tuning datasets with the target task datasets without stem-word extraction. The experimental results show that the performance of all tasks is greatly improved by using our fine-tuning datasets. Stem is a practical unit of vocabulary. Stem extraction enables us to capture effective and meaningful features and greatly reduce the repetition rate of features.

The effect of attention-based fine-tuning As given in Figure-3, we can observe that by adding an attention fine-tuning, our model advances accuracies. Attention-based fine-tuning relies on a semantic between words that would influence the overall model performance. In order to see the effectiveness of the attention-based fine-tuning more clearly, we visualize the attention scores with respect to the input texts on Uyghur. The randomly chosen examples of visualization with respect to different classes are given in Figure-4, where darker color means higher attention scores.

توكيو نولىمپىك تەشكىللەش كومىتېتىنىڭ رەئىسى :توكيو نولىمپىك تەنھەرىكەت مۇسابىقىسى يەلكىم بىۋاستە ئەمەلدىن قالدۇرۇلۇشى مۇمكىن

In English: Chairman of the Tokyo Olympic Organizing Committee: The Tokyo Olympics may be canceled directly.

(a) Sports

.شىنجاڭ گۈزەل جاي، بۇ يەرگە كېلىپ كەيپىياتىم خېلى ياخشى بولۇپ قالدى

In English: Xinjiang is a beautiful place and My mood feels very happy when I come here.

(b) Positive

.تۇرمۇش خۇددى نانغا نوخشايدۇ، سىز نېمىگە نېرىشىدىغانلىقىڭىزنى مەڭگۈ بىلمەيسىز

In English: Life is like bread, you never know what you will get.

(c) Life

Figure 4: Examples of attention visualization on Uyghur with respect to different classes

The effect of discriminative fine-tuning We compare with and without discriminative fine-tuning on the model. Discriminative fine-tuning improve performance across all three tasks, however, the role of improvement is limited, we still need a better optimization method to explore how discriminative fine-tuning can be better applied in the model.

6 Conclusion

We propose *AggglutiFiT*, an effective language model fine-tuning method that can be applied to a low-resource agglutinative language classification tasks. This novel fine-tuning technique that via stem extraction and morphological analysis builds a low-noise fine-tuning dataset as the target task dataset to fine-tune the cross-lingual pre-training model. Moreover, we propose an attention-based fine-tuning strategy that better selects relevant semantic and syntactic information from the pre-trained language model to provide meaningful and favorable-to-use feature for downstream text classification tasks. We also use discriminative fine-tuning to capture different types of information on different layers. Our method significantly outperformed existing strong baselines on nine low-resource agglutinative language datasets of three representative low-resource agglutinative text classification tasks. We hope that our results will catalyze new developments in low-resource agglutinative languages task for NLP.

7 Acknowledgments

This paper support by Xinjiang University Ph.D. Foundation Initiated Project Grant Number 620312343, National Language Commission Research Project Grant Number ZDI135-96.

References

Mijit Ablimit, Tatsuya Kawahara, Akbar Pattar, and Askar Hamdulla. 2016. Stem-affix based uyghur morphological analyzer. *International Journal of Future Generation Communication and Networking*, 9(2):59–72.

Mijit Ablimit, Sardar Parhat, Askar Hamdulla, and Thomas Fang Zheng. 2017. A multilingual language processing tool for uyghur, kazak and kirghiz. In *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 737–740. IEEE.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

James Bradbury, Stephen Joseph Merity, Caiming Xiong, and Richard Socher. 2018. Quasi-recurrent neural network, May 10. US Patent App. 15/420,710.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7057–7067.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 0(0).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL 2019*, pages 4171–4186.

Julian Eisenschlos, Sebastian Ruder, Piotr Czapla, Marcin Kadras, Sylvain Gugger, and Jeremy Howard. 2019. Multifit: Efficient multi-lingual language model fine-tuning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5706–5711.

Viktor Hangya, Fabienne Braune, Alexander Fraser, and Hinrich Schütze. 2018. Two methods for domain adaptation of bilingual tasks: Delightfully simple and broadly applicable. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 810–820. Association for Computational Linguistics.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *arXiv:1801.06146*.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75.

Alec Radford, Jeffrey Wu, Dario Amodei, Daniela Amodei, Jack Clark, Miles Brundage, and Ilya Sutskever. 2019. Better language models and their implications. *OpenAI Blog https://openai. com/blog/better-language-models*.

Sebastian Ruder and Barbara Plank. 2018. Strong baselines for neural semi-supervised learning under domain shift. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1044–1054.

Gözde Gül Şahin and Mark Steedman. 2019. Data augmentation via dependency tree morphing for low-resource languages. *arXiv preprint arXiv:1903.09460*.

Sam Shleifer. 2019. Low resource text classification with ulmfit and backtranslation. *arXiv preprint arXiv:1903.09244*.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *China National Conference on Chinese Computational Linguistics*, pages 194–206. Springer.

Yunzhe Tao, Saurabh Gupta, Satyapriya Krishna, Xiong Zhou, Orchid Majumder, and Vineet Khare. 2019. Finetext: Text classification via attention-based language model fine-tuning. *arXiv preprint arXiv:1910.11959*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Jason W Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Yige Xu, Xipeng Qiu, Ligao Zhou, and Xuanjing Huang. 2020. Improving bert fine-tuning via self-ensemble and self-distillation. *arXiv preprint arXiv:2002.10345*.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019a. Ernie: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2019b. Semantics-aware bert for language understanding. *arXiv preprint arXiv:1909.02209*.

Constructing Uyghur Named Entity Recognition System using Neural Machine Translation Tag Projection

Azmat Anwar, Xiao Li, Yating Yang, Rui Dong and **Turghun Osman**
Xinjiang Technical Institute of Physics & Chemistry, Chinese Academy of Sciences, Urumqi, China
University of Chinese Academy of Sciences, Beijing, China
Xinjiang Laboratory of Minority Speech and Language Information Processing, Urumqi, China
{azmat,xiaoli,yangyt,dongrui,turghun}@ms.xjb.ac.cn

Abstract

Although named entity recognition achieved great success by introducing the neural networks, it is challenging to apply these models to low resource languages including Uyghur while it depends on a large amount of annotated training data. Constructing a well-annotated named entity corpus manually is very time-consuming and labor-intensive. Most existing methods based on the parallel corpus combined with the word alignment tools. However, word alignment methods introduce alignment errors inevitably. In this paper, we address this problem by a named entity tag transfer method based on the common neural machine translation. The proposed method marks the entity boundaries in Chinese sentence and translates the sentences to Uyghur by neural machine translation system, hope that neural machine translation will align the source and target entity by the self-attention mechanism. The experimental results show that the Uyghur named entity recognition system trained by the constructed corpus achieve good performance on the test set, with 73.80% F1 score(3.79% improvement by baseline).

1 Introduction

Named Entity Recognition (NER) is a task of identifying named entities (NEs), especially person names (PER), location names (LOC), organization names (ORG), and classifying them into some pre-defined target entity classes (Hobbs et al., 1997). NER is essential to many natural language processing (NLP) tasks such as relation extraction (Christopoulou et al., 2019), event detection (Cakır and Virtanen, 2019), knowledge graph construction (Bosselut et al., 2019) and so on. Although the NER achieves great success by the introduction of the advanced neural networks (Collobert et al., 2011; Huang et al., 2015; Chiu and Nichols, 2016; Lample et al., 2016; Ma and Hovy, 2016; Yang et al., 2016; Peters et al., 2017; Liu et al., 2018; Peters et al., 2018), these methods are highly dependent on a large amount of annotated training data, and thus challenging to apply these models to low resource languages including Uyghur. Constructing a well-annotated NE corpus manually is very time-consuming and labor-intensive. Instead, Cross-lingual transfer is an effective solution, which addresses this challenge by transferring knowledge from a high-resource source language with abundant entity labels to a low-resource target language with few or no labels. According to the resource availability of the target language, different types of NER methods are proposed, such as bilingual parallel corpus based tag projection (Yarowsky et al., 2001; Ehrmann et al., 2011; Wang et al., 2013; Fang and Cohn, 2016; Ni et al., 2017), cross-lingual word embedding (Fang and Cohn, 2017; Wang et al., 2017; Huang et al., 2018), cross-lingual Wikification (Kim et al., 2012; Nothman et al., 2013; Tsai et al., 2016; Pan et al., 2017) or multi-task learning (Yang et al., 2016; Lin et al., 2018).

As a low resource language, Uyghur has no well-annotated corpus available for NER, but it is easy to get Uyghur-Chinese bilingual parallel corpus as Uyghur-Chinese machine translation is an important task of China Conference on Machine translation (CCMT). A common way of constructing NER corpus for the language which has a bilingual parallel corpus is using off-the-shelf NER tool in the source language to get entity annotations and transfer them to target language combing with the automatic

word alignment. Although some researchers have also applied this method to transfer NE annotations from Chinese to Uyghur and achieved remarkable results (Maimaiti et al., 2018), these pipeline methods inevitably introduce errors from the source language, including errors from NER tools and automatic word alignment. Figure 1 illustrates an Example of NER corpus construction based on the bilingual parallel corpus and automatic word alignment.

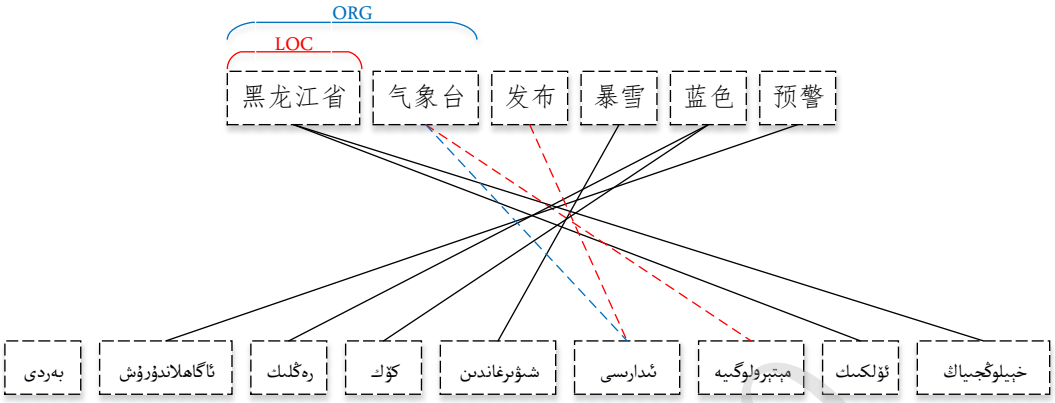


Figure 1: Example of NER corpus construction based on the bilingual parallel corpus and automatic word alignment. Errors from NER tools and automatic word alignments remarked in red color while blue indicates correct

In this paper, we address these challenges by a NE annotation transfer method based on neural machine translation (NMT). Given an Uyghur-Chinese parallel corpus, first, we train a general-purpose Chinese-Uyghur NMT system using the parallel corpus. Then, add the NE boundary information directly to the source Chinese sentence by multiple off-the-shelf NER tools. Finally, translate the Chinese sentences with entity boundary to Uyghur language using the pre-trained NMT system, we hope that NMT will align the source and target entity by the self-attention mechanism. Our method can be illustrated by the following example provided in Figure 2.

The main advantages of our method are used multi NER tools in the source language to minimize annotation errors and use general-purpose NMT without adding new tokens to indicate NE boundary in the parallel corpus, thus no need to annotate any training data manually.

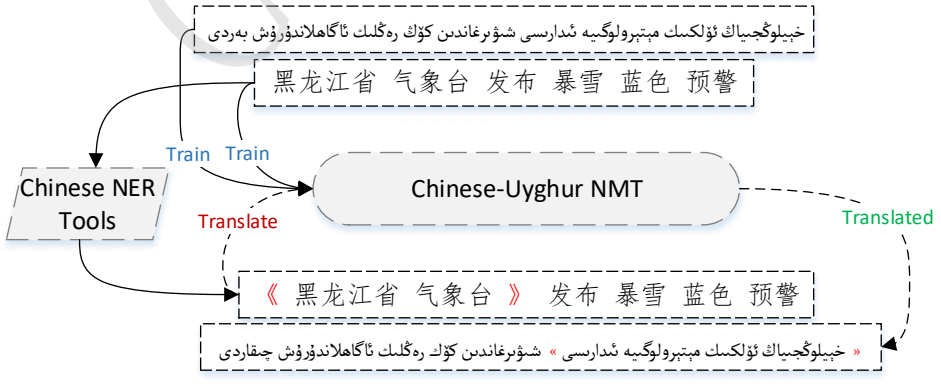


Figure 2: Example of transferring NE tags from Chinese to Uyghur using NER tools and NMT

2 Related Work

Named Entity Recognition: NER is typically framed as a task of sequence labeling which aims at automatic detection of NEs in free text(Marrero et al., 2013). CRF, SVM, and perceptron models with hand-crafted features are applied in early works (Ratinov and Roth, 2009; Passos et al., 2014; Luo et al.,

2015). With the great advantages of deep neural networks, research focuses on the neural network-based methods that need less feature engineering and domain knowledge(Lample et al., 2016; Žukov-Gregorič et al., 2018; Zhou et al., 2019). Collebert (2011) proposed a feed-forward neural network with a fixed-sized window for each word, which failed in considering useful relations between long-distance words. To overcome this limitation, Chiu et al. (2016) presented a bidirectional LSTM-CNNs architecture that automatically detects word and character-level features. Ma et al. (2016) further extended it into bidirectional LSTM-CNNs-CRF architecture, where the CRF module was added to optimize the output label sequence.

Transfer learning for NER: Low-resource languages often suffer from a lack of annotated corpora to estimate high-performing neural network models for many NLP tasks. Transfer learning is an efficient way to bridge the gap across languages. Transfer learning methods for NER can be divided into two types: parallel corpora based and shared representation based transfer. Early works mainly focus on parallel corpora to projecting information from high-resource languages to low-resource languages (Yarowsky et al., 2001; Ehrmann et al., 2011; Wang et al., 2013; Fang and Cohn, 2016; Ni et al., 2017). Chen et al. (2010) and Wang et al. (2013) proposed to jointly identify and align bilingual named entities. Kim el al. (2012), Nothman et al. (2013) and Tsai el al. (2016) using the Wikipedia information to improve low-resource NER. Mayhew et al. (2017) created a cross-language NER system by translating annotated data of high-resource to low-resource which works well for very minimal resource languages. On the other hand, the shared representation methods do not require parallel corpora. Fang et al. (2017) proposed cross-lingual word embeddings to transfer knowledge across resources. Pan et al. (2017) proposes a large-scale cross-lingual named entity dataset which contains 282 languages for evaluation. Yang el al. (2016), Wang et al. (2017), Lin et al. (2018) and Liu et al. (2018) shows that jointly training on multiple tasks or languages helps improve performance. Different from transfer learning methods, multi-task learning aims at improving the performance of all the resources instead of low resource only.

Token Added Machine Translation (TAMT): The researchers proposed TAMT methods to solve the different types of problems. Ugawa et al. (2018) add the entity tags to the source language sentences to disambiguate the multi-meaning entities in the target language. Li et al. (2018) use NE tags to indicate the NE boundary information in the source language sentences to get better customized entity translation. Bai et al. (2018) use some special tokens to mark the segmentation boundary for the slot value in the source sentence and transfer the source language spoken language understanding corpus to the target language.

3 Methodology

3.1 General-Purpose NMT System

Machine translation (MT) translates text sentences from a source language to a target language and the Transformer model is the first NMT model relying entirely on self-attention to compute representations of its input and output without using recurrent neural networks (RNN) or convolutional neural networks (CNN). Our general-purpose NMT system is based on the Transformer model.

The Transformer model is an encoder-decoder structure like most competitive neural sequence transduction models, as shown in Figure 3. The encoder is including three steps, in the first step, the input words are projected into an embedding vector space, position embedding is also added to input vectors to capture the notion of token position within the sequence. The second step is a multi-head self-attention. This is an extension of the previous attention scheme. Instead of using a single attention function, this step computes multiple attention blocks over the source, concatenates them and projects them linearly back onto a space with the original dimensionality. The scaled dot-product attention with different linear projections is computed over attention blocks individually. Finally, a position-wise fully connected feed-forward network is used, which consists of two linear transformations with a ReLU activation.

The decoder works similarity, from left to right with generates one word at a time. It including five steps. The first step: embedding and position encoding, is similar to the encoder. The second step is masked multi-head attention, which masks future words forces to attend only to past words. The third

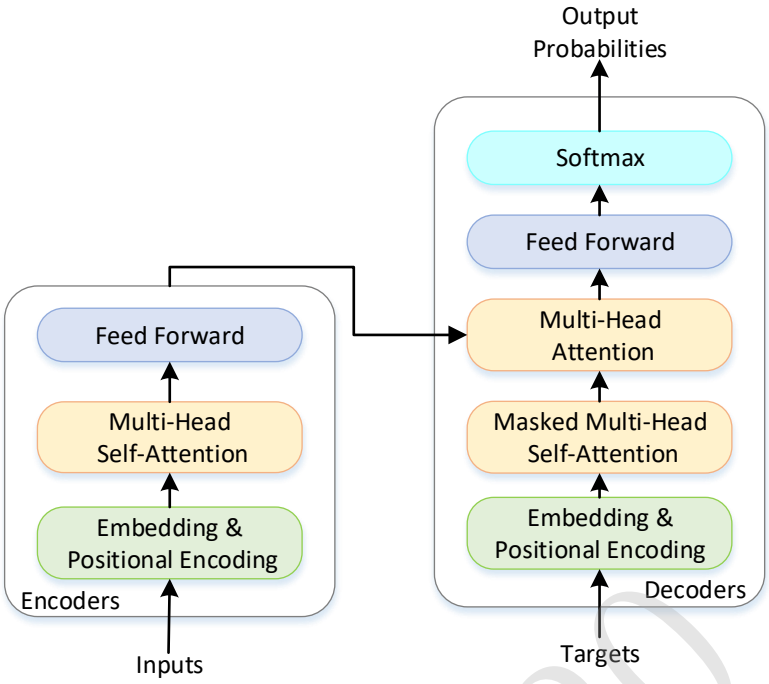


Figure 3: Simplified diagram of the Transformer model

step is a multi-head attention that not only attends to these past words, but also to the final representations generated by the encoder. The fourth step is another feed-forward network. Finally, a softmax layer applied to map target word scores into target word probabilities. More details about the model are found in the original paper (Vaswani et al., 2017).

3.2 Source language Named Entity Tags

We consider three NE classes in this paper (PER, LOC, ORG). For every NE in source sentence, we generate the candidate NE class tags using three types of third-party NER tools: Pyltp [0] from Harbin Institute of Technology, PaddleHub [1] from Baidu and THULAC [2] from Tsinghua University. In order to get the best tags from candidates, we will try two kinds of strategies described as follows:

Single tag combination (STC): Check these tools on a test set to get accuracy for each NE class, then use the highest accuracy tool to get the specific single class tag, such as PER from Pyltp, LOC from PaddleHub and so on.

Multi-tag combination (MTC): For single sentences, tags are comes from all three tools and combine them by following rules:(1) Tag kept for a single NE only if all of the three tags are identical. (2) Tag kept for the longer NE if NE from one tool includes another one. (3) Drop the sentences not satisfy any of the first two rules.

3.3 Token-added Translation

To make the general-purpose NMT aware of NEs, we propose a token added translation approach. This approach uses some special tokens to mark the segmentation boundary for the NE in the source sentence. These special tokens are common in both the source vocabulary and target vocabulary of the general-purpose NMT and their translation is unique and easy to spot. To avoid complexity, we use the same common special tokens for all NEs while keeping order and mark all NEs in the translated target sentences with the original order. For example, punctuation like parentheses and double quotes are good candidates as special tokens. Enclosing NEs in source sentences by these special tokens can help iden-

[0]https://github.com/HIT-SCIR/pyltp
[1]https://github.com/PaddlePaddle/PaddleHub
[2]https://github.com/thunlp/THULAC-Python

tify NE boundaries in the translation outputs. In our example in Figure 2, the special tokens we choose is a pair of Chinese punctuation named title mark (《》), which translated to corresponding Uyghur punctuation («»).

In token-added translation, no additional word alignment process is required. However, such an approach relies heavily on the NMT general training data where the special tokens (e.g. parentheses or double quotation marks) are kept in both source and target data. For different language pairs, different special tokens might be chosen for the best translation quality. Empirically we find that title marks are highly effective for Chinese to Uyghur NE translation.

3.4 NER Model

The hierarchical CRF model consists of three components: a character-level neural network, either an RNN or a CNN, that allows the model to capture subword information, such as morphological variations and capitalization patterns; a word-level neural network, usually an RNN, that consumes word representations and produces context-sensitive hidden representations for each word; and a linear-chain CRF layer that models the dependency between labels and performs inference.

In this paper, we closely follow the architecture proposed by Lample et al. (2016), and use bidirectional LSTMs for both the character level and word level neural networks. Specifically, given an input sequence of words $(w_1, w_2, ..., w_n)$, and each word's corresponding character sequence, the model first produces a representation for each word, x_i, by concatenating its character representation with its word embedding. Subsequently, the word representations of the input sequence $(x_1, x_2, ..., x_n)$ are fed into a word level BiLSTM, which models the contextual dependency within each sentence and outputs a sequence of context sensitive hidden representations $(h_1, h_2, ..., h_n)$. A CRF layer is then applied on top of the word level LSTM and takes in as its input the sequence of hidden representations $(h_1, h_2, ..., h_n)$, and defines the joint distribution of all possible output label sequences. The Viterbi algorithm is used during decoding.

4 Experiment

4.1 Data

The CCMT 2017 Chinese-Uyghur corpus [3] is used to train the general-purpose Chinese-Uyghur NMT system and the MSRA dataset from international Chinese language processing Bakeoff 2006 [4] is used to evaluate the performance of Chinese NER tools. As no publicly available test set to evaluate the performance of Uyghur NER, we will randomly choose 2000 sentences from Uyghur named entity relation corpus (Abiderexiti et al., 2016), in which tagged entity tags and relation types, checked the entity tags manually and used 1000 sentences as our Uyghur NER test set and another 1000 sentences as development set. The 1,500,000 Uyghur sentences crawled from the Tianshan website [5] is used to train the Uyghur word embeddings and the BIO tag schema is used where the B, I, O refer to the beginning, inside and outside of an entity, respectively.

4.2 Setup

General-Purpose NMT: We use the Transformer model (Vaswani et al., 2017) implemented in PyTorch in the fairseq-py (Ott et al., 2019) toolkit and all experiments are based on the "base" transformer model. We use word representations of size 512, feed-forward layers with inner dimension 2048, and multi-headed attention with 8 attention heads. We apply dropout with probability 0.3. Models are optimized with Adam using β_1= 0.9, β_2 = 0.98, and ε = 1e-8 . We use the same learning rate schedule as Vaswani et al. (Vaswani et al., 2017) , i.e., the learning rate increases linearly for 4,000 steps to 5e-4 (or 1e-3 in experiments that specify 2x *lr*), after which it is decayed proportionally to the inverse square root of the number of steps. We use label smoothing with 0.1 weight for the uniform prior distribution over the vocabulary. All experiments are run on 2 NVIDIA V100 GPUs interconnected by Infiniband.

[3] http://ee.dlut.edu.cn/CWMT2017/index_en.html
[4] http://sighan.cs.uchicago.edu/bakeoff2006/
[5] http://uy.ts.cn/

NER Model: We use the 300-dimensional word embeddings pretrained by Word2Vec, FastText, and Glove respectively. We set the character embedding size to be 100, character level LSTM hidden size to be 25, and word-level LSTM hidden size to be 100. For OOV words, we initialize an unknown embedding by uniformly sampling from range $[-\sqrt{\frac{3}{emb}},+\sqrt{\frac{3}{emb}}]$ where *emb* is the size of embedding, 300 in our case. We train the model for 100 epochs and optimize the parameters by Stochastic Gradient Descent (SGD) with momentum, gradient clipping, and learning rate decay. We set the learning rate (*lr*) and the decay rate (*dr*) as 0.01 and 0.05 respectively. To prevent overfitting, we apply dropout with a rate of 0.5 on outputs of the two Bi-LSTMs.

4.3 Results and Analysis

4.3.1 Comparison of tag combination strategy

1)Result of the STC Strategy

To obtain the accuracy of the three named entity recognition systems for the recognition of each entity type, we conducted experiments on the MSRA data set, and the experimental results are shown in Table 1.

NER system	Entity Type	Accuracy	Recall	F1
PaddleHub	LOC	81.09	66.77	73.24
	PER	83.16	80.08	81.59
	ORG	70.31	61.38	**65.54**
	ALL	79.51	69.86	74.37
Pyltp	LOC	86.26	71.81	**78.38**
	PER	90.73	61.53	73.33
	ORG	82.21	48.61	61.10
	ALL	86.88	63.53	73.40
THULAC	LOC	73.58	65.73	69.43
	PER	86.93	85.25	**86.08**
	ORG	78.06	16.30	26.97
	ALL	79.24	61.32	69.14

Table 1: The results of three Chinese NER system.

The experimental results show that PaddleHub has the best recognition for ORG while Pyltp for LOC and THULAC for PER. Therefore, the results of three NER systems are fused according to the STC strategy, and the fusion results are shown in Table 2. It can be seen that the recognition performance of the single and all entity is higher than the original system.

Strategy	Entity Type	Accuracy	Recall	F1
STC	LOC	90.10	70.56	79.14
	PER	86.93	85.25	86.08
	ORG	70.47	61.31	65.57
	ALL	84.70	73.26	**78.56**

Table 2: The results of the STC.

2)Result of the MTC Strategy

the result of the MTC is strategy shown in Table 3. Comparing the results of Table 2 and Table 3, it can be seen that the STC strategy is better than the MTC strategy for the recognition of Chinese named entities, and the following experiments are based on the STC strategy.

4.3.2 Baseline

To show the effectiveness of the proposed method, a strong baseline system is needed. In this paper, we will gradually explore the impact of different word alignment tools and different word vector models on

Strategy	Entity Type	Accuracy	Recall	F1
MTC	LOC	88.35	67.23	76.37
	PER	81.56	71.89	75.31
	ORG	69.67	54.38	61.08
	ALL	79.86	64.50	71.36

Table 3: The results of MTC.

cross-lingual entity migration, and finally, build a cross-language entity migration baseline system based on the parallel corpus and word alignment tools.

1) Comparison of Word Alignment Tools

word alignment accuracy is very important for word alignment based cross-lingual NER system and GIZA++ (Casacuberta and Vidal, 2007), fast_align (Dyer et al., 2013), and efmaral(Östling and Tiedemann, 2016) are currently popular word alignment tools. We will construct an Uyghur NER system using these three types of word alignment tools with the STC strategy based on the Uyghur-Chinese parallel corpus and The performance is shown in Table 4. It can be seen that efmaral word alignment tool has the best performance for our task.

Tools	Entity Type	Accuracy	Recall	F1
GIZA++	LOC	82.36	43.22	56.69
	PER	95.15	32.24	48.16
	ORG	73.07	41.11	52.62
	ALL	82.04	39.92	**53.71**
fast_align	LOC	80.61	56.36	66.43
	PER	96.93	36.35	52.87
	ORG	65.30	44.25	52.75
	ALL	79.08	48.37	**60.03**
efmaral	LOC	80.17	69.83	74.65
	PER	87.54	40.46	55.34
	ORG	66.88	53.48	59.44
	ALL	77.93	58.44	**66.69**

Table 4: Comparison of three word alignment tools.

2) Comparison of Word Embeddings

Word embeddings can provide rich semantic information and allow the system to better capture the semantic relevance between words. we will use the static word embeddings generated from Word2Vec, Glove, and FastText separately to initialize the network input and explore the effect of different word vectors on Uyghur NER construction. The Experimental results are shown in Table 5 and it can be seen that Word2Vec generated embeddings have good performance for our task.

Word Embedding	Accuracy	Recall	F1
Random	77.93	58.44	66.69
Glove	78.50	61.37	68.89
FastText	78.58	61.50	69.00
Word2Vec	79.17	62.75	**70.01**

Table 5: Comparison of three types of word embedding.

4.3.3 Analysis of Token-added Translation Method

We use the STC strategy to get named entity tags in Chinese and use the proposed token-added translation method to translated the entities to Uyghur to construct tagged NER corpus. Finally, train an Uyghur NER system using this data and the performance is shown in Table 6.

Method	Entity Type	Accuracy	Recall	F1
Token-add translation	LOC	66.45	50.91	57.65
	PER	79.96	69.57	74.41
	ORG	47.28	28.75	35.75
	ALL	66.70	50.33	**57.37**

Table 6: Uyghur NER based on token-added translation method.

From Table 6, it can be seen that the token-add translation method has worse performance compared with baseline. After analyzing the data, we found that only Uyghur stems are included in the special token while most of the affixes appended by the stem are being excluded. As an agglutinative language, Uyghur has rich affixes to express grammatical information in the sentence. For example, as shown in Figure 4, the original Chinese entity "新疆" is included in the Chinese bookmark (《》) and translated to Uyghur by MT, it can be found that the translated Uyghur entity also included in Uyghur bookmark («») while appended affix is excluded.

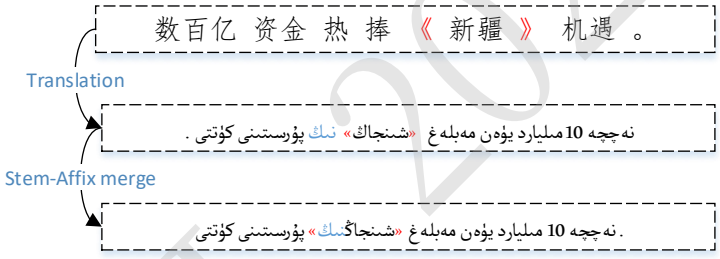


Figure 4: The example of entity boundary characters based entity translation

To prevent the problem, we apply a stem-affix merge method for translated Uyghur sentences and merge the stem with the followed word if it is affix. We train a new Uyghur NER system using handled corpus and the result is shown in Table 7. It can be seen that the combination of stem and affixes can effectively avoid the affix as a separate word in the corpus, thereby greatly improving the quality of the corpus and the performance of trained Uyghur NER system significantly, the f1 score is 3.79% higher than the baseline.

Method	Entity Type	Accuracy	Recall	F1
Stem-Affix merged	LOC	78.57	70.91	74.54
	PER	80.78	81.58	81.18
	ORG	67.05	61.32	64.06
	ALL	76.47	71.32	**73.80**

Table 7: Result of Stem-Affix merged method

5 Conclusion

Aiming at the lack of Uyghur named entity recognition training corpus, this paper proposes a cross-language named entity tag transfer method based on general machine translation and entity boundary token. First obtains the named entity tags of Chinese sentences in Chinese-Uyghur parallel corpus through a variety of Chinese named entity recognition tools and uses tag fusion strategies to fuse multi-source

tags, then select appropriate special symbols to surround the entities and uses Chinese-Uyghur neural machine translation system to translate the Chinese sentences to Uyghur. Finally, the Uyghur stems and affixes merge method is used to obtain a high-quality Uyghur named entity recognition corpus. The Uyghur NER system trained with this corpus achieved good performance, which was 3.79% points higher than the baseline system.

Acknowledgements

This work is supported in part by A Class Funded Project of the Western Light Talent Training Program of the Chinese Academy of Sciences(2017-XBQNXZ-A-005), NSFC(U1703133), The West Light Foundation of The Chinese Academy of Sciences(Grant No.2019-XBQNXZ-B-008), The National Key R&D Plan(2017YFC0822505-04).

References

Kahaerjiang Abiderexiti, Maihemuti Maimaiti, Tuergen Yibulayin, and Aishan Wumaier. 2016. Annotation schemes for constructing uyghur named entity relation corpus. In *2016 International Conference on Asian Language Processing (IALP)*, pages 103–107. IEEE.

He Bai, Yu Zhou, Jiajun Zhang, Liang Zhao, Mei-Yuh Hwang, and Chengqing Zong. 2018. Source-critical reinforcement learning for transferring spoken language understanding to a new language. *arXiv preprint arXiv:1808.06167*.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. *arXiv preprint arXiv:1906.05317*.

Emre Cakır and Tuomas Virtanen. 2019. Convolutional recurrent neural networks for rare sound event detection. *Deep Neural Networks for Sound Event Detection*, 12.

Francisco Casacuberta and Enrique Vidal. 2007. Giza++: Training of statistical translation models. *Retrieved October*, 29:2019.

Yufeng Chen, Chengqing Zong, and Keh-Yih Su. 2010. On jointly recognizing and aligning bilingual named entities. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 631–639. Association for Computational Linguistics.

Jason PC Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. A walk-based model on entity graphs for relation extraction. *arXiv preprint arXiv:1902.07023*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(Aug):2493–2537.

Chris Dyer, Victor Chahuneau, and Noah A Smith. 2013. A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648.

Maud Ehrmann, Marco Turchi, and Ralf Steinberger. 2011. Building a multilingual named entity-annotated corpus using annotation projection. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 118–124.

Meng Fang and Trevor Cohn. 2016. Learning when to trust distant supervision: An application to low-resource pos tagging using cross-lingual projection. *arXiv preprint arXiv:1607.01133*.

Meng Fang and Trevor Cohn. 2017. Model transfer for tagging low-resource languages using a bilingual dictionary. *arXiv preprint arXiv:1705.00424*.

Jerry R Hobbs, Douglas Appelt, John Bear, David Israel, Megumi Kameyama, Mark Stickel, and Mabry Tyson. 1997. Fastus: A cascaded finite-state transducer for extracting information from natural-language text. *Finite-state language processing*, pages 383–406.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Lifu Huang, Kyunghyun Cho, Boliang Zhang, Heng Ji, and Kevin Knight. 2018. Multi-lingual common semantic space construction via cluster-consistent word embedding. *arXiv preprint arXiv:1804.07875*.

Sungchul Kim, Kristina Toutanova, and Hwanjo Yu. 2012. Multilingual named entity recognition using parallel data and metadata from wikipedia. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 694–702. Association for Computational Linguistics.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Zhongwei Li, Xuancong Wang, Ai Ti Ai, Eng Siong Chng, and Haizhou Li. 2018. Named-entity tagging and domain adaptation for better customized translation.

Ying Lin, Shengqi Yang, Veselin Stoyanov, and Heng Ji. 2018. A multi-lingual multi-task architecture for low-resource sequence labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 799–809.

Liyuan Liu, Jingbo Shang, Xiang Ren, Frank Fangzheng Xu, Huan Gui, Jian Peng, and Jiawei Han. 2018. Empower sequence labeling with task-aware neural language model. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Gang Luo, Xiaojiang Huang, Chin-Yew Lin, and Zaiqing Nie. 2015. Joint entity recognition and disambiguation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 879–888.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Maihemuti Maimaiti, Aishan Wumaier, and Kahaerjiang Abiderexiti. 2018. Construction of uyghur named entity corpus. *Belt & Road: Language Resources and Evaluation*, page 2.

Mónica Marrero, Julián Urbano, Sonia Sánchez-Cuadrado, Jorge Morato, and Juan Miguel Gómez-Berbís. 2013. Named entity recognition: fallacies, challenges and opportunities. *Computer Standards & Interfaces*, 35(5):482–489.

Stephen Mayhew, Chen-Tse Tsai, and Dan Roth. 2017. Cheap translation for cross-lingual named entity recognition. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2536–2545.

Jian Ni, Georgiana Dinu, and Radu Florian. 2017. Weakly supervised cross-lingual named entity recognition via effective annotation and representation projection. *arXiv preprint arXiv:1707.02483*.

Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R Curran. 2013. Learning multilingual named entity recognition from wikipedia. *Artificial Intelligence*, 194:151–175.

Robert Östling and Jörg Tiedemann. 2016. Efficient word alignment with markov chain monte carlo. *The Prague Bulletin of Mathematical Linguistics*, 106(1):125–146.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958.

Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon infused phrase embeddings for named entity resolution. *arXiv preprint arXiv:1404.5367*.

Matthew E Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models. *arXiv preprint arXiv:1705.00108*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, pages 147–155.

Chen-Tse Tsai, Stephen Mayhew, and Dan Roth. 2016. Cross-lingual named entity recognition via wikification. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 219–228.

Arata Ugawa, Akihiro Tamura, Takashi Ninomiya, Hiroya Takamura, and Manabu Okumura. 2018. Neural machine translation incorporating named entity. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3240–3250.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Mengqiu Wang, Wanxiang Che, and Christopher D Manning. 2013. Joint word alignment and bilingual named entity recognition using dual decomposition. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1082.

Dingquan Wang, Nanyun Peng, and Kevin Duh. 2017. A multi-task learning approach to adapting bilingual word embeddings for cross-lingual named entity recognition. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 383–388.

Zhilin Yang, Ruslan Salakhutdinov, and William Cohen. 2016. Multi-task cross-lingual sequence tagging from scratch. *arXiv preprint arXiv:1603.06270*.

David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the first international conference on Human language technology research*, pages 1–8. Association for Computational Linguistics.

Joey Tianyi Zhou, Hao Zhang, Di Jin, Xi Peng, Yang Xiao, and Zhiguo Cao. 2019. Roseq: Robust sequence labeling. *IEEE transactions on neural networks and learning systems*.

Andrej Žukov-Gregorič, Yoram Bachrach, and Sam Coope. 2018. Named entity recognition with parallel recurrent neural networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 69–74.

Recognition Method of Important Words in Korean Text based on Reinforcement Learning

Feiyang Yang
Department of Computer Science and Technology, Yanbian University.
917936177@qq.com

Yahui Zhao *
Department of Computer Science and Technology, Yanbian University.
903873610@qq.com

Rongyi Cui
Department of Computer Science and Technology, Yanbian University.
cuirongyi@ybu.edu.cn

Abstract

The manual labeling work for constructing the Korean corpus is too time-consuming and laborious. It is difficult for low-minority languages to integrate resources. As a result, the research progress of Korean language information processing is slow. From the perspective of representation learning, reinforcement learning was combined with traditional deep learning methods. Based on the Korean text classification effect as a benchmark, and studied how to extract important Korean words in sentences. A structured model Information Distilled of Korean (IDK) was proposed. The model recognizes the words in Korean sentences and retains important words and deletes non-important words. Thereby transforming the reconstruction of the sentence into a sequential decision problem. So you can introduce the Policy Gradient method in reinforcement learning to solve the conversion problem. The results show that the model can identify the important words in Korean instead of manual annotation for representation learning. Furthermore, compared with traditional text classification methods, the model also improves the effect of Korean text classification.

1 Introduction

The languages of ethnic minorities have created the diversity of Chinese characters and are an important part of Chinese characters, providing important support for the development of national culture. However, the research on Korean natural language processing in my country is still in the development stage, and the related research is still relatively lagging behind South Korea and North Korea(Bi, 2011). For manual annotation of Korean sentences, the structure division requires a lot of energy and time. So for this problem, we associate the method of representation learning. Representation learning has been widely used in text classification, sentiment analysis, language reasoning and other fields in recent years. It is a basic problem in the field of artificial intelligence, and it is particularly important in the field of natural language processing. Therefore, we use this method as the core logic of the model, aiming at Korean text, identifying important words and performing sentence classification tasks on newly constructed sentences. The resulting structural representation does not require manual annotation, greatly reducing manpower and scientific research resources.

In order to find important Korean words in sentences, we use the effect of text classification as feedback in reinforcement learning. The current mainstream text classification models are roughly divided into four types: bag-of-words model, sequence model, structure representation model, and attention model. The bag-of-words representation model often ignores the order of words, such as deep average networks, self-encoders (Joulin A, 2017); the sequence representation model often only considers the words themselves, but ignores the phrase structure, such as CNN, RNN and other neural network models (Y, 2014); structural representation models often rely on pre-specified parse trees to construct structured representations, such as Tree-LSTM, recursive autoencoders (Zhu X, 2015); representation models based on attention mechanisms need to use input words or sentences The attention scoring function is used to

*Corresponding author.

construct a representation, such as Self-Attention (Yang Z, 2016), and the effect is very dependent on the reliability of scoring. In the existing structured representation model, the structure can be provided as input, or it can be predicted using the supervised method of explicit tree annotations, but few studies have studied the representation with automatically optimized structure. *Yogatama et al.* proposed to construct a binary tree structure for sentence representation only under the supervision of downstream tasks, but this structure is very complicated and the depth is too large, resulting in unsatisfactory classification performance (Yogatama D, 2017). *Chung et al.* proposed a hierarchical representation model to capture the latent structure of sequences with latent semantics, but the structure can only be found in the hidden space (Chung J, 2014). *Tianyang et al.* proposed a method that combines the strategy gradient method in reinforcement learning with the LSTM model in deep learning. The effect of text classification is used as the baseline for reinforcement learning to carry out unsupervised structuring, and its structuring effect is closer to Human, and the classification effect is significantly better than other mainstream models (Tianyang Zhang, 2018).

Inspired by *Tianyang et al.* , we propose a method that incorporates reinforcement learning. By identifying the structure related to the task , it does not require explicit structural annotations to construct a sentence representation. Among them, the structure discovery problem is transformed into a sequence decision problem. Using the policy gradient method in reinforcement learning (Policy Gradient), the value of the delayed reward function is used to guide the self-discovery of the structure. The definition of the reward function is expressed in the same text according to the structure. The classification effect of the classifier is derived,Each time the structured representation obtained needs to be used after all sequential decisions have been made,The model incorporates an attention mechanism and a baseline of reinforcement learning convergence on the basis of predecessors to optimize it. The main purpose of the model IDK we designed is to delete the unimportant words in the sentence, retain the words most relevant to the task, and construct a new sentence representation, in which the strategy network, the structured representation, and the classification network are seamlessly integrated. The strategy network defines the strategy used to discover the structure. The classification network calculates the classification accuracy based on the structured sentence representation, and passes the value of the reward function to the strategy network to promote the self-optimization of the entire network model.

2 Strategy network based on reinforcement learning combined with attention mechanism

The core idea of the strategy network is the Policy Gradient method , which is different from the traditional method. Instead of backpropagating through errors, the observation reward reward value is used to enhance or weaken the possibility of selecting actions. That is, the probability that a good action will be selected next time will increase; the probability that a bad action will be selected next time will decrease. A complete strategy represents a sequence of actions taken in each state in a round, The cumulative sum of the revenue generated by each action represents the round reward value. We use a random strategy $\pi\left(a_t|\mathbf{s}_\mathrm{t};\Theta\right)$. And use the reward generated by each round delay to guide strategy learning, For each state of the structured representation model generated each time, different actions are sampled. First, all the words of the entire sentence must be sampled for action, so as to determine the actions of all states corresponding to a sentence ,Secondly, the determined action sequence is passed into the representation model to generate a new structured representation; then the generated structured representation is passed into the classification network, and this representation model is used to calculate the classification accuracy $P(y|X)$, Finally, the calculated reward is used for strategy learning. In the loop iteration, a better strategy is found, so that a better structured representation is obtained, and then a better classification effect is obtained. The strategy is defined as follows:

$$\pi\left(a_t|\mathbf{s_t};\Theta\right)=\sigma\left(\mathbf{W}*\mathbf{s}_t+\mathbf{b}\right) \tag{1}$$

Which a_t represents the probability of choosing at; σ represents the sigmoid function; Θ represents the parameters of the strategy network. During the training, actions are sampled according to the probability

in equation 1. During the test, the action with the highest probability will be selected to achieve a better classification effect.

$$a_t^* = \text{argmax}_a \pi\left(a|\mathbf{s}_\mathrm{t};\Theta\right) \tag{2}$$

When all actions are sampled by the strategy network, the structured representation of the sentence is determined by the representation model, and the determined representation model is passed to the classification network to obtain, where y is the classification label, reward will be calculated from the predicted distribution, and There are also factors for thinking about the trend of structural choices. Therefore, the Policy Gradient method in the reinforcement learning algorithm is used to optimize the parameters of the strategy network (Keneshloo Y, 2019), so as to maximize the expected return, as shown in equation 3.

$$\begin{aligned} J(\Theta) &= \sum_{\mathbf{s_1}a_1\cdots\mathbf{s_L}a_L} P_\Theta\left(\mathbf{s_1}a_1\cdots\mathbf{s_L}a_L\right) R_L \\ &= \sum_{\mathbf{s_1}a_1\cdots\mathbf{s_L}a_L} p\left(\mathbf{s_1}\right)\prod_t \pi_\Theta\left(a_t|\mathbf{s_t}\right) p\left(\mathbf{s_{t+1}}|\mathbf{s_t},a_t\right) R_L \\ &= \sum_{\mathbf{s_1}a_1\cdots\mathbf{s_L}a_L} \prod_t \pi_\Theta\left(a_t|\mathbf{s_t}\right) R_L \end{aligned} \tag{3}$$

The reward calculation is only for one round, because the state at step $t+1$ is completely determined by the state at step t, so the probability sum is 1. Through the likelihood ratio technique, the following gradient update strategy network (Sutton R S, 2000) is finally used, where N represents the round number. In the iterative process of reinforcement learning, the variance is generally large. If the loss value is always positive, the direction of the iteration is easy to move toward. It has been proceeding in the wrong direction, so the introduction of b as a baseline can accelerate convergence, as shown in equation 4.

$$\nabla_\Theta J(\Theta) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{L}(R_L - b)\nabla_\Theta \log \pi_\Theta\left(a_t|\mathbf{s_t}\right) \tag{4}$$

On this basis, the attention mechanism is introduced into the strategy network, and the Encoder-Decoder framework is adopted. Use Bi-LSTM (Graves A, 2005) as the encoder model, LSTM as the decoder model, and the output of the structured representation model as the input of the strategy network, because the core logic of the attention model is from focusing on the whole to focusing on the core. The purpose of this article is the same, so the combination of reinforcement learning and Soft Attention mechanism to make up for the shortcomings of the predecessors in the traditional attention model in the text classification process, relying heavily on the scoring function (Bahdanau D, 2015). After introducing the attention mechanism, the corresponding actions in each state are output as shown in Figure 1.

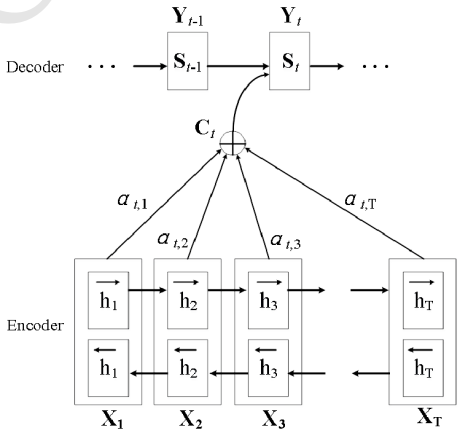


Figure 1: Soft attention mechanism

$$S_t = f(S_{t-1}, Y_{t-1}, C_t) \tag{5}$$

$$C_t = \sum_{j=1}^{T} \alpha_{tj} h_j \tag{6}$$

$$\alpha_{tj} = \frac{exp(e_{tj})}{\sum_{k=1}^{T} exp(e_{tk})} \tag{7}$$

$$e_{tj} = g(S_{t-1}, h_j) \tag{8}$$

Where h_j is the hidden vector of the input, f is the tanh activation function; C is the attention distribution; and α_{tj} is the attention obtained by each input. After introducing the attention mechanism, the global observation can be better, so that the generated action sequence is optimized in two aspects of the strategy gradient and the attention mechanism, thereby improving the model effect.

3 Information Distilled of Korean(IDK)

3.1 The main idea of the model

Our ultimate goal is to reconstruct more concise Korean sentences by finding important, task-related words, and at the same time get a structural representation for Korean text classification, and improve text classification through optimized structured representation. While the text classification has been improved, the structured representation has also been optimized, and the two promote each other. The model consists of three parts: strategy gradient network, structured representation model, classification network. The strategy network adopts a random strategy to sample the actions corresponding to each state, sampling until the end of the sentence, and generating a sequence of actions for the current sentence. Then the structured representation model converts the action sequence into a structured representation. Based on this idea, the IDK model is proposed. The classification network classifies based on the obtained structured representation and provides the reward function calculation for the strategy network. Since a complete structured representation can be given to calculate the reward of the current structured representation, this process can be solved by the Policy Gradient method. The specific model is shown in Figure 2.

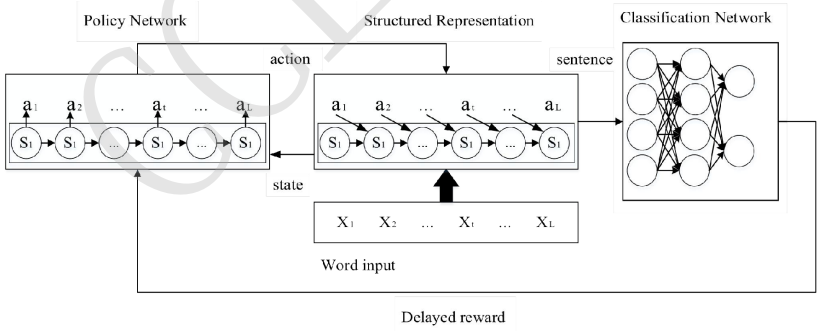


Figure 2: Network model structure diagram based on reinforcement learning

The model is interleaved by three parts. The state representation of the strategy network comes from the structured model. The structured model is generated by the action sequence of the strategy network and the input of the sentence. The classification network is classified and predicted by the resulting structured model. Strategy The network obtains the reward function value from the classification effect obtained by the classification network, thereby guiding the strategy to learn a better structured representation.

3.2 Model specific construction

Inspired by the ID-LSTM of *Tianyang et al.* , an attention mechanism was introduced into the original strategy network to double optimize the action sequence. The main idea of IDK proposed in our

thoughts is to build a structured representation of a sentence by extracting important words and deleting irrelevant words in the sentence. In the well-known Chinese and English text processing tasks, there are many examples such as: "with", "and ", "in", "of" and other stop words, such stop words rarely help complete text processing tasks, so it is necessary to refine important features in sentences.Different from the traditional method, this method does not create a stop word list, and deletes all stop words together. Because many stop words often constitute a special phrase structure, combing the logical relationship between the context and deleting it directly without filtering, it will cause the loss of language content and semantic information, so this method is chosen in our thoughts to purify the final representation form, thus Concentrate sentences to enhance the effectiveness of downstream classification tasks.

The IDK model converts the sequence of actions transferred from the strategy network into a structured representation of sentences,Given a sentence X shaped like $X = x_1 x_2 \cdots x_L$, After the sentence X is transferred to the strategy network, each action a_i corresponding to the word position x_i is selected from keeping the current word or deleting the current word, which satisfies the following rules:

$$\mathrm{S_t}, \mathrm{C_t}, = \begin{cases} \mathrm{S_{t-1}}, \mathrm{C_{t-1}}, & a_t = Delete \\ \Phi\left(\mathrm{S_{t-1}}, \mathrm{C_t}, \mathrm{Y_{t-1}}\right), & a_t = Retain \end{cases} \tag{9}$$

The Φ represents the function of the entire model (including gating unit and update function) , S is the hidden state corresponding to the Decoder cell unit; Y is the output corresponding to the hidden state of the cell unit; C is the hidden state distribution of the Encoder cell unit;When deleting a word, the storage unit and hidden state attention distribution of the current position will be copied from the previous position.

For classification, the last hidden state of the IDK model is used as the input of the classification network, where $\mathbf{W_s} \in R^{d \times K}$, $\mathbf{b_s} \in R^K$ is the parameter of the classification network, d is the dimension of the hidden state, is the label of the category, K is the number of classification clusters, the classification network is based on the IDK model The obtained structured representation produces a probability distribution on the class label, as shown in equation 10.

$$P(y|X) = \mathrm{softmax}\left(\mathbf{W}_\mathrm{s}\mathrm{S_L} + \mathbf{b}_\mathrm{s}\right) \tag{10}$$

To calculate reward, take the logarithm of the output probability calculated by the classification network in equation 10, as shown in equation 11,where c_g stands for classification label. In order to make the model use as few words as possible, the two items in the formula are controlled by calculating the ratio of the number of deleted words in the sentence to the length of the sentence. Maintain accuracy and balance the two effects of using a few words,Where L' represents the number of deleted words, and γ represents the hyperparameter between 0 and 1 that balances the two terms.

$$R_L = \log P\left(c_g|X\right) + \gamma L'/L \tag{11}$$

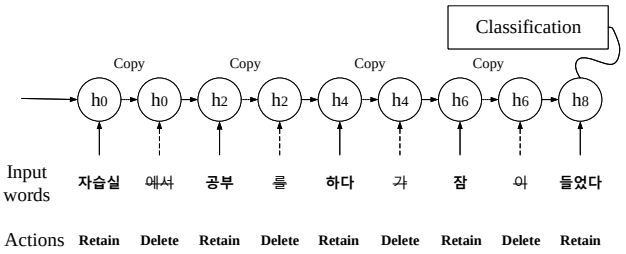


Figure 3: ID-Korean model

As shown in Figure 3, after word recognition is performed on Korean text, only important words are retained, and the ratio of the amount retained to the number deleted is controlled by the reward function.

4 Experimental results and analysis

4.1 Data set description

The data set used in the experiments in this article comes from the corpus constructed by the laboratory to undertake the ”China-Korea Science and Technology Information Processing Comprehensive Platform” project. It is further organized into a corpus composed of abstracts of Korean scientific and technological literature. There are about 30,000 documents, divided into 13 categories such as animals, oceans, and aerospace. Each Category randomly selects documents according to a 7:3 ratio to form a training set and a test set. (MingJie Tian, 2018)(Xianyan Meng, 2019). The details of the data set are shown in Table 1.

Table 1: Data set introduction

category	Number of entries	category	Number of entries
Animal	4582	Botany	6172
Microorganism	5472	Biotechnology	1215
Biomedical Science	2752	Climate	708
The marine environment	810	Geology	1735
Marine technology	819	Materials Engineering	781
Measurement Technology	1728	Aerospace	4436
Others	1478		

For Korean corpora, sentences are composed of phrases separated by spaces, and these phrases are usually followed by auxiliary words or endings. According to the grammatical characteristics of Korean,In the preprocessing process, the Hannanum word segmentation system developed by the Korea University of Science and Technology is used to cut out the auxiliary words and endings in the phrase, and restore the predicate to the word itself.

4.2 Model training

When training a classification network, a cross-entropy loss function is used, in which the probability distribution of the ground truth of the corresponding sentence is coded by one-hot, as shown in equation 12.

$$\mathcal{L} = -\sum_{X\in\mathcal{D}}\sum_{y=1}^{K}\hat{p}(y,X)\log P(y|X) \tag{12}$$

GloVe training is used to initialize the word vector in the representation model (Pennington J, 2014), the dimension is set to 256 dim, and it is updated together with other parameters. When using gradient descent to update parameters, the speed of model learning depends on the learning rate and partial derivative value,To smooth the update of Policy Gradient, multiply the suppression factor γ by equation 4 and set it to 0.1, γ is set to 0.2 in the IDK, equation 11,In the training process, the Adam optimizer (Kingma D, 2015) is used to optimize the parameters, the learning rate is 0.0005, the Dropout tailoring is used before the classification network classification, the probability is 0.5, and the mini-batch is 5.

In the model training process, the classification accuracy rate using the IDK model changes with the number of iterations as shown in Figure 4. The text classification accuracy rate is about 68% at the beginning of the training, and the accuracy rate increases as the number of iterations increases. When the number of iterations is between 400 and 600, the classification accuracy of the model rises fastest. After 800 iterations, the classification accuracy of multilingual text tends to be stable, indicating that the training of the neural network model has converged. At this time, the text classification of the IDK model The accuracy rate reached 83.23%.

4.3 Comparative Experiment

In the comparative experiment, a variety of baselines were selected: basic neural network model CNN without specific structure; LSTM; Bi-LSTM; attention model Self-Attention (Lin Z, 2017). The dimen-

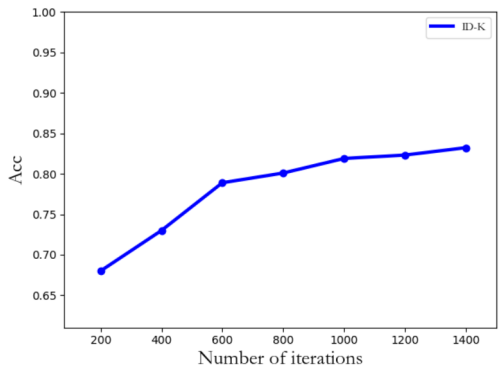


Figure 4: Soft attention mechanism

sion of the word vector used by these baselines is the same as this article, and the effect is shown in Table 2.

Table 2: Accuracy under different classifiers

Models	ACC	Models	ACC
CNN	78.5	T-BLSTM-CNN	81.68
LSTM	74.6	Self-Attention	82.91
Bi-LSTM	78.14	**IDK**	**83.23**

As shown in Table 2, the classification effect shows that: in different models, our method performs well in classification. When comparing with previous methods, we combine reinforcement learning and attention model to use a self-discovery structure and Optimize the structured representation model for text classification,Different from the predecessors who only focused on the sequence model and its optimization, this paper designs the model from two aspects of reinforcement learning and attention. Its classification effect also proves the effectiveness and necessity of representation learning and text structuring.

4.4 Examples of structured presentation results

Original	도시는 인류 활동의 영향을 가장 크게 받는 지구의 표면 이고 도시 시스템의 탄소 순환은 세계 및 지역의 탄소 순환에서 중요한 위치와 역할을 한다（Cities are the surface of the earth, which is most affected by human activity, and the carbon cycle of urban systems plays an important position and role in the carbon cycle of the world and regions.）
IDK	**도시는 인류 활동의 영향을 ~~가장 크게~~ 받는 지구의 표면 이고 도시 시스템의 탄소 순환은 ~~세계 및 지역의 탄소 순환에서~~ 중요한 위치와 역할을 ~~한다~~**
Original	곤충 병원체에서 직접 샘플을 채취하고 검출 및 정량화하는 것은 곤충 유행병학 조사에서 병원체 풍도를 직접 반영할 수 있다（Taking, detecting, and quantifying samples directly from insect pathogens can directly reflect pathogen wind levels in insect epidemiological surveys.）
IDK	**곤충 병원체에서 직접 샘플을 채취하고 검출 ~~및~~ 정량화하는 것은 곤충 유행병학 ~~조사에서~~ 병원체 풍도를 직접 반영할 ~~수 있다~~**

Figure 5: Structural representation example

The specific structured example is shown in Figure 5. In the IDK model, the strike through indicates the word to be deleted on the original text. The Korean texts mood words, auxiliary words, and some

adjectives are deleted, and important part of the nouns are retained. The larger the model segmentation structure is, the closer it is to manual annotation, and the original text is more useful for downstream text classification tasks after being structured.

5 Conclusion

We combined reinforcement learning methods to learn Korean sentence representations by finding important words related to the task. In the framework of reinforcement learning and attention mechanism, this paper uses the IDK model, which is used to extract task-related words and express them in purified sentences. Among them, reinforcement learning uses the accuracy rate of text classification as a baseline to optimize the action sequence, and the action sequence can generate a text structure representation that is more suitable for classification. An attention mechanism is introduced in the process of action sequence generation to compensate for the variance of the reinforcement learning method. The disadvantage of being too large and difficult to fit, compared with the traditional attention model, not only has its advantages of taking into account the overall situation, but also adds a more ingenious way to improve the accuracy of downstream tasks, and the experimental results have performed well. Experiments show that our method can find important words related to tasks without explicit structure annotation. The model not only improves the effect of Korean text classification, but also works well in the task of processing Korean text important word recognition.

6 Acknowledgements

This work was supported by the National Language Commission Scientific Research Project (YB135-76); Yanbian University Foreign Language and Literature First-Class Subject Construction Project (18YLPY13).

References

Bengio Y Bahdanau D, Cho K. 2015. Neural machine translation by jointly learning to align and translate. In *International Conference on Machine Learning*, page arXiv:1409.0473v7.

Yude Bi. 2011. A research on korean natural language processing. *Journal of Chinese Information Processing*, 25(06):166–169+182.

Cho K Chung J, Gulcehre C. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop*.

Jrgen Schmidhuber Graves A. 2005. Frame wise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks, 2005*, 18(5-6):602–610.

Bojanowski P Joulin A, Grave E. 2017. Bag of tricks for efficient text classification. In *The European Chapter of the ACL (EACL)*, pages 427–431.

Reddy C K Keneshloo Y, Ramakrishnan N. 2019. Deep transfer reinforcement learning for text summarization. *Society for Industrial and Applied Mathematics*, page arxiv1810.06667v2.

Ba J. Adam Kingma D. 2015. A method for stochastic optimization. In *International Conference on Learning Representations*.

Santos C N Lin Z, Feng M. 2017. A structured self-attentive sentence embedding. In *International Conference on Learning Representations*.

RongYi CUI MingJie Tian, YaHui Zhao. 2018. Identifying word translations in scientific literature based on labeled bilingual topic model and co-occurrence features. In *The 17th China National Conference on Computational Linguistics*, pages 79–92.

Manning C D Pennington J, Socher R. 2014. Glove:global vectors for word representation. In *In EMNLP, 2014*, pages 1532–1543.

Singh S P Sutton R S, McAllester D A. 2000. Policy gradient methods for reinforcement learning with function approximation. In *NIPS, 2000*, pages 1057–1063.

Li Zhao Tianyang Zhang, Minlie Huang. 2018. Learning structured representation for text classification via reinforcement learning. In *the Association for the Advance of Artificial Intelligence*.

Yahui Zhao Xianyan Meng, Rongyi Cui. 2019. Multilingual text classification method based on bidirectional long-short memory unit and convolutional neural network. *Computer Application Research*, 132(04):1–6.

Kim Y. 2014. Convolutional neural networks for sentence classification. In *The 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

Dyer C Yang Z, Yang D. 2016. Hierarchical attention networks for document classification. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1480–1489.

Dyer C Yogatama D, Blunsom P. 2017. Learning to compose words into sentences with reinforcement learning. In *International Conference on Learning Representations*.

Guo H Zhu X, Sobihani P. 2015. Long short-term memory over recursive structures. In *International Conference on Machine Learning*, pages 1604–1612.

Mongolian Questions Classification Based on Mulit-Head Attention

Guangyi Wang, Feilong Bao, Weihua Wang*
College of Computer Science, Inner Mongolia University, China
Inner Mongolian Key Laboratory of Mongolian
Information Processing Technology, China
wanggycs@163.com
{csfeilong, wangwh}@imu.edu.cn

Abstract

Question classification is a crucial subtask in question answering system. Mongolian is a kind of few resource language. It lacks public labeled corpus. And the complex morphological structure of Mongolian vocabulary makes the data-sparse problem. This paper proposes a classification model, which combines the Bi-LSTM model with the Multi-Head Attention mechanism. The Multi-Head Attention mechanism extracts relevant information from different dimensions and representation subspace. According to the characteristics of Mongolian word-formation, this paper introduces Mongolian morphemes representation in the embedding layer. Morpheme vector focuses on the semantics of the Mongolian word. In this paper, character vector and morpheme vector are concatenated to get word vector, which sends to the Bi-LSTM getting context representation. Finally, the Multi-Head Attention obtains global information for classification. The model experimented on the Mongolian corpus. Experimental results show that our proposed model significantly outperforms baseline systems.

1 Introduction

When people read a specific sentence on a flyer or some magazine, they can understand the context or intent of the sentence. And they can also extract information from the sentence. How to make a computer think like a human. Natural Language Processing (NLP) and Natural Language Understanding (NLU) study how to make the computer understand the semantics of natural language. The computer uses natural language to communicate with people to realize human-machine interaction. Deep learning models have achieved state-of-the-art performance in various natural language processing tasks such as text summarization (Rush et al., 2015), question answering (He and Golub, 2016) and machine translation (Kudo, 2018). In recent years, question answering is a key technology in intelligent applications. It has aroused widespread concern. Pipeline the first task of question system is to classify the domain of the dialogue after the user enters the message (text or voice). Question classification divides questions into several semantic categories. The machine gets a predicted category of the dialogue and the system returns a concise and accurate answer. The understanding of questions provides constraints for improving the accuracy of question answering system. Moldovan et al. (2003) have studied the influence of each part of the question answering system on the system performance. The question classification recognition has the greatest influence on the system performance. Therefore, to get a good question answering system, it is necessary to design a high accuracy model of question classification.

However, the research of the Mongolian questions classification is very fewer. The reason is that Mongolian corpus is scarce and there is no public Mongolian corpus. Data collected from internet are noisy and uncertain in terms of coding and spelling. The word-formation is different from Chinese and English. It consists of roots, stems and affixes. These problems result in unlimited vocabulary. The existing short text classification methods are not effective. How to classify the questions accurately is a complicated problem.

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1026-1034, Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

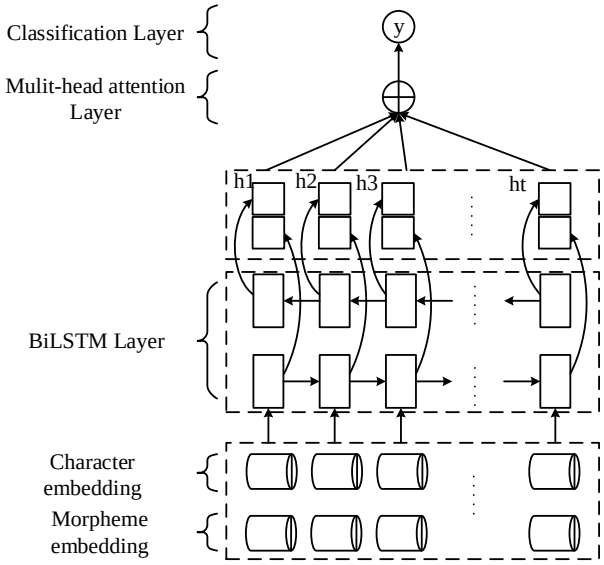


Figure 1: The model architecture of MA-B.

In this article, the training data were crawled from the Mongolian web sites. After cleaning the invalid data, we constructed a question classification data set. We propose a method of the Mongolian question classification, which combines the Bi-LSTM model with the Multi-Head Attention mechanism. As shown in Figure 1, the model is named MA-B. To better learn semantic information from sentences, we introduce the morphemes representation. The character vector and the morpheme vector are concatenated to get word vector. It sends to Bi-LSTM getting context representation. The Multi-Head Attention mechanism extracts relevant information from different dimensions. In the classification layer, we use the softmax classifier to output the probability of each category.

The paper is organized as follows: Section 2 gives the related work. Section 3 presents the question classification method in detail. Section 4 shows the experiments and results. Section 5 summarizes the full text and give some future works.

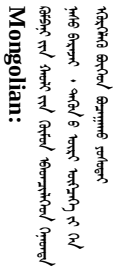
2 Related Work

Question classification is a kind of short text classification (Alsmadi and Gan, 2019). There have been many studies on questions classification. Chinese and English, which are rich in resources, have achieved good results. The traditional method was based feature engineering such as bag of words (BOW) and n-gram. Both were combined with term frequency-inverse document frequency (TF-IDF) and other element features as text features. However, these methods ignore the context semantic information. There were some methods based machine learning, including Nearest Neighbors (NN) (Yang and Liu, 1999), Naive Bayes (McCallum et al., 1998), and Support Vector Machine (SVM) (Cristianini and Shawe-Taylor, 2010). In (Wang et al., 2013), the authors utilized the external knowledge base for text classification. In recent years, researchers have tried to extract semantic information from sentences via deep learning. The combination of TextCNN (Kim, 2014), TextRNN (Liu et al., 2016), LSTM (Xiao et al., 2018), TextGCN (Yao et al., 2019), with word embedding has been widely used in text classification.

There are some researches on rare resource languages to classify questions. For example, Uyghur is also a few resource language and have complex word-formation. Parhat et al. (2019) proposed a method of Uyghur short text classification based reliable sub-word morphology. Mongolian language processing has been further developed, such as morphological segmentation (Wang et al., 2019b), spelling correction (Lu et al., 2019), named entity recognition (Wang et al., 2019a). The Mongolian question classification needs to be solved urgently.

Mongolian alphabet	Latin letter	Mongolian alphabet	Latin letter	Mongolian alphabet	Latin letter	Mongolian alphabet	Latin letter	Mongolian alphabet	Latin letter
ᠠ	a	ᠡ	E	ᠺ	k	ᠮ	m	ᠲ	t
ᠡ	e	ᠨ	n	ᠻ	K	ᠯ	l	ᠳ	d
ᠢ	i	ᠩ	N	ᠴ	C	ᠯᠠ	L	ᠶ	y
ᠤ	q	ᠪ	b	ᠵ	Z	ᠽ	Z	ᠵ	c
ᠥ	v	ᠫ	p	ᠹ	H	ᠼ	Q	ᠷ	j
ᠦ	o	ᠸ	w	ᠺ	R	ᠰ	s	ᠷ	r
ᠧ	u	ᠹ	f	ᠭ	g	ᠬ	x	ᠭ	h

Figure 2: Comparison between Latin alphabet and Mongolian alphabet.



Latin:
kqmpani-y'in havli-y'in homun ebedcileged genedte nasv barajai , tegun-u" ori ogcege-y'i hen egurgelehu boged bvcagahv yqsqtai
Means:
Who should bear and return the debts of the company due to the sudden death of the legal person?
Category:
Company Law

Figure 3: Example of traditional Mongolian script, Latin transliterature, category tag and their meanings.

3 Model Architecture

In this section, we will introduce this model from bottom to up. The Mulit-Head Attention mechanism can fully capture the long-distance text features. But it is difficult to deal with the sequence information. The recurrent neural network can effectively obtain the context order information of sequences. It can effectively supplement the Mulit-Head Attention mechanism. As depicted in Figure 1, MA-B model is proposed by combining Bi-LSTM network with Mulit-Head Attention mechanism.

3.1 Morpheme Vector

Mongolian is a kind of agglutinative language, which consists of roots, stems and suffixes. The Chinese words need to be segmented, which is called Chinese word segmentation (Zhou et al., 2019). There are natural spaces between words in Mongolian, but morphological segmentation is needed in Mongolian because the root and stem suffixes of Mongolian words are connected with many different endings. The Mongolian word formation features result in unlimited vocabulary. This paper uses Latin to deal with Mongolian. The contrast between Latin characters and Mongolian letters is shown in Figure 2.

In this paper, we introduce Mongolian morphemes representation. The suffix is segmented by identifying a narrow uninterrupted space (NNBS) (U+202F, Latin: ”-”) to make it an independent training unit. As shown in Figure 3, after segmentation the suffix, the sentence will be turned into *“kqmpani -y’in havli -y’in homun ebedcileged genedte nasv barajai , tegun -u” ori ogcege -y’i hen egurgelehu boged bvcagahv yqsqtai”*. The length of this sentence is changed to 19 units.

The Word2vec is a common tool for training word vectors. The Word2vec (Mikolov et al., 2013) contains CBOW (Continuous Bag of Word) and Skip-gram. This paper uses the Skip-gram model to train morpheme vectors. Given a sequence of morphemes $\mathbf{m} = m_1,...,m_T \in M$. The output of the model is a probability distribution. The morpheme skip-gram model predict contextual morphemes when given current morpheme.The formula is as follows:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(m_{t+j} \mid m_t) \tag{1}$$

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1026-1034,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

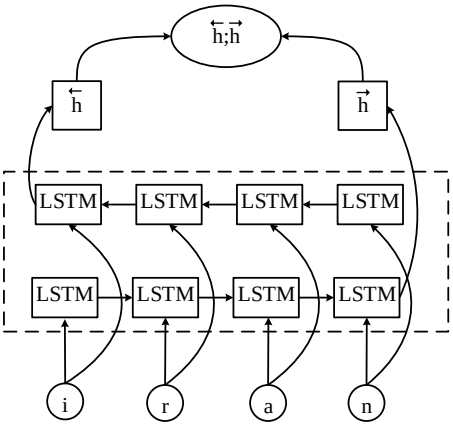


Figure 4: The character embedding of Mongolian morpheme.

where c is the size of the context window for the current central morpheme m_t. The simplest formulation of the probability $p(m_{t+j}|m_t)$ is:

$$p(o \mid c) = \frac{\exp({u_o}^T v_c)}{\sum_{m=1}^{M} \exp({u_m}^T v_c)} p \tag{2}$$

where o is the ids of the output morpheme, c is the ids of the central morpheme, u is the output morpheme vector, v is the input morpheme vector, and M is the morphemes set.

3.2 Character Vector

To better represent the semantic information in sentences, we use the Bi-LSTM model to learn the character embedding from training data. The character Bi-LSTM network consists of forward LSTM layer and backward LSTM layer. The forward layer can learn word prefix information. And the backward layer learns the morphological information. Both layers are connected to the same output layer. We get the character representation. As shown in Figure 4 is the structure of Bi-LSTM character embedding network.

3.3 Bi-LSTM Layer

LSTM (Hochreiter and Schmidhuber, 1997) network is a special type of recursive neural network, which can capture the context order information of the sequence and solve the problem of long dependency. LSTM is a variant of RNN. It introduces some gates to solve the gradient problem. LSTM calculates an output vector according to the current input and the output of the previous unit. The output vector is then used as input to the next unit.

LSTM is mainly composed of four parts: storage unit c_t, input gate i_t, output gate o_t, and forget gate f_t. Those gates control the proportion of history to omit or to store in the next time stamp. LSTM calculates the output vector based on the current input and the output of the previous unit, which is then used as the input of the next unit. The calculation formula is as follows:

$$\begin{aligned}
f_t &= \sigma(W_{(f)} x_t + U_{(f)} h_{t-1} + b_{(f)}) \\
i_t &= \sigma(W_{(i)} x_t + U_{(i)} h_{t-1} + b_{(i)}) \\
o_t &= \sigma(W_{(o)} x_t + U_{(o)} h_{t-1} + b_{(o)}) \\
c_t &= \tilde{c} + i_t \odot \tanh(W_{(c)} x_t + U_{(c)} h_{t-1} + b_{(c)}) \\
\tilde{c} &= f_t \odot c_{t-1} \\
h_t &= o_t \odot \tanh(c_t)
\end{aligned} \tag{3}$$

where i_t is the input gate and o_t is the output gate. The forget gate f_t is a reset memory unit. x_t the input vector. h_t represents the hidden unit vector. σ is the point product sigmoid function. $\odot$ represents the

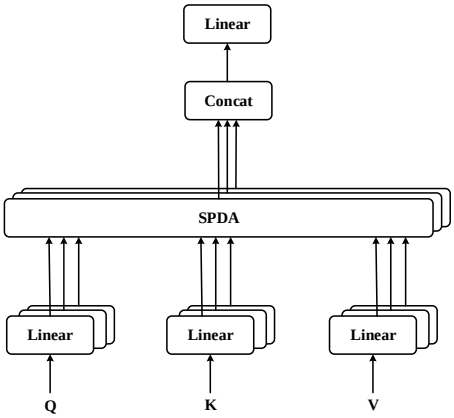


Figure 5: The flowchart of scaled dot-product attention.

corresponding multiplication of elements. W_i, W_f, W_o is the weight matrix of the input gate, the forget gate, and the output gate respectively. U_f, U_i, U_c, U_o denote the different weight matrices for hidden h_t. And b_i, b_f, b_c, b_o represent the bias.

The LSTM can only encode historical information, but it is often not enough. The paper adopted the Bidirectional LSTM network which is composed of forward LSTM and backward LSTM. So, *h* is the concatenate of $\overleftarrow{h_t}$, $\overrightarrow{h_t}$ and *h* is shown as below.

$$h = \overleftarrow{h_t} + \overrightarrow{h_t} \tag{4}$$

where $\overrightarrow{h_t}$ is the forward output vector and $\overleftarrow{h_t}$ is backward output vector.

3.4 Mulit-Head Attention Layer

In recent years, *Transformer* (Vaswani et al., 2017) model is very popular, which used in NLP tasks. It uses the Mulit-Head Attention mechanism. The Mulit-Head Attention is the optimization of the traditional attention mechanism and it is used to fully capture the features of long distance and obtain the global information. It firstly projects the input into multiple feature spaces, then compute correlation score and utilize the scores to weight context representation, finally concatenates vectors weighted as output.

The input of Mulit-Head Attention mechanism consists of *Q*(queries), *K*(keys) and *D*(dimension). The merging vector output from Bi-LSTM layer is the input of *Q, K* and *V*. Then *Q, K, V* are linearly transformed and finally input into scaled dot-product attention(SDPA). This process calculates one head at a time. As shown in Figure 5, the model independently compute dot product attention for each part $head_i$. The details are described below.

$$SDPA(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5}$$

where *softmax* is a normalization function.The calculation formula is as follows:

$$softmax(g\,(Q, K)) = \frac{e^{g(Q,K)}}{\sum_i e^{g(Q,K_j)}} \tag{6}$$

where *g(Q,K)* represents the similarity between *Q* and *K*. Similarity calculation is obtained by *Q* and *K* point product operation.

Then, all the scaled dot-product attention results of *m* times, are concatenated and the value obtained by a linear transformation is used as the result of the Mulit-Head Attention model.

$$head_i = SDPA\left(QW_i^Q, QW_i^K, VW_i^V\right) \tag{7}$$

$$MA(Q, K, V) = Concat(head_1, \cdots, head_h) \tag{8}$$

where W_i^Q, W_i^K and W_i^V are projection matrices corresponding to *Q, K* and *V* respectively.

3.5 Classification Layer

Questions classification is a multi classification problem. The classification layer consists of two parts: a linear layer and a softmax layer. The text vector *h* can be used as features for questions classification.

$$y = softmax(W_h h + b_h) \tag{9}$$

We use the negative log likelihood of the correct classification as training loss.

$$L = -\sum_t \log y_{ti} \tag{10}$$

where *i* is the label of the text *t*.

4 Experiments

Our model is trained on the selected data set. By evaluating the classification results and comparing with baseline, we can evaluate the questions classification performance of the model.

4.1 Setting Up

The training data mainly comes from China Mongolian News Network, People's Daily Online (Mongolian version), China Mongolian Broadcasting Network, China Judgements Online (Mongolian version) and other web sites. After removing duplicate data and cleaning invalid data, 115688 sentences were obtained by manual correction and annotation. The data of question classification is divided into eleven categories, as shown in Table 1. We divided the dataset into train, dev and test with the percent 80%, 10% and 10%,respectively.

Label	Categories	Number	Label	Categories	Number
0	Marriage and Family	10359	6	Property Disputes	9435
1	Labor Disputes	9621	7	Infringement	11258
2	Traffic Accident	11421	8	Company Law	9900
3	Credit and Debt	9401	9	Medical Disputes	8743
4	Criminal Defense	13020	10	Administrative Litigation	13872
5	Contract Disputes	8658			

Table 1: The data is divided into eleven categories.

4.2 Evaluation Metrics

Question classification is a multi classification task, so we use *precision, recall* and F_1 as the evaluation index. These metrics are calculated as:

$$\begin{aligned} P &= \frac{TP}{TP + FP} \\ R &= \frac{TP}{TP + FN} \\ F_1 &= \frac{2PR}{P + R} \end{aligned} \tag{11}$$

where *TP* is the number of correctly predicted question sentences. *FP* is the number of sentences that predicted as question sentences, but in actuality those are negative class. If the prediction is failed, and the positive class is predicted as a false negative(*FN*). F_1 is the harmonic mean of precision and recall.

4.3 Results

In this paper, TextCNN, Bi-LSTM and Attention-BiLSTM model are used as baselines. TextCNN (Kim, 2014) applies Convolutional Neural Networks(CNN) to text classification tasks. The key information in sentences is extracted by using multiple different size kernels. So it can better capture the local correlation. TextCNN is a commonly used baseline. Bi-LSTM and Attention-BiLSTM are commonly used models to extract text features. Attention is essentially an automatic weighted summation mechanism that makes the model more capable of handling long sequences.

The experiment is divided into two forms: 1) whether the combination of character vector and morpheme vector affects the performance of the model. 2) whether the introduction of Mulit-Head Attention mechanism into the model affects the performance of the model. The experimental results are shown in Table 2.

Model	Character embedding	Morpheme embedding	P(%)	R(%)	F_1(%)
TextCNN	Yes	No	82.57	79.36	80.93
TextCNN	Yes	Yes	83.27	81.42	82.33
Bi-LSTM	Yes	No	83.22	83.95	83.58
Bi-LSTM	Yes	Yes	84.56	83.93	84.24
Att-BiLSTM	Yes	No	84.67	83.91	84.31
Att-BiLSTM	Yes	Yes	85.13	84.89	85.01
MA-B	Yes	No	86.58	86.01	86.29
MA-B	Yes	Yes	**86.71**	**86.51**	**86.61**

Table 2: Comparison of experimental results.

We compare the results from the table:

1) Introducing morpheme features in the embedding layer can improve performance. The F_1 value of MA-B model remains the highest among all models. About 1.6% improvement compared with the highest Att-BiLSTM model in the baseline model.

2) In the whole model, the introduction of Mulit-Head Attention mechanism can effectively improve the model classification performance. Compared with Bi-LSTM model, our model is improved by about 2.2%. Compared with Att-BiLSTM model, our model's classification ability is also significantly enhanced.

The reasons for the above results are as follows:

1) When judging the questions categories of sentences, we mainly consider the semantic information of sentences. In Mongolian word formation, morpheme vector can learn more syntactic and semantic information. Therefore, the introduction of morpheme features into the model will have a good performance.

2) Compared with the baseline model, the advantage of MA-B model is to use BiLSTM network to obtain the internal relationship between the front and back directions of sentences and get local information. The long-distance feature is fully captured by Mulit-Head Attention mechanism, and relevant information is learned from different dimensions and representation subspaces.

5 Conclusion

In this paper, Bi-LSTM and Mulit-Head Attention mechanism are used to model Mongolian corpus texts. By combining the ability of multi head attention to obtain global information with the ability of Bi-LSTM to obtain local sequence information, a better effect has been achieved. At the same time, in order to make the model better learn the text semantic information, Mongolian morphemes representation are further introduced.

However, there is a lot of room for improvement in the field of Mongolian questions classification. From the experiment, it can be seen that the introduction of pre training morphemes features has a good effect. In the future, feature engineering can be further reduced by using pretraining language models.

At the same time, the research of Mongolian question intention recognition provides a good foundation for Mongolian question answering system in the future.

Acknowledgements

The project (Nos. 2018YFE0122900, CGZH2018125, 2019GG372, 2020GG0046) are supported by Inner Mongolia Science & Technology Plan; National Natural Science Foundation of China (Nos. 61773224); Natural Science Foundation of Inner Mongolia (Nos. 2018MS06006, 2020BS06001). Weihua Wang is the corresponding author.

References

Issa M. Alsmadi and Keng Hoon Gan. 2019. Review of short-text classification. *Int. J. Web Inf. Syst.*, 15(2):155–182.

Nello Cristianini and John Shawe-Taylor. 2010. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press.

Xiaodong He and David Golub. 2016. Character-level question answering with attention. In Jian Su, Xavier Carreras, and Kevin Duh, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1598–1607. The Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1746–1751. ACL.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 66–75. Association for Computational Linguistics.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Recurrent neural network for text classification with multi-task learning. In Subbarao Kambhampati, editor, *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 2873–2879. IJCAI/AAAI Press.

Min Lu, Feilong Bao, Guanglai Gao, Weihua Wang, and Hui Zhang. 2019. An automatic spelling correction method for classical mongolian. In *International Conference on Knowledge Science, Engineering and Management*, pages 201–214. Springer.

Andrew McCallum, Kamal Nigam, et al. 1998. A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, pages 41–48. Citeseer.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In Yoshua Bengio and Yann LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Dan I. Moldovan, Marius Pasca, Sanda M. Harabagiu, and Mihai Surdeanu. 2003. Performance issues and error analysis in an open-domain question answering system. *ACM Trans. Inf. Syst.*, 21(2):133–154.

Sardar Parhat, Mijit Ablimit, and Askar Hamdulla. 2019. Uyghur short-text classification based on reliable subword morphology. *IJRIS*, 11(3):250–255.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 379–389. The Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

Xiang Wang, Ruhua Chen, Yan Jia, and Bin Zhou. 2013. Short text classification using wikipedia concept based document representation. In *2013 International Conference on Information Technology and Applications*, pages 471–474. IEEE.

Weihua Wang, Feilong Bao, and Guanglai Gao. 2019a. Learning morpheme representation for mongolian named entity recognition. *Neural Processing Letters*, 50(3):2647–2664.

Weihua Wang, Rashel Fam, Feilong Bao, Yves Lepage, and Guanglai Gao. 2019b. Neural morphological segmentation model for mongolian. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE.

Lizhong Xiao, Guangzhong Wang, and Yang Zuo. 2018. Research on patent text classification based on word2vec and lstm. In *2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, volume 1, pages 71–74. IEEE.

Yiming Yang and Xin Liu. 1999. A re-examination of text categorization methods. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 42–49.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7370–7377.

Jianing Zhou, Jingkang Wang, and Gongshen Liu. 2019. Multiple character embeddings for chinese word segmentation. In Fernando Emilio Alva-Manchego, Eunsol Choi, and Daniel Khashabi, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28 - August 2, 2019, Volume 2: Student Research Workshop*, pages 210–216. Association for Computational Linguistics.

The Annotation Scheme of English-Chinese Clause Alignment Corpus

Shili Ge
Laboratory of Language and Artificial Intelligence, Guangdong University of Foreign Studies, Guangzhou, China 510420
geshili@gdufs.edu.cn

Xiaoping Lin
Center for Linguistics and Applied Linguistics, Guangdong University of Foreign Studies, Guangzhou, China 510420
lxpteresa@126.com

Rou Song ✉
Laboratory of Language and Artificial Intelligence, Guangdong University of Foreign Studies, Guangzhou, China 510420
College of Information Science, Beijing Language and Culture University, Beijing 100083
songrou@126.com

Abstract

A clause complex consists of clauses, which are connected by component sharing relations and logic-semantic relations. Hence, clause-complex level structural transformations in translation are concerned with the expression adjustment of these two types of relations. In this paper, a formal scheme for tagging structural transformations in English-Chinese translation is designed. The annotation scheme include 3 steps operated on two grammatical levels: parsing an English clause complex into constructs and assembling construct translations on the clause complex level; translating constructs independently on the clause level. The assembling step involves 2 operations: performing operation functions and inserting Chinese words. The corpus annotation shows that it is feasible to divide structural transformations in English-Chinese translation into 2 levels. The corpus, which unfolds formally the operations of clause-complex level structural transformations, would help to improve the end-to-end translation of complicated sentences.

1 Introduction

The grammatical levels of a natural language include morpheme, word, group/phrase, clause, and clause complex. Units of a higher level are made up of units of a lower level. Therefore, the central task for machine translation is language transformations on each grammatical level between languages. So far, there have been many studies on group/phrasal- and clausal-level structures and structural transformations. However, clause-complex level (CC-level) structures and structural transformations are far less discussed.

Halliday and Matthiessen (2004) describes the structures of English clause complex based on the theory of Systemic-Functional Grammar. Wang (2012) carries out an in-depth study on the structures of Chinese complex sentence in comparison with English. Luo (1992) points out that clauses should be considered as the translation unit in English-Chinese translation. These studies are enlightening, but they are limited to theoretical illustrations and discussions. Song and Ge (2015) study clause complex for language engineering. They put forward and demonstrate the PTA (Parsing-Translating-Assembling) model for English-Chinese translation on the CC-level, which is only a tentative idea and has not been tested through corpus annotation. Ge and Song (2020) clarify the concept of Component Sharing, define clause and clause complex based on this concept, and propose the design of the annotation scheme and specification for English-Chinese Clause Alignment Corpus (ECCA Corpus). Yet, the details of the annotation scheme and specification of the ECCA Corpus still need further study and exploration, especially on the structural transformations between English and Chinese clause complexes and their annotation.

A clause complex consists of clauses, but many clauses are not connected linearly because there are shared components between them. In order to present the alignment of English and Chinese clauses, it is necessary to show how English and Chinese clauses correspond under various component sharing mechanisms. In ECCA Corpus, the correspondence relationship between English and Chinese clauses is

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1035-1044, Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

shown through the annotation process of CC-level structural transformations, including construct analyzing, construct translating, and construct and component translations assembling. The work of this paper completes the annotation scheme, including defining the operation unit of CC-level structural transformations, i.e. constructs, specifying the content of each annotation step, formalizing assembling operations, and summarizing the operation functions used and the Chinese words inserted.

It is believed that ECCA Corpus is significant for theoretical linguistics and cognitive linguistics by providing samples for comparing CC-level structures and studying structural transformations between English and Chinese. Meanwhile, the corpus is believed to be significant in application. Although machine translation has been greatly improved with data-driven approaches, it still fails to produce satisfying results when it comes across long sentences with complicated structures. This corpus explores the feasibility of and practical ways for mechanical transformations on the CC-level. It is hoped that the knowledge of CC-level structural transformations may help to improve the performance of machine translation in dealing with complicated sentences.

The remainder of this paper is organized as follows: Section 2 introduces the objective of annotation, Section 3 introduces the annotation scheme, Section 4 and 5 present operation functions and inserted Chinese words applied in annotation; Section 6 provides relevant statistical results, and Section 7 concludes the paper.

2 Clause-Complex Level Structural Transformation

The ECCA Corpus is designed to annotate CC-level structural transformations between English and Chinese. In most linguistic theories, a clause complex is generally regarded as a group of clauses combined together based on logic-semantic relations. This being the case, CC-level structural transformations during translation should involve only reordering of clauses, which are usually organized in different logical ways between languages. However, there is another important transformation that should be noticed, i.e. the transformation of naming-telling structural relations.

> **Example 1**: There are fewer than 100 potential customers for supercomputers priced between $15 million and $30 million – presumably the Cray-3 price range.
>
> **Chinese Translation**: 价格 在1500万美元至3000万美元之间 的 超级计算机 的潜在客户不到100家，这个区间是克雷 3 号机大概的价格范围。
>
> **Machine Translation**: 价格 在1500万美元到3000万美元之间 的 超级计算机 的潜在客户不到100家——大概是Cray-3的价格区间。

In Example 1, the English clause complex contains a "modified component & modifying component" structure and a "described component & describing component" structure. As stated in Fang et al. (2016), the modifying and describing components are tellings, while the modified and described ones are namings. The two namings are highlighted in grey. The modifying telling, which closely follows its modified naming, "supercomputers", is marked with a single underline. The describing telling, which closely follows its described naming, "between $15 million and $30 million", is marked with a wave underline. It can be seen that the described naming is embedded inside the previous modifying telling. In the Chinese translation, the translation of the modifying telling "priced between $15 million and $30 million" is reordered and placed before the translation of its modified naming, "supercomputers". Thus, the translation of the describing telling, "– presumably the Cray-3 price range", could not share its described naming as it does in the English text. To deal with the problem, the described naming is reproduced in the Chinese translation as a generalized form "这个区间" and combined with the translation of its describing telling into a new clause. However, the machine translation does not reproduce the described naming and thus fails to translate the "described component & describing component" structure correctly. This example shows that the adjustment of naming-telling relationship is no less important than logic-semantic relationship adjustment in CC-level structural transformations.

Previous corpus studies prove that naming-telling structures are prevalent in both Chinese and English clause complexes. Although the two languages share the same types of naming-telling structures, they have different distributions of the structure types (Ge and Song, 2016). As a result, naming-telling structure adjustment is often necessary in English-Chinese translation. Meanwhile, the two languages arrange clauses in different logical ways, which leads to the other kind of structural transformations.

To sum up, the annotations of CC-level structural transformations are to demonstrate the adjustment of naming-telling structures and logical expressions in English-Chinese translation.

3 Design of the Annotation Scheme

The CC-level structural transformations of Example 1 are illustrated in Figure 1.

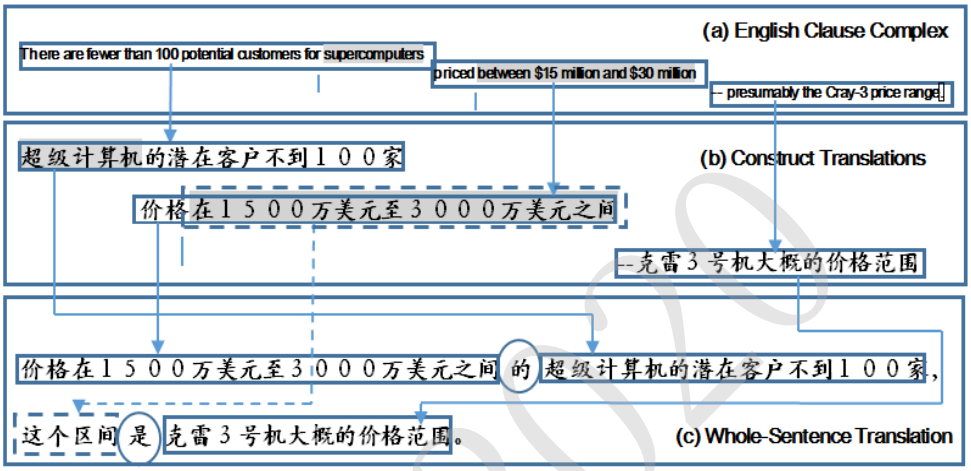


Figure 1. CC-Level Structural Transformation of Example 1

In Figure 1-(a), the English clause complex is firstly segmented into three constructs based on naming-telling structural analysis. The grey parts are namings, whose left-boundaries are marked by the symbol "|" below the line. Tellings modifying or describing these namings take up new lines and are indented to the right after their namings. This way of demonstrating the naming-telling relationship is called newline-indent schema.

Each line in Figure 1-(a) is considered as one construct for making up the English clause complex, and they are translated independently in Figure 1-(b). Each line of translations in Figure 1-(b) is called a construct translation. Contruct translations are also displayed in the newline-indent schema, with the translations of tellings indented to the right side after the translations of their namings. The arrows between Figure 1-(a) and 1-(b) start from each English construct and point to their Chinese counterparts. Figure 1-(c) shows the whole-sentence translation. The solid line arrows between Figure 1-(b) and 1-(c) start from each construct translation, and point to their new positions in the whole-sentence translation. The dash line arrow starts from the translation of a naming and points to its generalized form. The circles in Figure 1-(c) mark the insertion of the particle "的" and the linking verb "是".

The graphic demonstration in Figure 1 clearly displays how the English clause complex is transformed step by step into a Chinese one. However, the demonstration is quite complicated, hard to be annotated and not convenient for statistical analysis. Hence, a more formal annotation scheme for annotating structural transformations is designed.

The formal annotation scheme follows the 3 steps in the graphic demonstration: (1) segment English clause complexes into constructs and display them in newline-indent schema; (2) translate independently each construct into Chinese; (3) rearrange construct translations for a whole-sentence translation.

The structural transformations are to be annotated at the end of each line of the whole-sentence translation. The parts that make up the whole-sentence translation are encoded as numbers, and the operations implemented on these parts are tagged as operation functions. In this way, structural

transformations could be annotated formally. The following is a detailed illustration of the designs.

Whole-Sentence Translation of Example 1:
价格在1500万美元至3000万美元之间的超级计算机的潜在客户不到100家，//2+的+1
这个区间是克雷3号机大概的价格范围。//sum(2.2)+是+delt(3)

As shown above, structural transformations are tagged after the symbol "//" at the end of each line. The numbers represent the parts making up the whole-sentence translation. For example, the number 2 of "2+的+1" represent the second line of construct translations, namely "价格在1500万美元至3000万美元之间". The number 2.2 of "sum(2.2)+是+delt(3)" represent the second section of the second line of construct translations, namely "在1500万美元至3000万美元之间". In the annotation scheme, the translations of namings are usually processed as a single unit. When the translation of a naming is positioned within a construct translation, the construct translation is segmented by the translation of this naming into several parts, which includes the naming translation, the parts before and/or after the naming translation. These segments are named as component translations. The component translations on the n^{th} line are encoded from left to right as n.1, n.2, and n.3 etc. In this example, the second line of construct translation contains the translation of a naming at its end, and thus it is divided into two components. The component before the naming translation is encoded as 2.1, while the naming translation is encoded as 2.2. From this example, it can be seen that the parts making up a whole-sentence translation include construct translations and component translations. These two types of constituents in translations are the basic units to be dealt with by operation functions, and thus they are called operation units in this paper.

As for operation functions, they are used to mark the operations implemented on operation units. The symbol "+" means linking two operation units. The function "sum(2.2)" means turning the encoded component 2.2, namely "在1500万美元至3000万美元之间", into a more generalized expression "这个区间". The function "delt(3)" means deleting the dash in the translation of the encoded construct 3, namely "–克雷3号机大概的价格范围". The designing of operation functions will be discussed in detail in section 4.

Additionally, it is noted that the translation of every construct in the second step is independent of its context. Certainly, the disambiguation of a certain word still need reference to its context, but it is not allowed to add extra words, delete words or change the structures based on the context.

4 Operation Functions

There are two types of operations for CC-level structural transformations: (1) processing and assembling the operation units, and (2) inserting Chinese words. The first type of operation is annotated as operation functions, which will be discussed in this section. The second type of operation will be discussed in Section 5.

Operation functions are written in the format of FunctionName(x) or FunctionName(x,y), in which FunctionName specifies the operation to be implemented, while x and y specify the objects to be processed, which are all called operation units.

Twenty operation functions are designed, which involve 6 types of operations: link, reorder, add, delete, rewrite, and substitute. The 20 operation functions are listed in Table 1.

Operation Types	Operation Functions
Link	concatenate(x,y) (i.e. x+y)
Reorder	demonstrated with the codes of operation units
Add	corcj(x), corcj2(x) , prd(x)
Delete	ignore(x) (i.e. *x), delcj(x) , delcj2(x) , delpn(x) , delt(x)
Rewrite	det(x) , ndet(x) , sum(x) , pron(x) , rel(x) , paren(x) , n2v(x)
Substitute	rpw(x,y) , r2n(x,y) , n2r(x,y)

Table 1. Operation Functions

Of all these functions, link and reorder are common operations in almost all processed whole-sentence translations. The usage of these two functions is shown above in Example 1. Other functions are divided into two types based on their adjustments to clause complex structures. Some of the two classes are discussed with examples in the following subsections. Due to limited space, the functions not discussed in this paper can be referred to in Song et al. (2020).

4.1 Operation Functions for Transforming Naming-Telling Structures

Due to different distributions of naming-telling structural types, it is often necessary to transform naming-telling structures during English-Chinese translation. Generally, there are 3 ways to rearrange English tellings in Chinese translations: (1) inserting the telling translation as a modifier on the left of its naming translation, (2) keeping the telling translation as a statement or a description on the right of its naming translation, (3) reproducing the naming and rendering it another way before linking it with the telling translation. Of these 3 ways, the previous two requires only the link and reorder operations. When it comes to the third way, extra processing is needed, namely to reproduce the naming and render it in certain forms. This is because in a clause complex, a naming, if referred to more than once, should take different forms for its respective occurrence. To be more specific, a naming usually appear at first in its full name or its indefinite form, and then appear in its definite form, as a pronoun, or as a more generalized form. The operation functions det(x), ndet(x), pron(x) and sum(x) are specially designed for rewriting a naming. Table 2 presents the definitions of operation functions used to transform naming-telling structures.

Operation Types	Operation Functions	Definition
Rewrite	det(x)	change x into its definite form
Rewrite	ndet(x)	change x into its indefinite form
Rewrite	pron(x)	change x into a corresponding pronoun
Rewrite	sum(x)	change x into a more generalized term
Rewrite	rel(x)	concretize x based on the current context
Delete	ignore(x)	delete the relative pronoun/adverb in x
Substitute	rpw(x,y)	replace the relative pronoun/adverb in x with y

Table 2. Operation Functions for Transforming Naming-telling Structures

The usage of sum(x) has been illustrated in Example 1. The usage of ignore(x) and rpw(x,y) will be discussed in the following.

Since attributive clauses do not have clear semantic meanings by themselves, they need special treatment in annotation. In an attributive clause, the relative pronoun is only a formal substitute for the antecedent, and it is meaningless by itself. As a result, attributive clauses cannot be translated independent of context theoretically. To handle the problem, it is specified that relative pronouns in capitalized forms should be used to occupy the positions where the translations of antecedents should have been in construct translations.

In most cases, capitalized relative pronouns occupy the positions of a subject at the beginning of construct translations. Hence, the ignore(x) function is used to delete the capitalized relative pronouns before construct translations are linked with the translations of their namings.

Sometimes, capitalized relative pronouns occupy positions in the middle of construct translations. In this case, the function rpw(x,y) should be used to replace relative pronouns with the translations of their antecedents. Such substitutions are operable since capitalized relative pronouns are identifiable with their special forms. Example 2 shows the usage of this function.

Example 2: The Company has proposed an internal reorganization plan in Chapter 11 bankruptcy proceedings, under which it would remain an independent company.

(1) Newline-Indent Schema of English Clause Complex:
The Company has proposed an internal reorganization plan in Chapter 11 bankruptcy proceedings,
| under which it would . . . company.

(2) Construct Translations:
该公司已在第 1 1 章破产程序中提出了一个内部重组计划，
| 根据WHICH 它将仍为一个独立公司。

(3) Whole-Sentence Translation:
该公司已在第 1 1 章破产程序提出了一个内部重组计划，//1
根据该计划它将仍为一个独立公司。//rpw(2,sum(1.2))

The second line in Example 2-(1) is an attributive clause, with "an internal reorganization plan" as its antecedent. In this example, the antecedent is a naming while the attributive clause is its telling. In Example 2-(3), the result of "sum(1.2)" is a generalized term for "一个内部重组计划", namely "该计划"(this plan). The function "rpw(2,sum(1.2))" means replacing "WHICH" in the second line of construct translations with "该计划".

4.2 Operation Functions for Transforming Logical Expressions

English and Chinese clause complexes differ in logical expressions in the following 3 aspects: (1) clausal order, (2) the use of logical conjunctions, and (3) naming sharing of logically-related clauses. These differences may give rise to different translation problems, and thus different functions are designed to deal with them.

Operation Types	Operation Function	Definition
Substitute	r2n(x,y)	replace the pronoun in x with the corresponding noun in y
Substitute	n2r(x,y)	replace the noun in x with the corresponding pronoun in y
Add	corcj(x)	add the matched conjunction for the first one in x
Add	corcj2(x)	add the matched conjunction for the second one in x
Delete	delcj(x)	delete the first conjunction in x
Delete	delcj2(x)	delete the second conjunction in x
Delete	delpn(x)	delete the relevant pronoun in x

Table 3. Operation Functions for Transforming Logical Expressions

Firstly, English and Chinese clause complexes have different clausal orders. The differences lie in two aspects: (1) In English, main clauses are usually placed before subordinate clauses, while it is the opposite in Chinese. (2) In English, quotation verbs are placed after or between quotations, while in Chinese, quotation verbs are usually placed before quotations. In the annotation scheme, the operation of reorder is demonstrated by the line numbers referring to clause translations. Sometimes, the reorder of clauses is accompanied with the necessity of changing referential order. The two functions r2n(x,y) and n2r(x,y) are specially designed for dealing with this situation.

Example 3: Yields may blip up again before they blip down because of recent rises in short-term interest rates.

(1) Newline-Indent Schema of English Clause Complex:
Yields may blip up again
before they blip down
because of recent rises in short-term interest.

(2) Construct Translations:
收益率可能会再次上升
它们在下降之前
因为最近短期利率上升。

(3) Whole-Sentence Translation:
因为最近短期利率上升，//3
收益率在下降之前，//r2n(2,1)
它们可能会再次上升。//n2r(1,2)

In Example 3, the English clausal orders should be adjusted in the Chinese translation. The rearrangement of clausal orders is displayed in Figure 2.

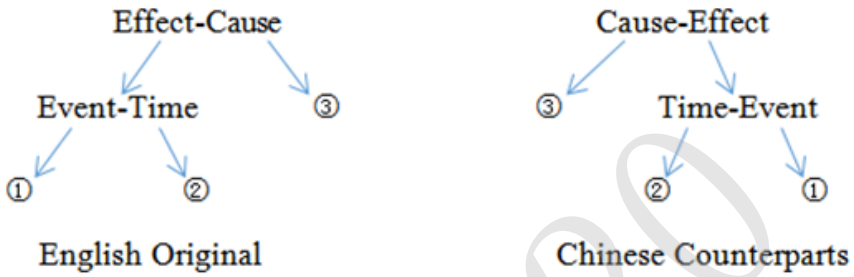


Figure 2. Logical Orders of Clauses in Example 3 and Its Chinese Counterparts

The exchange of clausal orders is demonstrated with the exchange of orders of line numbers. As the first line shown in Example 3-(3), the number "3" at its end means that this line comes from the third line of construct translations. Meanwhile, the interclausal order between the first and second line of construct translations has also been changed in Example 3-(3). The first line of construct translations with the noun "收益率" is placed after the second line with the pronoun "它们". However, in general terms, the line with a pronoun is supposed to appear after the line with the noun it refers to. Hence, it is necessary to exchange the noun and pronoun concerned in the two lines. The function r2n(2,1) means replacing the pronoun "它们" in second line of construct translations with the corresponding noun "收益率" in the first line. The function n2r(1,2) means replacing the noun "收益率" in first line of construct translations with the corresponding noun "它们" in the second line.

However, the whole-sentence translation above is not optimal. A better whole-sentence translation is shown as the following.

因为最近短期利率上升，//3
所以收益率在下降之前可能会再次上升。//corcj(3)+r2n(2,1)+delpn(n2r(1,2))

In this new whole-sentence translation, line 2 and line 3 in the original whole-sentence translation are combined into one by deleting the pronoun "它们". The deletion of the pronoun is tagged as delpn(n2r(1,2)), which means deleting the pronoun in the result of n2r(1,2). With the operation of this function, the result of n2r(1,2), namely "它们可能会再次上升", is turned into "可能会再次上升". Meanwhile, the conjunction "所以" is added, matching that of the third line of construct translations. This addition of a conjunction is tagged as corcj(3).

Example 3 shows relevant functions for dealing with English-Chinese differences on clausal orders and on the use of logical conjunctions.

5 Inserted Chinese Words

The inserted Chinese words are function words such as linking verbs, particles, conjunctions and prepositions. The Chinese words can be classified into 2 types based on their functions: (1) words indicative

of argumentative relations between operation units, and (2) words indicative of logical relations between operation units.

Words Indicative of Argumentative Relations Between Operation Units	
的	connect a modifier with its modified noun
是	connect a naming with its descriptive noun phrase
即	connect a naming with its parallel noun phrase
有	introduce a naming, functioning like “there be” in English
在	connect a naming with its telling as a location
将	inserted before a naming whose telling is about the processing or recognition of the naming
在…中	a frame in which a naming used as a location is embedded
Words Indicative of Logical Relations Between Operation Units	
使	logical component inserted before a telling which is the logical result of a behavior or an event
而	logical component inserted between two verb phrases which are in the cause-effect relations
就	logical component inserted before a telling which is the logical result of a behavior or an event
并	logical component suggesting a further movement

Table 4. Inserted Chinese Words

Of words indicative of argumentative relations between operation units, the three most frequently used Chinese words are “的”, “是” and “即”. “的” is inserted between an attribute and its modified noun phrase. “是” and “即” are often inserted between a naming and its telling which is usually an appositive in English clause complexes. “是” is applied when the telling is describing the property of its naming, while “即” is applied when the telling has the same reference as its naming. The usage of “的” and “是” has been shown in Example 1. Example 4 is to show the usage of the logical conjunction “并”.

Example 4: Mrs. Yeargin concedes that she went over the questions in the earlier class, adding: “I wanted to help all” students.

(1) Newline-Indent Schema of English Clause Complex:
Mrs. Yeargin concedes
that she went over the questions in the earlier class,
adding:
“I wanted to help all” students.

(2) Construct Translations:
叶尔金太太承认
她在上一届的班里复习了那些问题，
补充说：
“我想帮助所有”学生。

(3) Whole-Sentence Translation:
叶尔金太太承认她在上一届的班里复习了那些问题，//1+2
并补充说：//并+3
“我想帮助所有学生。”//4

There are two clauses in this example. One clause is constituted by lines 1 and 2 in Example 4-(1), and the other is constituted by the naming in line 1, i.e. “Mrs. Yeargin”, and the telling, lines 3 and 4. Semantically speaking, the second clause is the continuation of the first one, involving the action to be taken after that of the first clause. In the English clause complex, the logical relation is presented by using an infinite verb for the action in the second clause, namely “adding”, to lower the grammatical hierarchical level of the clause. However, in Chinese, there is no such grammatical device as changing verb forms. Therefore, the logical conjunction “并” is added for connecting the two clauses logically.

6 Statistical Data

So far, we have annotated 2108 clause complexes on 136 documents from English Penn Treebank. Of the annotated clause complexes, 336 contain only one clause. Of the clause complexes containing more than one clause, 532 do not involved CC-level structural transformations. Therefore, only a total of 1240 clause complexes are annotated with relevant functions and Chinese words, accounting for 58.82% of the 2108 clause complexes.

Function	*x	pron(x)	sum(x)	det(x)	delt(x)	delpn(x)
Freq.	361	136	103	56	32	23
Function	rpw(x,y)	corcj(x)	r2n(x,y)	paren(x)	n2r(x,y)	delcj(x)
Freq.	20	20	16	14	11	9
Function	n2v(x)	prd(x)	rel(x)	ndet(x)	corcj2(x)	delcj2(x)
Freq.	8	6	6	4	1	1

Table 5. Frequency of Operation Functions in ECCA Corpus

The frequency of each operation function in ECCA Corpus is shown in Table 5. The number of each inserted Chinese word is also counted. The most frequently used words, “的”, “是” and “即”, appear for 486, 112 and 22 times, respectively. Other inserted Chinese words are used for less than 5 times.

7 Conclusions and Discussions

Component sharing relations and logic-semantic relations are organized differently in English and Chinese clause complexes. As a result, during English-Chinese translation, it is necessary to adjust the expressions of these two relations with some structural transformations on the clause complex level. This paper divides English-Chinese clause complex translation into two grammatical levels. On the clause complex level, an English clause complex is parsed into constructs, and the translations of these constructs are assembled into a whole-sentence translation. On the clause level, each construct is translated independently. The two-level translation mechanism, including operation functions and inserted Chinese words used in the assembling step, has been designed formally and proved feasible with corpus manual annotation.

By designing the two-level translation mechanism, this paper follows a common strategy for AI problem solving, namely to decompose a complicated task into sequential simple tasks. It is believed that this mechanism could reduce the demanded data scale and calculation complexity for machine-learning-based machine translation, since the task of translating a sentence is decomposed into simple tasks of translating and assembling shorter constructs. Meanwhile, although the mechanism cannot produce perfect results in some cases, it is an explainable translation process and thus is worth further exploring.

The work present in this paper is only initial. In the future, efforts will be made to enlarge the corpus size, improve the quality of annotated translations, provide multiple translation alternatives, design algorithms for realizing operation functions and discover linguistic knowledge based on the ECCA Corpus.

Acknowledgements

This research is supported by National Natural Science Foundation of China (61672175).

References

Fang, F., Ge, S., & Song, R. (2016). Error analysis of English-Chinese machine translation. In Sun, M., Huang, W., Lin, H., Liu, Z., & Liu, Y. (eds.), *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data* (pp. 35-49). Gewerbestrasse: Spring.

Ge, S., & Song, R. (2016). The naming sharing structure and its cognitive meaning in Chinese and English. In Xiong, D., Duh, K., Agirre, E., Aranberri, N., & Wang, H. (eds.), *Proceedings of the 2nd Workshop on Semantics-Driven Machine Translation (SedMT 2016)* (pp. 13-21). Stroudsburg: Association for Computational Linguistics (ACL).

Ge, S., & Song, R. (2020). English-Chinese clause alignment corpus tagging system based on corpus annotation. *Journal of Chinese Information Processing, 34*(6), 27-35.

Halliday, M. A., & Matthiessen, C. M. I. M. (2004). *An introduction to functional grammar third edition*. London: Edward Arnold.

Luo, X. (1992). Unit of transfer in translation. *Foreign Language Teaching and Research, 4*, 32-37.

Song, R., & Ge, S. (2015). English-Chinese translation unit and translation model for discourse-based machine translation. *Journal of Chinese Information Processing, 29*(5), 125-136.

Song, R., Ge, S., Chen, X., & Lin, X. (2020). English-Chinese clause alignment corpus annotation guidelines. *Technical Report of Collaborative Innovation Center for Language Research & Service of Guangdong University of Foreign Studies*. Guangzhou.

Wang, L. (2012). *The complete works of Wang Li volum 8: Chinese grammar theory*. Beijing: Zhonghua Book Company.

Categorizing Offensive Language in Social Networks: A Chinese Corpus, Systems and an Explanation Tool

Xiangru Tang, Xianjun Shen*, Yujie Wang, Yujuan Yang
School of Computer, Central China Normal University, China
National Language Resources Monitoring & Research Center for Network Media, China
Hubei Provincial Key Laboratory of Artificial Intelligence and Smart Learning, China
xjshen@mail.ccnu.edu

Abstract

Recently, more and more data have been generated in the online world, filled with offensive language such as threats, swear words or straightforward insults. It is disgraceful for a progressive society, and then the question arises on how language resources and technologies can cope with this challenge. However, previous work only analyzes the problem as a whole but fails to detect particular types of offensive content in a more fine-grained way, mainly because of the lack of annotated data. In this work, we present a densely annotated data-set COLA (**C**ategorizing **O**ffensive **LA**nguage), consists of fine-grained insulting language, antisocial language and illegal language. We study different strategies for automatically identifying offensive language on COLA data. Further, we design a capsule system with hierarchical attention to aggregate and fully utilize information, which obtains a state-of-the-art result. Results from experiments prove that our hierarchical attention capsule network (HACN) performs significantly better than existing methods in offensive classification with the precision of 94.37% and recall of 95.28%. We also explain what our model has learned with an explanation tool called Integrated Gradients. Meanwhile, our system's processing speed can handle each sentence in 10msec, suggesting the potential for efficient deployment in real situations.

1 Introduction

In modern society, the occupation of offensive language on the online world, such as social media, is becoming a paramount concern. Offensive language differs considerably, ranging from pure abuse to more rigorous types of writing. Thus, offensive language is hard to be automatically identified. However, it's essential to track this; for example, the appearance of offensive language on social media is related to hate crimes in a real social situation. [Müller and Schwarz2018]. Moreover, it can be pretty troublesome to distinguish fine-grained offensive language because few general definitions exist [Davidson et al.2017].

Recently, researchers have proposed some guidelines to identify the type and the attributes of offensive language [Zampieri et al.2019a]. However, the online world's offensive language is a general category containing specific examples of profanity or insult. In our work, "Offensive language" in the online world is defined in more detail and fine-grained. And to the best of our knowledge, though offensive language identification being a burgeoning field, there is no data-set yet for Chinese.

"Offensive" is pretty much something people identify as against morals, very inappropriate, or disrespectful. However, "offensive" is a broad general term and does not define the precise extent or the limits of its application. Thus, we classify the term "offensive" into three categories: "insulting," "antisocial" and "illegal" through stepwise refinement. "Insulting" is something rude, insensitive and/or offensive, directed at another person or group of people. This emphasizes that the content is a direct attack against specific others. "Antisocial" is harmful to organized society, or the language describes a behavior deviating from the social norm for long. "Illegal" language means it violates the language policy. Where the

* Corresponding author.

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1045-1056, Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

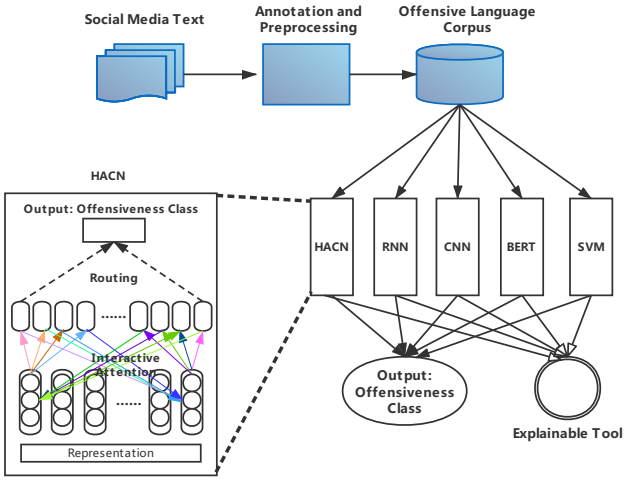


Figure 1: Data Processing Pipeline System and The architecture of Hierarchical Attention Capsule Network (HACN).

language policy refers to the government through legislation or policies to formally decide how languages are used. However, the language policies of each country are not completely consistent.

Thus, two questions arise a) how LRs can cope with the large numbers of offensive language in the online world, and b) can LT provide means to process and respond promptly to such language data streamed in a huge amount at high speed? Firstly, there is no existing data-set for the Chinese language to provide for correctives of hate speeches, cyberbullying, or fake news. Then, current methods can not produce highly precise results for detecting offensive content and behaviors. Also, They used inflexible proprietary APIs, which is hard to reproduce. On the other hand, there is a real need for methods to detect and deal with online words quickly because of the enormous amount of data created every day.

In this context, we present a sizeable Chinese classification corpus of offensive language called COLA. Then, we employ a deep dilated capsule network to extract hierarchical structure. We further design hierarchical attention to aggregate and fully utilize information within a hierarchical representation. Correctly, each sentence is embedded into capsules and incrementally distilled into task-relevant categories during the hierarchical attention process. What is more, we present an explanation tool, which proves that our work for the Chinese language seizes the pattern of offensive language in some points, and almost correctly identifies different varieties of offensive language, like hate speech and cyberbullying.

In summary, our work aims at answering the two questions a) how LRs can cope with a huge amount of offensive language in the online world, and b) can LT provide means to process such language data at high speed? The major contributions are highlighted as follows:

- We describe COLA, the first Chinese offensive language classification dataset. COLA is designed to study how language resources and technologies can cope with this offensive language challenge in the online world. It is now publicly available.
- We propose a hierarchical attention capsule network(HACN), where the hierarchical attention mechanism is introduced to model the hierarchical structure. It is inspired by capsule, with modifications to handle the words explicitly.
- We show that our HACN model surpasses state-of-the-art methods for classification on COLA. Furthermore, our presented explanation tool clearly explains what our model has learned.

2 Related work

2.1 Corpus

Some previous works have discussed how to identify the offensive language, but in that literature, the offensive language is ranging from aggression to cyberbullying, toxic comments, and hate speech. In the

following, we explain each of these open public challenges briefly.

SemEval-2019 Task: In Task 6 of SemEval 2019, they propose three separate sub-tasks. A subtask is Offensive Language Selection, the other is Categorization of Offensive Language, and the last is Offensive Language Target Recognition. SemEval-2019 Task 6 is called OffensEval, and the collection methods of their data are explained in [Zampieri et al.2019b]. Additionally, it collected more than 14100 posts of sentences.

Aggression identification (TRAC): The TRAC study [Kumar et al.2018] provided players with a data set containing a training set and a validation set.They are composed of 15,000 Facebook posts and comments annotated in English and Hindi. For the test set, two different sets are used, one from Facebook and the other from Twitter. It aims at distinguishing three types of data: non-aggressive, covert aggressive, and over-aggressive.

Hate speech recognition: In [Kwok and Wang2013, Burnap and Williams2015, Djuric et al.2015], they present a Abusive language selection task. Specially, [Davidson et al.2017] provided the hate speech recognition data set, which contains more than 24000 English tweets marked as non-offensive, hate speech, and profanity.

Offensive language: The data-set provided by GermEval [Wiegand et al.2018] focused on offensive language recognition in German tweets. The study showed a data set of more than 8,500 tagged tweets. This data set is used to perform binary classification task of distinguishing between offensive and non-offensive information. Besides,the second task divided offending tweets into three categories: profanity, insult, and abuse. While similar to our work, there are three important differences: (i) we have a third level in our hierarchy, (ii) we use different labels in the second level, and (iii) we focus on Chinese.

Toxic comments (Kaggle): Kaggle holds a Toxic Comment Classification Challenge as an open dataset. The dataset in this competition was extracted from the comments of Wikipedia, and it was formed in six categories: toxicant, severe toxic,identity hate, threat, insult,obscene . Moreover, the data set is also employed outside of the competition [Georgakopoulos et al.2018], treated as an external training resource for the TRAC, as mentioned above [Fortuna et al.2018].

However, each of these tasks tackles a particular challenge of detecting offensive language. Thus, we present a new dataset, hoping it could become a valuable resource for improving offensive language categorizing.

2.2 Classification

Traditional classification methods are designed with rich features and syntactic structures to achieve the classification task [Jiang et al.2011]. But, these feature-based methods are labor-intensive, and the performance depends largely on the quality of the features. Recently, deep learning methods are becoming popular for aspect-level sentiment classification. Recurrent Neural Networks (RNNs) are the most commonly used technique for this task [Tang et al.2015]. The attention mechanism is further introduced to model the target-context association [Wang et al.2016]. Recently, CNN-based models have shown the strengths inefficiency to tackle the aspect-level sentiment classification [Xue and Li2018, Huang and Carley2019, Li et al.2018]. However, all the previous methods utilize static pooling operation or attention mechanism to locate the keywords, which fails to handle the overlapped features. We introduce vector-based feature representation and feature clustering to address this.

Capsule network was proposed to improve the representational limitations of CNN and RNN by extracting features in the form of vectors. The technique was firstly proposed in [Hinton et al.2011]. But is mainly devised for the image processing domain. Introducing capsules allows us to utilize a routing mechanism instead of pooling operation to generate high-level features, which is a more efficient way for features encoding. The routing module is able to cluster features in an iterative way, which achieved impressive performance recognizing highly overlapped digits. Several types of capsule networks have been proposed for natural language processing. [Zhao et al.2018] investigated capsule networks for text classification. They also found that capsule networks exhibit significant improvement when transferring single-label to multi-label text classification. However, interactive word-level attention is not considered in these typical capsule routing methods.

3 Data Collection

In this section, we describe the data set and how we annotate data set.

3.1 Overview

Data	Train	Test	Valid	Total
Neutral	5357	1700	1546	8603
Insulting	5075	1660	1493	8228
Antisocia	841	303	218	1362
Illegal	327	96	91	514
Total	11600	3759	3348	18707

Table 1: Statistics of the four classes in COLA data. Number of sentences in train set, test set, valid set.

We create a large-scale data-set that annotates offensive texts in Chinese. The texts are crawled from Youtube and Weibo: 18.7k comments in total. Three annotators categorised these texts in four classes: neutral, insulting, antisocial, and illegal. We build a Chinese dataset from social media that people can communicate on Internet, such as Sina Weibo[0] and YouTube comments. Our released COLA contains user-generated comments from different social media platforms, and as we know, it is the first of its kind. And, the dataset is marked as capture different types of offensive language. We propose four automatic classification systems, each designed to work for the Chinese language.

3.2 Data Acquisition

With more than 1000 comments and more than 10000 views as the thresholds, we selected 20 popular Chinese videos from YouTube. Furthermore, from the comments below the video are crawled through Google YouTube V3 API, which is offered by Google for researchers to collect comments. And a total of 20000 comments were received. We store the 20000 comments, and then we clean the data. We first convert the traditional Chinese character in the data set to simplified Chinese characters, and then filter out the useless data with messy codes and HTML tags.

There are some technologies we employ to crawl the data. Firstly, we retrieve 81718 Chinese sentences from Weibo and YouTube reviews in JSON format, and contain information such as timestamp, URL, text, user, re-tweets, replies, full name, id, and likes. Extensive processing is carried out to remove all the noisy sentences. We apply the following pre-processing steps: the documents are tokenized using NLTK, the URLs and mentioned users are removed, and all letters are converted to lower-case. As a result, a dataset of 18,707 offensive language sentences is created. Nevertheless, social media companies all have some methods to prevent crawlers. These methods can be divided into three categories: analyzing the headers of web page requests, monitoring the behavior of users visiting the website, and adjusting the directory and data loading methods. Corresponding to that, we adopt three approaches to crawl the data. For the first one, we could directly add HEADERS and REFERER to the code to bypass the check. And the same IP visits the same page multiple times in a short period, or the same account performs the same operation multiple times in a short time may cause the second one situation. For this situation, we can use the IP proxy to resolve it. We can use a browser to analyze the requests for the last situation. If we can obtain the AJAX request, then we can use the above two methods to resolve and obtain the corresponding data. However, if we cannot get AJAX requests, we can call the selenium + phantomjs framework and call its browser kernel to simulate human operations and JS scripts that trigger the page.

3.3 Annotation

We construct the data-set which comes from the hot issues of YouTube comments and Weibo. And web crawler gets our the data we needed. The data is annotated by three volunteers. After analyzing all the data, more than 18,707 sentences are selected. Then, we remove invalid tokens in the text, like HTML

[0] https://en.wikipedia.org/wiki/Sina_Weibo

tags and emoticons, and treat the text as the preliminary data for hand-operated annotation. After that, the vocabulary was divided into three categories: insulting language, antisocial language, illegal language. Due to the special combination of sensitive words, the standard of language structure is pretty vague. We note that different people may have different understanding of the text during the process of annotation. It means the boundary of the same word may be different. Thus, three people are asked to annotate the same text to ensure accuracy. We should note that inter-annotator agreement and intra-annotator agreement have been considered for the coherence of annotations While annotating.

4 Proposed Methods

In this part, we describe the categorizing task to be performed, how we perform the task and the excellent methods are proposed for especially this task.

4.1 Task

The recognition and classification of offensive language in the online world can be realized as a multiple classification task. In this section, we describe several proposed neural networks in details. The aim of aspect-level sentiment classification is to predict the class y of a sentence. In our task, the $y \in \{Neutral, Insulting, Antisocial, Illegal\}$.

4.2 Baseline Systems

Several baseline models are evaluated in Table 2.

SVM: For training our SVM classifier, scikit-learn[1] machine learning in Python library is used for benchmarking. During our experiments, we carry out 10-fold cross validation. We select the Linear SVM formulation, known as C-SVC and the value of the C parameter is 1.0.

RNN: RNN is the high-efficiency method to solve classification problem in NLP tasks. In this paper, we adopt GRU, which has great superiority compared to LSTM and basic RNN. In the final Multi Layer Perceptron layer, 128 neurons are used for classification. And Sigmoid activation function is applied to the final layer.

CNN: We adopt word-level CNN model which has 1D convolution layer with 150 filters and kernel size 6, dropout 0.2, cross entropy loss funtion and four dense layers with ReLU, tanh, sigmoid and softmax activation respectively.

BERT: BERT has displayed its great advantage of text representation in many NLP tasks. We fine-tune the task-specific components, such as a softmax classifier with BERT or deem BERT model as a feature extractor. First of all, we pack the input features as $H_0 = e_1, \cdots, e_T$, where $e_t (t \in [1, T])$ is the group of the token embedding, position embedding and segment embedding corresponding to the input token x_t. Then the L transformer layers is introduced to refine the characteristics of the token layer by layer. Specifically, the representations $H^1 = h_1^l, \cdots, h_T^l$ at the l-th ($l \in [1, L]$) layer are calculated below:

$$H^l = \boldsymbol{Transformerl}(H^{l-1}) \tag{1}$$

We treat H^L as the contextualized representations of the input tokens.And use them to execute the downstream task's predictions.

4.3 Challenge

However, there are still several limitations of the current approaches for offensive language categorizing. Firstly, little attention has been paid to the imbalance of different classes, which are essential and challenging because in categorizing tasks, it will be hard to capture the critical pattern of a specific class without sufficient data. Since the COLA data-set is unbalanced, the neural network may not have enough training examples of "illegal language" to learn. Consequently, it cannot catch the feature and structure of the "illegal language".

[1] https://scikit-learn.org/stable/

Second, existing research on offensive language detection cannot accurately detect offensive content because one sentence expresses multiple polarities, resulting in overlapped feature representation. The highly over-lapped features will confuse the classifier seriously, and the three types of specific offensive language do not have quite a considerable distinction. However, most existing methods only keep the most potent feature by max-pooling operation or utilize attention mechanisms to find the keywords, which fails to distinguish the over-lapped features.

Third, the dissemination and use of online platforms have grown significantly in every minute. Thus, the speed of the system is what we aspire to enhance, especially in this task. The systems are required to detect quickly and deal with this type of content in a short time. In this way, we should consider both accuracy and speed as the evaluation metric in a real application.

4.4 Hierarchical Attention Capsule Network

An original capsule network is a group of neurons obtained from the output of the convolutional operation performed on word representation h_n^a and h_n^c. So, the output of the capsule is a vector representing different properties of the same objective. The routing method [Hinton et al.2011] is employed in our model, and except for the high-dimensional output M, there is one more activation probability in our capsule.

We have already decided on the outputs of all the capsules C_L in the first capsule layer, and we now want to decide which capsules C_{L+1} to active in the layer above and how to assign each active low-level capsule to one active higher-level capsule. The vector-based features get clustered in the high-level capsules where the outputs of high-level capsules play the role of Gaussians, and the output vectors of low-level capsules play the role of the data points. To establish a semantic relationship model between aspect terms and context, we further devise an interactive attention-based routing mechanism.

Firstly, every primary capsule i is transformed by W_{ij} to cast a vote $V_{ij} = M_i W_{ij}$ for the output of high-level capsule j. Moreover, we can get the mean μ_j of the votes from the input capsules, and the variance σ_j about that mean for each dimension h:

$$\mu_j^h = \frac{\sum_i R_{ij} V_{ij}}{\sum_i R_{ij}} \tag{2}$$

$$\sigma_j^{h2} = \frac{\sum_i R_{ij} (V_{ij}^h - \mu_j^h)^2}{\sum_i R_{ij}} \tag{3}$$

Then, we can calculate the activation probability of capsule j by:

$$c_j^h = \left(\beta_u + log(\sigma_j^h)\right) \sum_i R_{ij} \tag{4}$$

$$a_j = \boldsymbol{sigmoid}(\lambda(\beta_\alpha - \sum_h c_j^h)) \tag{5}$$

In there, μ_j^h is the $h^t h$ component of the capsule $j$âs vectorized output M_j, and β_uã β_α are trainable parameters.

Then, for the part of capsule routing procedure, we propose a hierarchical attention to capture the hierarchical representation. In particular, the scaled dot-product attention is used to map a set of key-value pairs and the query to a weight on the word-level token. The queries are the averaged representation of the word representation h_c are transformed to dimension d_k by trainable parameters:

$$\alpha_n = \frac{\boldsymbol{exp}^{\frac{k_n^c \times q_a}{\sqrt{d_k}}}(q_a, k_n^c)}{\sum_{n=1}^{N} \boldsymbol{exp}^{\frac{k_n^c \times q_a}{\sqrt{d_k}}}(q_a, k_n^c)} \tag{6}$$

We use a spread margin loss, L_k for each top-level capsule k to directly maximize the gap between the activation of the target class. Overall, our loss function L is the sum of the losses of all mentioned capsules:

$$L = \sum_{k \neq t} (\boldsymbol{max}(0, m - (a_t - a_k))) \tag{7}$$

5 Explanation Tool

The explanation for an algorithm is importance in some fields, such as clinical and financial decisions, because its results affect people directly. The European Union brings the *right to explanation* regulation [Goodman and Flaxman2017] into force in 2018, which is a right to be given an explanation for an output of the algorithm. For example, a person who applies for a loan and is denied may ask for an explanation. However, machine learning algorithm is data driven, even the algorithm designers have no idea how it works, especially for the deep neural networks with thousands of parameters. Nowadays, a large amount of explanation methods [Guidotti et al.2018, Olah et al.2017, Ancona et al.2017, Koh and Liang2017, Ribeiro et al.2016] have been proposed to reveal the behavior of deep neural networks. Post hoc methods especially the attribution methods, is a big branch of explanation methods, which assign the output score to the contributions of input features.

$$\begin{aligned} \mathrm{IG}(x; F)_i = & \frac{x_i - x_i'}{m} \\ & \times \sum_{k=1}^{m} \frac{\partial F(x' + k/m \times (x_i - x_i'))}{\partial x_i}. \end{aligned} \tag{8}$$

We utilize *iNNvestigate* [Alber et al.2018] a toolbox for explanation methods to evaluate on different tasks. Especially, we employ Integrated Gradients [Sundararajan et al.2017] as a tool, which is similarly to GradInput and computes the average value while the input varies along a linear path from a baseline x' to x. It solve the problem of Sensitivity property violation. The baseline x' is defined by the user and often chosen to be zero. m is the number of steps in the Riemman approximation of the integral. Our explanation tool is released for other researchers to use.

6 Experiments and Results

For the training details, the method of cross-entropy loss was used in CNN, RNN, and BERT. And we used Adam as the optimizer, and the learning rate is set as 0.001. And also, we added dropout and early stop trick. The dropout trick randomly abandons a certain proportion of nodes in the training process to prevent the occurrence of over-fitting. Finally, dropout is Adopted as 0.5. The effects of the model are evaluated by early stop technology on the validation set after each iteration. When the validation set's evaluation result no longer improves in N consecutive rounds, the iterative process is truncated, and the process of training is suspended. The number of N has impacts on the time consumed and Whether the model is converge or not. We implemented our models with PyTorch, moreover, we finetuned the pre-trained based model, BERT, on two NVIDIA RTX 2080Ti GPUs. We also performed an ablation analysis of our HACN model. The weighted F1 drops 1.1% when we remove the hierarchical attention module.

We carry out several experiments during the evaluation phase, and the best experiment is taken into account for the evaluation phase. The systems are evaluated with the official competition metric, Precision, Recall, and Macro Averaged F1 score. What is more, this task is multiple classifications, so macro average, weighted average, micro average are also employed.

In the filtering of insulting language in Chinese, the best performing model achieves a macro averaged F1-score of 94.05%. In the situation of antisocial language, the best performing system for the Chinese achieves a macro averaged F1-score of 97.18%. Finally, in the detection of illegal language, the state of the art NLP model for the Chinese is up to a macro averaged F1-score of 67.30%. However, we adopted

Figure 2: Classification results with explanation tool.

a weighted average as a uniform evaluating method due to the unbalanced classes. Our hierarchical attention capsule network (HACN) is the best-performed model with an excellent result of 94.86%.

7 Analysis

In our experiment, we can find that RNN has an acceptable performance, however, there are some obvious shortcomings. When handling unbalanced data, such as a high recall rate, fail to classify certain classes because the class lacks visible character.

It is also necessary to remark that the experiments are performed with the default parameters. Thus there is an additional field for improvement with some finetuning, which we plan to consider for future research. Moreover, we note that the label distribution is extremely imbalanced because there might be a bias introduced by the algorithms.

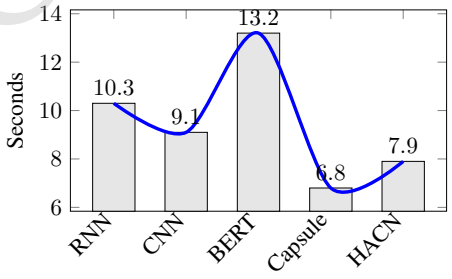
Extending from the experiment above, we present a comparison experiment in Figure, where we record the valid accuracy over time and spot trends with different systems. Figure 3 illustrates that our proposed HACN model can quickly converge to its stable equilibrium values. In the meantime, the starting point shows that HACN can get a promising result in a short time (after the first epoch).

Figure 4 shows times spent in the testing step for 3,759 test sentences when using different systems. The curves generated with these results suggested that our proposed HACN model can achieve the classification at the quickest speed. Considering the deployment in a sound system, we only compared the testing step. Nevertheless, we believe we still have substantial advantages in training because there is no need to pretrain the model on a large number of data compared with BERT.

Model	classes	Precision	Recall	F_1
SVM	Neutral	0.7895	0.8211	0.8050
	Insulting	0.8041	0.8444	0.8238
	Antisocial	0.7917	0.2000	0.3193
	Illegal	0.2188	0.1923	0.2047
	Weighted	0.7786	0.7966	0.7824
CNN	Neutral	0.9289	0.9524	**0.9405**
	Insulting	0.9559	0.9657	0.9607
	Antisocial	0.8056	0.9062	**0.8529**
	Illegal	0.4286	0.0380	0.0698
	Macro	0.7797	0.7156	0.7060
	Weighted	0.9270	0.9369	0.9281
	Micro	0.9369	0.9369	0.9369
RNN	Neutral	0.9171	0.9371	0.9270
	Insulting	0.9608	0.9446	0.9526
	Antisocial	0.7120	0.9271	0.8054
	Illegal	0.3415	0.1772	0.2333
	Macro	0.7328	0.7456	0.7296
	Weighted	0.9192	0.9233	0.9202
	Micro	0.9233	0.9233	0.9233
BERT	Neutral	0.9200	0.9459	0.9328
	Insulting	0.9666	0.9771	**0.9718**
	Antisocial	0.7593	0.8542	0.8039
	Illegal	0.7902	0.5861	**0.6730**
	Macro	0.8590	0.8408	0.8454
	Weighted	0.9260	0.9284	0.9259
	Micro	0.9284	0.9284	0.9259
Capsule	Weighted	0.9334	0.9419	0.9376
HACN	Weighted	0.9437	0.9528	**0.9486**

Table 2: Experimental Results and comparisons of our capsule networks and baselines.

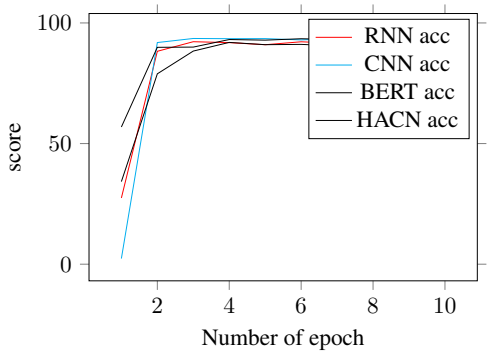
Figure4: Time costing in testing



Furthermore, We show a deep analysis of the mis-classified cases in the evaluation process of our experiments. Thus, we do a manual analysis for those mis-classified samples. This analysis aims at getting a deep comprehension of the areas our classifiers are lacking in. Our model fails to classify some metaphorical offensive words. It is usual for human to use euphemisms to tone down swear words in some situations. For some other cases, our model classifies some profanity text as offensive, which is actually not offensive. The classifier could miss these word variants, especially when the word variant is the only offensive word in the given sentence. In another situation, the word "sucks" is the only word that is often used offensively. However, the given tweet is not offensive because the author only describes their mood instead of insulting someone else. These misclassifications seem to indicate that

the classifier reacts to trigger words with negative connotations but may not be capable of interpreting the words concerning the broader context.

Figure3: Valid Accuracy Performance of each Epoch



8 Conclusion and Future Work

This work presents a Chinese corpus of offensive language crawled from microblog entries and video comments and manually categorized into 4 categories, and several models, including an allegedly novel architecture: Hierarchical Attention Capsule Network, for classification tested on the corpus. we describe the data-set (with simple baselines) and then talk about the modeling with both standard and non-standard methods and tools for explanations. For the dataset, we present an annotated corpus of offensive language in the online world, consisting of sentences and the corresponding annotations. The corpus consists of 18707 sentences annotated with four classes, including neutral language, insulting language, antisocial language, and illegal language. We also present several systems used for the classification of offensive language. The baselines are SVM, RNN, CNN, and BERT. What's more, we present a novel capsule network (HACN) with hierarchical attention to model the semantic structure. The best F1 score of 94.86% is achieved when using HACN. Finally, we propose an explanation tool to illustrate what our systems have learned.

Identifying offensive language in the online world is also interdisciplinary, as it overlaps with psychology, sociology, and economics, while also raising legal and ethical questions, so we expect it to attract a broader audience. Thus, in the future, we would like to bring the ideas and research achievements of other related fields to deliver and share technology and solutions for offensive language from online user-generated content.

Acknowledgements

This research is supported by the National Language Commission Key Research Project (ZDI135-61), the National Natural Science Foundation of China (No.61532008 and 61872157), and the National Science Foundation of China (61572223).

References

Maximilian Alber, Sebastian Lapuschkin, Philipp Seegerer, Miriam Hägele, Kristof T Schütt, Grégoire Montavon, Wojciech Samek, Klaus-Robert Müller, Sven Dähne, and Pieter-Jan Kindermans. 2018. innvestigate neural networks! *arXiv preprint arXiv:1808.04260*.

Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. 2017. A unified view of gradient-based attribution methods for deep neural networks. In *NIPS 2017-Workshop on Interpreting, Explaining and Visualizing Deep Learning*. ETH Zurich.

Pete Burnap and Matthew L Williams. 2015. Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making. *Policy & Internet*, 7(2):223–242.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Eleventh international aaai conference on web and social media*.

Nemanja Djuric, Jing Zhou, Robin Morris, Mihajlo Grbovic, Vladan Radosavljevic, and Narayan Bhamidipati. 2015. Hate speech detection with comment embeddings. In *Proceedings of the 24th international conference on world wide web*, pages 29–30. ACM.

Paula Fortuna, José Ferreira, Luiz Pires, Guilherme Routar, and Sérgio Nunes. 2018. Merging datasets for aggressive text identification. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 128–139.

Spiros V Georgakopoulos, Sotiris K Tasoulis, Aristidis G Vrahatis, and Vassilis P Plagianakos. 2018. Convolutional neural networks for toxic comment classification. In *Proceedings of the 10th Hellenic Conference on Artificial Intelligence*, page 35. ACM.

Bryce Goodman and Seth Flaxman. 2017. European union regulations on algorithmic decision-making and a âright to explanationâ. *AI Magazine*, 38(3):50–57.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2018. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):93.

Geoffrey E Hinton, Alex Krizhevsky, and Sida D Wang. 2011. Transforming auto-encoders. In *International Conference on Artificial Neural Networks*, pages 44–51. Springer.

Binxuan Huang and Kathleen M Carley. 2019. Parameterized convolutional neural networks for aspect level sentiment classification. *arXiv preprint arXiv:1909.06276*.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 151–160. Association for Computational Linguistics.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1885–1894. JMLR. org.

Ritesh Kumar, Atul Kr Ojha, Shervin Malmasi, and Marcos Zampieri. 2018. Benchmarking aggression identification in social media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 1–11.

Irene Kwok and Yuzhou Wang. 2013. Locate the hate: Detecting tweets against blacks. In *Twenty-seventh AAAI conference on artificial intelligence*.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018. Transformation networks for target-oriented sentiment classification. *arXiv preprint arXiv:1805.01086*.

Karsten Müller and Carlo Schwarz. 2018. Fanning the flames of hate: Social media and hate crime. *Available at SSRN 3082972*.

Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. 2017. Feature visualization. *Distill*, 2(11):e7.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144. ACM.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR. org.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2015. Effective lstms for target-dependent sentiment classification. *arXiv preprint arXiv:1512.01100*.

Yequan Wang, Minlie Huang, Li Zhao, et al. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the germeval 2018 shared task on the identification of offensive language.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. *arXiv preprint arXiv:1805.07043*.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019a. Predicting the type and target of offensive posts in social media. *arXiv preprint arXiv:1902.09666*.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019b. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). *arXiv preprint arXiv:1903.08983*.

Wei Zhao, Jianbo Ye, Min Yang, Zeyang Lei, Suofei Zhang, and Zhou Zhao. 2018. Investigating capsule networks with dynamic routing for text classification. *arXiv preprint arXiv:1804.00538*.

LiveQA: A Question Answering Dataset over Sports Live

Qianying Liu[*12], **Sicong Jiang**[*1], **Yizhong Wang**[13] **and Sujian Li**[1]
[1] Key Laboratory of Computational Linguistics, MOE, Peking University
[2] Graduate School of Informatics, Kyoto University
[3] University of Washington
ying@nlp.ist.i.kyoto-u.ac.jp; 512580728@qq.com;
yizhongw@cs.washington.edu; lisujian@pku.edu.cn

Abstract

In this paper, we introduce LiveQA, a new question answering dataset constructed from play-by-play live broadcast. It contains 117k multiple-choice questions written by human commentators for over 1,670 NBA games, which are collected from the Chinese Hupu[1] website. Derived from the characteristics of sports games, LiveQA can potentially test the reasoning ability across timeline-based live broadcasts, which is challenging compared to the existing datasets. In LiveQA, the questions require understanding the timeline, tracking events or doing mathematical computations. Our preliminary experiments show that the dataset introduces a challenging problem for question answering models, and a strong baseline model only achieves the accuracy of 53.1% and cannot beat the dominant option rule. We release the code and data of this paper for future research.[23]

1 Introduction

The research of question answering (QA), where a system needs to understand a piece of reading material and answer corresponding questions, has drawn considerable attention in recent years. While various QA datasets have been constructed to study how a QA system can understand a specific passage, the common sense knowledge and so on (Rajpurkar et al., 2016; Lai et al., 2017; Dunn et al., 2017; Rajpurkar et al., 2018), most questions in these datasets could be given their answers by extracting from a few relevant sentences so that the model only needs to find a small set of supporting evidences,whose temporal ordering does not effect the final answer. In other words, these questions are raised only considering a fixed document. However, in the real-life question answering, a question could have its **timelines**. To infer the answer, a good model needs to understand series of timeline information. For example, the question "how many points did Lebron James have?" would have different answers based on the time when the question was asked during a basketball game, and the answer would continuously change during the game. The other question "Which team would first earn 10 points?" would require a system to track down information of scoring points along the timeline until one team achieves 10 points.

According to the analysis above, we consider the timeline-based question answering problem as a gap which has not been covered by existing datasets. Thus, in this work we hope to construct a dataset where passages and questions both have timelines and question respondents are required to judge what information should be gathered for the questions involved in a timeline. Such a timeline inference-involved QA dataset introduces a new research line of reading comprehension, that evaluates the ability of understanding temporal information of a QA model.

Additionally, the real-world questions are often involved in some math calculation, such as addition, subtraction and counting. To answer the questions correctly, one not only needs to locate some specific sentences, but also do calculation or comparison on the extracted evidence. For example, "How many points did the winner team win?" needs one system to perform subtraction on the final score to get the correct answer.

[*] This denotes equal contribution.
[1] https://nba.hupu.com/games
[2] code: https://github.com/PKU-TANGENT/GAReader-LiveQA
[3] data: https://github.com/PKU-TANGENT/LiveQA

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1057-1067, Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

1: 第一节两队得分之和能否达到 51 分或更多？
Will the sum of the scores of the two teams in the first quarter reach 51 points or more?

2: 本场勒布朗-詹姆斯能否得到 35 分或更多？
Will Lebron James get 35 points or more in this match?

3: 暂停回来，开拓者首次进攻会不会得分？
Will the Portland Trailblazers score in the first attack after the timeout ends?

4: 本节比赛双方还能否再次命中三分球？
Will another three-pointer take place in the rest of the quarter?

Figure 1: Question Examples from the LiveQA dataset.

To these ends, we construct a QA dataset *LiveQA* based on a Hupu-live-broadcasting-dataset, which is a set of Chinese live-broadcasting passages of NBA. Hupu is a sports news website that has live-broadcasting for basketball games. In the Hupu-live-broadcasting, the host of one sport game describes the details of the game vividly with emotion and different sentence structures, and presents many game-related quizzes during the game. We collect the description texts and their quizzes into LiveQA. Answering the quizzes requires one model to correctly understand the timeline information of the context: some quizzes ask about information of one-whole quarter of the game or which player reaches a certain score earlier. Thus, the model needs to fully understand the temporal information of the live-broadcasting and then performs inference based on the temporal information. Figure 1 shows four question examples in the LiveQA dataset. Answering the first two questions requires an addition math operation, and the 3^{rd} and 4^{th} questions need comparison operation. Meanwhile, we can see that all these questions are time-dependent and require temporal inference.

In summarize, the main characteristics of our LiveQA dataset include the following two aspects. Firstly, the questions are time-awared. The model needs temporal inference to obtain the final answer. Secondly, in our dataset, reading comprehension is not limited to extracting a few specific text spans from the document, but is involved with math calculation. These characteristics make LiveQA challenging for previous QA systems to answer its questions. In this paper, we present an analysis of the resulting dataset to show how these characteristics appear in the data. We also show how questions are involved with temporal inference, and these questions also require mathematical inference. To demonstrate how these characteristics affect the performance of the QA model, we design a pipeline method, which first tries to find supporting sentences and then uses a strong baseline multi-hop inference model named Gated-Attention Reader, to judge the baseline performance on LiveQA. Our experimental results show that such strong baseline model only slightly exceeds random choice, which achieve 53.1% and cannot beat the dominant option rule. The analysis and experimental results show how this dataset can effectively examine how a QA system can perform multi-hop temporal and mathematical inference, which is not covered by previous studies.

The following of this paper is organized as follows: In section 2, we give a brief introduction of current QA research lines and research on live text processing. In section 3, we describe how we constructed the dataset. In section 4, we give statistics of the dataset and analyse the timelineness and mathematical

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under gravity. The main forms of precipitation include drizzle, rain, sleet, snow, graupel and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals within a cloud. Short, intense periods of rain in scattered locations are called "showers".

What causes precipitation to fall?

gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?

graupel

Where do water droplets collide with ice crystals to form precipitation?

within a cloud

Figure 2: Examples of question-answer pairs in SQuAD

inference in the data. In section 5, we give evaluation results of baseline models and error analysis.

2 Related Works

In this section, we mainly introduce the various QA datasets which can be categorized as datasets with extractive answers, datasets with descriptive answers and datasets with multiple-choice questions.

2.1 Datasets with Extractive Answers

A number of QA datasets consist of numerous documents or passages which have considerable length. Each passage is equipped with several questions, answers of which are segments of the passage. The goal of a reading comprehension model is to find the correct text span. In other words, it may offer a begin position and an end position in the passage instead of generating the words itself. Such corpora are regarded as datasets with extractive answers.

The most famous dataset of this kind is Stanford Qustion Answering Dataset (SQuAD) (Rajpurkar et al., 2016). SQuAD v1.0 consists of 107,785 question-answer pairs compiled by crowdworkers from 536 Wikipedia articles, and is much larger than previous manually labeled datasets. Over 50,000 unanswerable questions are added in SQuAD v2.0 (Rajpurkar et al., 2018). It is more challenging for existing models because they have to make more unreliable guesses. As performances on SQuAD have become a common way to evaluate models, some experts regard SQuAD as the ImageNet (Deng et al., 2009) dataset in the NLP field.

Another frequently used dataset with extractive answers is CNN/Daily Mail dataset (Hermann et al., 2015), which was released by Google DeepMind and University of Oxford in 2015. One shining point of it is that each entity is anonymised by using an abstract entity marker to prevent models from using word-level information or n-gram models to find the answer rather than comprehending the passage.

CBT (Hill et al., 2015a), NewsQA (Trischler et al., 2016), TriviaQA (Joshi et al., 2017) and many other datasets can also be categorized into this class. They constitute a high proportion of MRC datasets, and can test the abilities of extractive models in various ways. Thus, we aim to construct a novel dataset, on which extractive models are likely to make mistakes in looking for the location of an answer, that the dataset can open a new research line for question answering by testifying the ability of models to understand timelineness.

2.2 Datasets with Descriptive Answers

Instead of selecting a span from the passage, datasets with descriptive answers require a reading comprehension model to generate whole and stand-alone sentences. These corpora are more closer to reality, because most questions in the real world cannot be solved simply by presenting a span or an entity. This kind of dataset is getting popular nowadays, and may be the trend of the development of MRC datasets.

Original Version	Anonymised Version
Context	
The BBC producer allegedly struck by Jeremy Clarkson will not press charges against the "Top Gear" host, his lawyer said Friday. Clarkson, who hosted one of the most-watched television shows in the world, was dropped by the BBC Wednesday after an internal investigation by the British broadcaster found he had subjected producer Oisin Tymon "to an unprovoked physical and verbal attack." ...	the *ent381* producer allegedly struck by *ent212* will not press charges against the " *ent153* " host , his lawyer said friday . *ent212* , who hosted one of the most - watched television shows in the world , was dropped by the *ent381* wednesday after an internal investigation by the *ent180* broadcaster found he had subjected producer *ent193* " to an unprovoked physical and verbal attack . " ...
Query	
Producer **X** will not press charges against Jeremy Clarkson, his lawyer says.	producer **X** will not press charges against *ent212* , his lawyer says .
Answer	
Oisin Tymon	*ent193*

Figure 3: An example of anonymised entity in CNN/Daily Mail

Text	Question	Choices	Answer
.....			
哈登弧顶控球！！面对克莱-汤普森紧逼！			
左侧横移！！！！			
哨响！克莱-汤普森逼得太紧了！吃到一次犯规！			
勇士要个长暂停！！！！！			
	停回来，勇士第一轮进攻能否得分？（罚球也算，直到球权转换）	能/不能	不能
稍等！！！			
第一节还有7分29秒！			
	本节勇士队最后一分是否由伊戈达拉获得?	是/不是	不是
好的！！！比赛继续！！！			
哈登走上罚球线！！！			
两罚都有！！14-9			
利文斯顿弧顶控球！！！			
.....			

Figure 4: A partial example of LiveQA timeline.

MS MARCO (Microsoft MAchine Reading Comprehension) (Nguyen et al., 2016) is a dataset released by Microsoft in 2016. This dataset aims to address the questions and documents in the real world, as its questions are sampled from Bing's search query logs and its passages are extracted from web documents retrieved by Bing. The questions in MS MARCO are about ten times as many as SQuAD, and each question is equipped with a human generated answer. The dataset also includes unanswaerable questions. All of the above characteristics make MS MARCO worthy of trying.

NarrariveQA (Kočiský et al., 2017) is another dataset with descriptive answers released by DeepMind and University of Oxford in 2017. The dataset consists of stories, books and movie scripts, with human written questions and answers based solely on human-generated abstractive summaries. Answering such questions requires readers to integrate information which may distribute across several statements throughout the document, and generate a cogent answer on the basis of this integrated information. In other words, they test that the reader comprehends language, not just that it can pattern match. We judge it a referential advantage of a dataset, so LiveQA requires the ability of tracking events as well as we show in Figure 4, which will be detailedly introduced in following sections.

2.3 Datasets with Multiple-choice Questions

Datasets with descriptive answers have various advantages, but they are relatively difficult to evaluate the system performance precisely and objectively. Thus, corpuses with more gradable QA-pairs are also needed, which leads to the development of datasets with multiple-choice questions. Through diversified types of questions, these datasets can examine almost every ability of a reading comprehension model mentioned above and are easier to get a conclusive score. Many datasets of this kind have been released in recent years, and they have covered multiple domains. For example, RACE (Lai et al., 2017) and CLOTH (Xie et al., 2017) are collected from English exams, MCTest (Richardson et al., 2013) is sampled from friction stories, and ARC (Clark et al., 2018) is extracted from science-related sentences. However, there is still not a reliable dataset which is built on sports events for MRC. Thus, our LiveQA dataset has the potential for filling several gaps in the field of MRC.

2.4 Live Text Processing

Previously, various studies have been conducted on automatically generate sports news from live text commentary scripts, which has been seen as a summarization task. Zhang et al. (2016) proposed an investigation on summarization of sports news from live text commentary scripts, where they treat this task as a special kind of document summarization based on sentence extraction in a supervised learning to rank framework. Yao et al. (2017) further verify the feasibility of a more challenging setting to generate news report on the fly by treating live text input as a stream for sentence selection. Wan et el. (2016) studied dealing with the summarization task in Chinese. All these studies focuses on using the live text commentary scripts as the input of summarization and selecting sentences to form the summary. So far, we are the first to point out the importance of timelineness and mathematical reasoning in understanding live text commentary scripts.

3 LiveQA: Dataset Construction

In this section we introduce how to construct LiveQA from the raw Hupu text and present the corpus statistics. The whole process of building LiveQA mainly includes crawling the raw data and acquiring the game texts with corresponding quizzes.

3.1 Data Crawling

In Hupu, each game has a unique ID which is connected with its url. We collected the IDs from the Hupu's live schedule pages. Their formats are *https://nba.hupu.com/games/year-month-date*. There are links to all the NBA games so that their IDs can be saved. After we saved the IDs into a file, we used the web debugging tool Fiddler to get a sample of the url of a game, and then changed the IDs in the url to make access to all the games. We are authorized by the legal department of hupu website to construct the dataset for only academical purpose.

3.2 Data Processing

Most previous datasets usually do not care for the storing positions of the passages and their questions. But in our dataset, the quizzes and the contexts shouldn't be separated because the time (the position) one quiz occurs is quite important for the final answer. If we separate the quizzes and their contexts, most quizzes may have different answers and cannot be answered even by human. Here we use some rules to clean the dataset. The lines starting with '@' are always interactions between the host and some active readers, which are irrelevant to the game. During the half-time break, the host will give out some "gift" questions to please the readers waiting for the second-half. Some of the questions appear like normal quizzes, but they need information outside the game to answer them, thus we exclude them from the data (i.e.. which team won more matches in the history?). Usually they have a prefix – "中场福利" in common. Besides, we exclude the descriptions of pictures from our data.

3.3 Data Structure

Here we give an explanation for the structure for each independent data sample.

For each live-stream of one match, the timeline data is sorted in time order, where the questions are inserted into the corresponding timeline position so that the timeline features of the questions could be inferred. As we show in Figure 4, the plain content text and the question text share the same timeline, but question records have choices and answers along with the text. For each record in the timeline, it either contains a piece of live-stream text or a question bonded with the corresponding choices and the correct answer. Each question has two answer choices.

4 Dataset Statistics

Element	Count
Document	1,670
Sentences in Total	1786616
Sentences in Average	1069.83
Quizzes in Total	117050
Quizzes in Average	70.09

Table 1: The details of statistics of the dataset.

We show the statistics of the dataset in Table 1. The LiveQA dataset contains 1,670 documents, each of which has 70.09 quizzes and 1069.83 sentences on average. Next we analyze the questions from two different views. First, we simply classify the questions according to the positions of their answers. In general, some of the questions can be solved by extracting information from neighboring sentences, which involves a time period of the origin game. Such questions occupy 68.6% of all the questions. Some questions can be replied only by summarizing all the information after the game ends and occupy about 30.6%. Still, there exists a small percentage (0.8%) of questions which are impossible to be answered from the passage. Table 2 lists some examples for each type of questions.

Because most of the questions are associated with some numerical data in the game, we also classify the questions according to how the numerical data is performed. Four types of operations are commonly used including: *Comparison*, *Calculation*, *Inference* and *Tracking*. Then the questions are correspondingly classified are introduced in the following subsections. We also give some examples in Table 3.

4.1 Comparison

To answer the comparison questions, we usually need to find the comparative figures for the corresponding objects. For example, the commentator asks which of the two players will score more or which team will win. The second row in Table 3 belong to the *Comparison* questions. The easiest way to solve this kind of questions is to find the two figures appearing in the text and comparing them. It is likely to acquire such figures after the game ends, and the specific figures usually appear together in a summary of the game in the end. Thus, matching techniques are still necessary to the final answer.

4.2 Calculation

The *Calculation* questions require extracting two or three figures and calculating their sum or difference. They differ from the Comparison questions in two ways – the figures are more scattered and a calculation step is needed. This means that a respondent has to look for more information efficiently. After the figures are obtained, if a respondent misjudges the type or the direction of the calculation, he will still probably get a wrong answer. Similar to the *Comparison* questions, the *Calculation* questions are mainly dependent on the correct sentences where the figures are located. These two kinds of questions are relatively easy compared to those ones which are not based on certain sentences. The second row in Figure 3 give two example questions.

4.3 Inference

The third and fourth type of questions require the ability of summarizing and tracking information. A question of the third type needs a respondent to infer some figures through the text. For example, a ques-

Question type	Proportion	Example	Translation
Answered after the game ends	30.6%	本场森林狼能否赢快船4分或更多?	Will the Timberwolves beat the Clippers by more than 4 points?
		本场比赛谁会赢?	Which team will win?
Answered through the context	68.6%	第二节谁先命中三分球?	Which team will make a three-pointer first in the second quarter?
		首节最后一分会不会由罚球获得?	Will the last point in the first quarter scored through a free-throw?
Impossible to answer	0.8%	第二节比赛开始1分30秒时间内会不会有三分球命中?	Will a three-pointer be made in the first 90s of the second quarter?
		本场比赛会不会在北京时间10时58分之前结束?	Will the game end before 10:58 a.m.?

Table 2: Questions statistics and examples sorted by the location of their corresponding evidence.

Question type	Proportion	Example	Translation
Comparison	16.6%	勒布朗-詹姆斯本场能否得到26分或更多?	Will Lebron James get 26 points or more in this game?
		本场谁的得分会更高?	Who will get higher score in this game?
Calculation	25.4%	本场凯尔特人能否赢猛龙3分或更多?	Will the Celtics beat the Raptors by more than 3 points?
		本场两队总得分能否达到207分或更多?	Will the total score of the two teams reach 207 points or more?
Inference	28.5%	暂停回来，雷霆队首次进攻能否得分?	After the timeout, will the Thunder score in their first round of attack?
		第二节比赛雷霆队最后一分会不会由威斯布鲁克得到?	Will the last point of the Thunder in the second quarter be got by Westbrook?
Tracking	29.5%	太阳队能否在本场命中8个或更多三分球?	Will the Suns make 8 three-pointers or more in this game?
		凯文-乐福首节犯规数会不会达到2次?	Will Kevin Love commit 2 fouls or more in the first quarter?

Table 3: Questions statistics and examples sorted by how the inference process is done.

tion may be ”After this timeout, will the Cavaliers score in the first round of attack?”. The commentator obviously will not say that ”The Cavaliers scored 2 points.” or ”The Cavaliers didn’t score.” A respondent may get the answer as ”JR Smith makes a 2-point shot.” Another example is ”Will the last point of this quarter be scored through a free throw?” The information comes from the text of ”Anthony Davis makes his second free throw ... The match ends!”. It is impossible to get a reasonable answer by matching.

4.4 Tracking

The Tracking questions require more scattered information. A respondent should collect and accumulate specific information from a part of the passage, as the question is based on events happening repeatedly in a quarter or half of the game. For example, some questions ask about how many free-throws a player *A* will make in a quarter. As this figure does not appear in the passage, a respondent needs to count how many times the event '*A* makes a free-throw' occurs. In other words, it is necessary to track events relevant to the player '*A*' and 'free-throw'. When the player(*A*) is replaced with one team name, the new question is even more difficult because the information about each player belonging to the team should be tracked. Therefore, information tracking leads this kind of questions to be the most challenging ones in the dataset.

5 Baseline Models and Results

5.1 Models

To evaluate the QA performance on the LiveQA dataset, we implement 3 baseline models. The first is based on random selection, where the system randomly chooses a choice as the answer. The second is to choose the dominant option of each question. More concretely, 80.0% of questions are in format of 'yes' and 'no', where 57.8% has the answer 'no'. For the other multiple choice questions, 50.6% of them take the second option as the right answer. Thus, for 'yes/no' questions, we choose 'no', otherwise we choose the second option.

We also build a neural-network style baseline for our dataset to evaluate how state-of-the-art QA systems perform on the LiveQA dataset. Due to the uniqueness of our dataset, most of existing machine comprehension models are not suitable to it. For example, the QANet (Yu et al., 2018) model, which used to be a state-of-art model of SQuAD (Rajpurkar et al., 2016), is unavailable because it predicts the probability distribution of an answer's starting position and ending position in the context. But in LiveQA, a number of right answers do not directly appear in the context (e.g. an answer in format of 'can' or 'cannot'). Up to now, none of machine reading comprehension models has been designed for a dataset with consideration of timeline and mathematical computations. That means that the existing ones will not be likely to perform well on our dataset. The closest work to ours is multi-hop question answering, and thus we use a novel model Gated-Attention Reader (Dhingra et al., 2016) to experiment on LiveQA.

Gated-Attention Reader (GA) is an attention mechanism which uses multiplicative interactions between the query embedding and intermediate states of a recurrent neural network reader. GA enables a model to scan one document and the questions iteratively for multiple passes, and thus the multi-hop structure can target on most relevant parts of the document. It used to be the state-of-art model of several datasets, such as CNN/Daily Mail dataset (Hermann et al., 2015) and CBT dataset (Hill et al., 2015b).

The full context, which is usually composed of more than 1,000 sentences on average, is too heavy for GA as input. To apply GA to our dataset, we propose a pipeline method to first extract a set of candidate evidence sentences from the full content, and then apply the GA model on this set of sentences to predict the final answer. We employ TF-IDF style matching score to extract 50 most relevant sentences as the supporting evidence. To improve the accuracy of selecting the evidence candidates, if the question clearly requires some information after the game ends, we use the ending part of the content as the input.

Specifically, taken the embedding representation of a token, the Bi-directional Gated Recurrent Units (BiGRU) process the sequence in both forward and backward directions to produce two sequences of token-level representations, which are concatenated at the output as the final representation of the token. To perform multi-hop inference, the GA model reads the document and the query over k horizontal layers, where layer k receives the contextual embeddings $X_{(k-1)}$ of the document from the previous layer. At each layer, the document representation $D^{(k)}$ is computed by taking the full output of a document BiGRU where the previous layer embedding $X_{(k-1)}$ is the input. At the same time, a layer-specific query representation $Q^{(k)}$ is computed as the full output of a separate query BiGRU taking the query embedding Y as the input. The Gated-Attention is applied to $D^{(k)}$ and $Q^{(k)}$ to compute the contextual

embedding $X^{(k)}$.

$$X^{(k)} = GAttn(BiGRU(X^{(k-1)}), BiGRU(Y)) \quad (1)$$

After obtaining the query-awared document representation, we perform answer prediction by matching the similarity of answer and content. We use bidirectional Gated Recurrent Units to encode the candidate answers into vectors $A^(i)$, and then we compute matching score between summarized document and candidates using a bilinear attention. Finally we calculate the probability distribution of the options with softmax. The operations are similar to those in RACE (Lai et al., 2017).

$$s = softmax([Blin(A^i, D^{(k)});]_n^{i=1}) \quad (2)$$

5.2 Model Evaluation

Model	Acc
Random	50.0%
Dominant	**56.4%**
GA	53.1%

Table 4: The results of different baseline models on the test set. Random denotes randomly selecting an answer. Dominate denotes selecting the dominate option. GA denotes the gated-attention reader.

For the three baseline models, performance is reported with the accuracy on the test set in Table 4. The random selection method (Random) scores 50.0%, while the dominant option method (Dominate) reaches a score of 56.4%, which shows that our dataset does not have a certain pattern for the answers. Meanwhile, GA, which is a strong baseline for previous question answering problems, failed to perform better than the dominant option method and only achieves a score of 53.1%. Such results show that our dataset is challenging and needs further investigation for model design. In future work, how to incorporate temporal information and mathematical calculation into a QA model is the focus.

5.3 Case Study

In this subsection, we further analyze the prediction ability of the GA model. Table 5 shows some prediction cases in experimental results. From the first two questions, we can see that the model gives the correct answers when judging the result of a specific event. But for the other three questions which involve multiple events, the model fails to answer them correctly. A possible explanation is that, although GA is designed for multi-hop inference, it lacks ability in both information tracking and math calculation, which makes it difficult for the model to track down some complicated events.

We can see, for reading comprehension models that extract answers based on the similarity between the answer and the content, they would fail on LiveQA due to the fact that they cannot track down temporal information nor perform mathematical calculation. To outperform existing models on LiveQA, the system should consider focusing on tracking information of a certain event through the timeline. It should also have the ability to perform mathematical inference between different contents.

6 Conclusion

In this paper, we present LiveQA, a question answering dataset constructed from play-by-play live broadcast. LiveQA can evaluate a machine reading comprehension model in its ability to understand the timeline, track events and do mathematical calculation. It consists of 117k questions, which are time-dependent and need math inference. Due to the novel characteristics, it is hard for existing QA models to perform well on LiveQA. We expect our dataset will stimulate the development of more advanced machine comprehension models.

Question	Translation	Correct answer	Answer given by the model
跳球之争！本场比赛哪支球队获得第一轮进攻球权?	Jump ball fight! Which team will win the chance of the first round of offence?	勇士(The Warriors)	勇士(The Warriors)
湖人全场总得分是奇数还是偶数?	Will the total score of the Lakers at the end of the game be odd or even?	奇数(odd)	奇数(odd)
尼克杨第二节能否命中3分球?	Can Nick Young make a three pointer in the second quarter?	能(Yes)	不能(No)
第三节结束，76人能否领先湖人4分或更多?	At the end of the third quarter, Will the 76ers lead the Lakers by 4 points of more?	不能(No)	能(Yes)
谁先获得30分?	Who will score his 30th point earlier?	24分的哈登(James Harden who has got 24 points)	25分的托马斯(Isaiah Thomas who has got 25 points)

Table 5: Cases in the experimental results

Acknowledgement

We thank the anonymous reviewers for their helpful comments on this paper. This work was partially supported by National Natural Science Foundation Project of China (61876009), National Key Research and Development Project (2019YFB1704002), and National Social Science Foundation Project of China (18ZDA295). The corresponding author of this paper is Sujian Li.

References

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Jia Deng, Wei Dong, Richard Socher, Li Jia Li, Kai Li, and Fei Fei Li. 2009. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision & Pattern Recognition*.

Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William W Cohen, and Ruslan Salakhutdinov. 2016. Gated-attention readers for text comprehension. *arXiv preprint arXiv:1606.01549*.

Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. 2017. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015a. The goldilocks principle: Reading children's books with explicit memory representations. *Computer Science*.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015b. The goldilocks principle: Reading children's books with explicit memory representations. *arXiv preprint arXiv:1511.02301*.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, and Edward Grefenstette. 2017. The narrativeqa reading comprehension challenge.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Tri Nguyen, Mir Rosenberg, Song Xia, Jianfeng Gao, and Deng Li. 2016. Ms marco: A human generated machine reading comprehension dataset.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250.*

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

Matthew Richardson, Christopher JC Burges, and Erin Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 193–203.

Adam Trischler, Wang Tong, Xingdi Yuan, Justin Harris, and Kaheer Suleman. 2016. Newsqa: A machine comprehension dataset.

Xiaojun Wan, Jianmin Zhang, Jin-ge Yao, and Tianming Wang. 2016. Overview of the nlpcc-iccpol 2016 shared task: Sports news generation from live webcast scripts. In Chin-Yew Lin, Nianwen Xue, Dongyan Zhao, Xuanjing Huang, and Yansong Feng, editors, *Natural Language Understanding and Intelligent Applications*, pages 870–875, Cham. Springer International Publishing.

Qizhe Xie, Guokun Lai, Zihang Dai, and Eduard Hovy. 2017. Large-scale cloze test dataset created by teachers. *arXiv preprint arXiv:1711.03225*.

Jin-ge Yao, Jianmin Zhang, Xiaojun Wan, and Jianguo Xiao. 2017. Content selection for real-time sports news construction from commentary texts. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 31–40.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension. *arXiv preprint arXiv:1804.09541*.

Jianmin Zhang, Jin-ge Yao, and Xiaojun Wan. 2016. Towards constructing sports news from live text commentary. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1361–1371.

Chinese and English Elementary Discourse Units Recognition based on Bi-LSTM-CRF Model

Yancui Li Chunxiao Lai Jike Feng Hongyu Feng
School of Information Engineering, Henan Institute of Science and Technology,
Xinxiang , Henan, China
Key Laboratory of Advanced Theory and Application in Statistics and Data Science
(East China Normal University), Ministry of Education,
Shanghai, China
Liyancui@hist.edu.cn

Abstract

Elementary Discourse Unit (EDU) recognition is the basic task of discourse analysis, and the Chinese and English discourse alignment corpus is helpful to the studies of EDU recognition. This paper first builds Chinese-English parallel discourse corpus, in which EDUs are annotated and aligned. Then, we present the framework of Bi-LSTM-CRF EDUs recognition model using word embedding, POS and syntactic features, which can combine the advantage of CRF and Bi-LSTM. The results show that F1 is about 2% higher than the traditional method. Compared with CRF and Bi-LSTM, the Bi-LSTM-CRF model can combine the advantages of them and obtains satisfactory results for Chinese and English EDUs recognition. The experiment of feature contribution shows that using all features together can get best result, the syntactic feature outperforms than other features.

1 Introduction

Discourse analysis is helpful for the performance of machine translation, question answering, summarization and other application. EDU recognition is a basic work in discourse analysis task. Only by recognition EDU, can we make further discourse analysis or other works. At present, the existing Chinese-English parallel corpus only align paragraphs, sentences and other linguistic units, but do not annotate bilingual EDUs alignment, which due to EDU recognition is mainly carried out on monolingual. However, EDUs recognition on Chinese and English is vital to bilingual analysis, machine translation et al. For Example1 is a bilingual sentence of Chinese and English, Chinese EDUs are numbered sequentially by e1, e2 and e3, and English EDUs are marked by e1′, e2′ and e3′. Obviously, e1 and e1′, e2 and e2′, e3 and e3′ are alignment pair.

Example 1 A) [京杭运河古来繁华，] ^{e1}[两岸商贾云集，]^{e2} [贸易发达。]^{e3}

B) [The Beijing - Hangzhou Grand Canal has been prosperous since ancient times,] $^{e1'}$ [with both sides of the bank swarming with merchants] $^{e2'}$ [and well - developed trade.]$^{e3'}$

The main work of this paper is recognition the EDUs of Chinese and English as much as possible. The following is the contribution of this paper:

We annotate Chinese and English discourse alignment corpus, which is first corpus contain EDUs alignment information as far as we know;

We get satisfactory results without any handcraft feature by using Bi-LSTM-CRF Model;

We conduct to find out the contribution of various model and features.

This paper combines existing research and Chinese-English discourse alignment corpus to identify and analyze Chinese-English EDUs. Section 2 builds Chinese-English EDUs alignment corpus; Section 3 describes the Bi-LSTM-CRF model and the framework this paper used; Section 4 reports and analyzes the experimental results; Section 5 overviews the related work; Finally, Section 6 summarizes this paper and points out the future research direction.

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1068-1078, Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

2 Chinese-English Alignment Corpus

2.1 Chinese-English EDUs alignment methods

In order to represent the discourse, the first task is to define the EDUs. Inspired by the work of Li et al(2014) and Feng (2013), we give the definition of Chinese EDUs. Firstly, a clause should contain more than one predicate, expressing not less than one proposition. Secondly, one EDU should have propositional function to another EDU. Finally, a clause should be segmented by some punctuation. As for English EDU, it is the corresponding content of Chinese EDU.

When annotate, we dividing Chinese sentence into parts, and adapting the alignment strategy of the source language is preferred. That is to say, it is segmented according to the established Chinese EDUs, and then align in English. Therefore, EDUs in Chinese and English sentences is correspondence. Such as Example 1, recognition and alignment are achieved under the guidance of this principle. Since Chinese EDUs are preferential when making alignment rules, some sentences with widely ranges may appear in English translation. These EDUs are not adjacent in English sentences, it will affect the alignment of Chinese and English EDUs. EDUs cannot be completely corresponding in this case, and the solution is to align the main parts.

2.2 Chinese-English alignment corpus

According to the alignment annotation principle mentioned in the 2.1. We annotate alignment corpus of Chinese and English. Corpus select from Xinhua daily, and we have marked 100 Chinese-English translation documents. The Chinese-English parallel corpus is marked with Chinese as the main language, supplemented the parallel EDUs by English.

Due to the marked Chinese-English alignment corpus has many contents, and experiments are mainly for EDUs, this paper mainly introduces the annotation principle of EDU in corpus. After practical operation and analysis, the following three points are obtained:

1) The meaning of English and Chinese sentences. According to the logical semantic relations, the corresponding relations of the adjacent EDUs in the alignment corpus can be found respectively, and the relationship is used to divide and align English-Chinese corpus.

2) Structure. Combined with the structure of Chinese language and English language, the order of subject-verb-object in English-Chinese is consistent, and the translation of some noun clauses and adverbial clauses are also consistent, so it is possible to find out the corresponding words in English-Chinese so as to find the corresponding sentence components in English-Chinese for division.

3) Following the punctuation clues. In the translated English corpus, the punctuation in English is mostly consistent with that in Chinese. And according to the distribution of punctuation, the meaning of the text and the translated English EDUs can be more clearly inferred.

There are 100 documents, 513 paragraphs, 899 Chinese sentences,1281 English sentences and 2153 Chinese-English EDU pairs which have been effectively marked. The Chinese EDU average length is 11 words, while the English EDU average length is 20 words. In the paper, the preset program is used to automatically find the parent node information of English EDUs, and the search is carried out in the automatic syntactic analysis tree of Stanford. The method of search is to look up the words from the beginning and the end of the English clauses successively until a common parent node is found in the syntactic tree. By the way of making statistics on the information of parent node which can be found, it is not difficult to find that the main syntactic structure which can make Chinese EDUs corresponding to English clauses are S、VP、NP、PP etc. The syntactic structure and occurrence frequency corresponding to English EDUs are shown in Figure1 From Figure1 we can see most of EDU’s syntactic tag are S and VP, which is consistent with the definition of our EDU.

2.3 Tagging Strategies and Consistency

Two senior students of Chinese department carried out annotation training under the guidance of the project supervisor. 20 parallel paragraphs were randomly selected from the Xinhua daily to mark training corpus. We developed a platform for EDU annotation. The annotation training is mainly composed of three stages: 1) The tutor demonstrates the annotation of 10 documents, and explains the main annotation

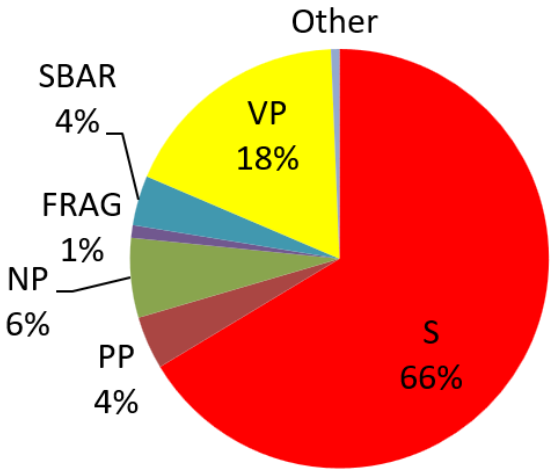


Figure 1: Syntactic structure distribution of the EDUs

strategies, the annotation method and the operation of the annotation platform; 2) Two students mark the remaining 20 documents respectively; 3) Two students respectively proofread the 60 documents marked by themselves with the tutor, and the proofreading was completed in three times, mainly discussing the existing problems and the strategies and methods of correction and annotation. On this basis, the two students annotated the whole corpus together.

In the annotation, we employ left to right segment and alignment method. Consistency is a major criterion of annotation quality. The alignment EDUs annotation evaluation should take into account the recognition consistency and alignment consistency. So, the consistency of Chinese annotation, English annotation, and Chinese-English alignment annotation of the two annotators are considered:

Chinese consistency: the consistency of two annotators on the same Chinese text.

English consistency: the consistency of two annotators on the same English text.

Chinese-English alignment consistency: the consistency of Chinese annotation on the same text by two annotators and the consistency of corresponding English alignment annotation.

We use Method1 and Method2 to compute the consistency.

Method1: computes the consistency of all possible annotations. There are punctuation marks at the recognition positions of Chinese EDUs, and punctuation marks that may be used as recognition marks. The recognition of EDUs in English is not based on punctuation, any space can be calculated as the recognition mark.

Method2: calculating the consistency of intersection (A ∩B) in all (A U B). Sentence Position="X1...X2 | Y1...Y2", calculate the case that A and B mark the same position of recognition. Compared with method 1, this method is more accurate and can unify the evaluation criteria of Chinese and English EDUs recognition.

	Chinese consistency	English consistency	Chinese-English alignment consistency
Method1	0.972	0.992	–
Method2	0.968	0.930	0.909

Table 1: The consistency of EDUs annotation for Chinese and English

As shown in Table 1, recognition alignment shows good consistency, with Chinese alignment up to 0.972/0.968, English alignment up to 0.992/0.930. Even under the strictest circumstances of Method2, Chinese-English alignment up to the consistency rate of 0.909.

It is worth noting under the Method1, English consistency is better than Chinese, with 0.992 >0.972. Under the Method2, Chinese better than English, this is because the consistency in the calculation, Chinese punctuation only for limited computation, but the English is for any Spaces.

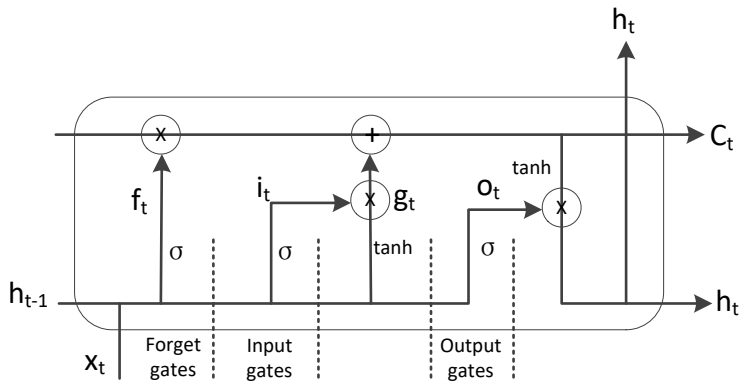


Figure 2: LSTM memory cell

However, the reality is that Chinese alignment is better than English with the same alignment evaluation criteria. This can be shown under Method2 (0.968 >0.930), because Chinese recognition is marked by punctuation, which is relatively easy. However, English recognition is not marked by punctuation, and it is easy to recognition incorrectly. Therefore, Method2 can more accurately reflect the difference in bilingual alignment effect compared with Method1.

3 The Model of EDU Recognition based on Bi-LSTM-CRF

In this section, we introduced the Bi-LSTM-CRF model we used, which is the combination of CRF and Bi-LSTM and have been used in several NLP task.

3.1 CRF

CRF is extension of both Hidden Markov Models and Maximum Entropy Model (Lafferty et al., 2001). It often solves some NLP problems, such as word recognition and image recognition. EDUs recognition is a sequence labeling problem. One solution is that it can assign each word in the sentence with label Y (word is EDU boundary) or N (word is not EDU boundary). CRF is a sequence labelling model with flexible feature space. Therefore, with given feature set and labeled training data, the CRF model solve EDUs recognition task. The model is defined as Eq. (1):

$$p(Y|X) = \frac{1}{Z(X)} exp(\sum_{k} \lambda_k f_k) \tag{1}$$

In Eq. (1), Z(X) is a probability normalization factor conditioned on X. λ_k is the corresponding weight of the feature set. f_k is the input sequence sentences, and Y is the output label of Y or N.

3.2 Bi-LSTM

RNN is a model suitable for sequence data, which uses previous and current state to determine the final output. However, in practical applications, RNN has only short-term memory because the gradient vanishing and exploding problem. Hochreiter and Schmidhuber(1997) propose LSTM network ,a variant of RNN to solve this problem.

Figure2 illustrates a single LSTM memory cell. We can see that it contains input, forget and output gate. The gates determine the current information, in a certain proportion or discarded, transferred to the next moment. Through the gate, LSTM can remove or add information to the cellular state. Therefore, they can solution the data long range dependencies problem.

LSTM memory cell is implemented as the Eq.(2): As shown in Eq. (2), the logistic sigmoid function is denoting as σ . it is the input gate. ct is cell vectors. i_t decides the information will be stored in c_t . ft is forgot gate, and it decides the information can through from the previous cell. o_t is output gate,

$$
\begin{aligned}
i_t &= (W_{ii}x_t + b_{ii} + W_{hi}h_{(t-1)} + b_{hi}) \\
f_t &= (W_{if}x_{t+} b_{if+} W_{hf}h_{(t-1)} + b_{hf}) \\
g_t &= \tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{(t-1)+} b_{hg}) \\
o_{t=} &(W_{io}x_t + b_{io} + W_{ho}h_{(t-1)} + b_{ho}) \\
c_t &= f_t c_{(t-1)} + i_t g_t \\
h_{t=} &o_t \tanh(c_t)
\end{aligned}
\qquad (2)
$$

it decides the information output to the current hidden state h_t. W is the weight matrix, and b is bias vectors of each gate. They are learned during training. $\oplus$ denotes the vector concatenation. For sequence tagging task, Graves and Jürgen utilize a bidirectional LSTM(Bi-LSTM) network. Bi-LSTM is extended on the basis of LSTM, and it contains two difference direction layers. The sequence $\overline{h} = (\overline{h_1 h_2}.....\overline{h_n})$ of the Bi-LSTM layer is obtained past and future input features by the forward and backward LSTM. The LSTM allows more context dependent information than LSTM.

3.3 Bi-LSTM-CRF

We describe our Bi-LSTM-CRF models in details. Figure 3 shows the Bi-LSTM-CRF framework. As we can see from Figure 3, there are input layer, embedding layer, Bi-LSTM layer, CRF layer and output layer. First, words in sentences and their features are vectorized. Secondly, the Bi-LSTM model is fed with feature vectors to learn contextual features from the forward and backward directions. Then, Bi-LSTM output result is input to CRF layer. Finally, the CRF layer predicts the globally optimal clause sequence. In addition, to reduce the influence of overfitting, we add a dropout layer at ends of the Bi-LSTM model.

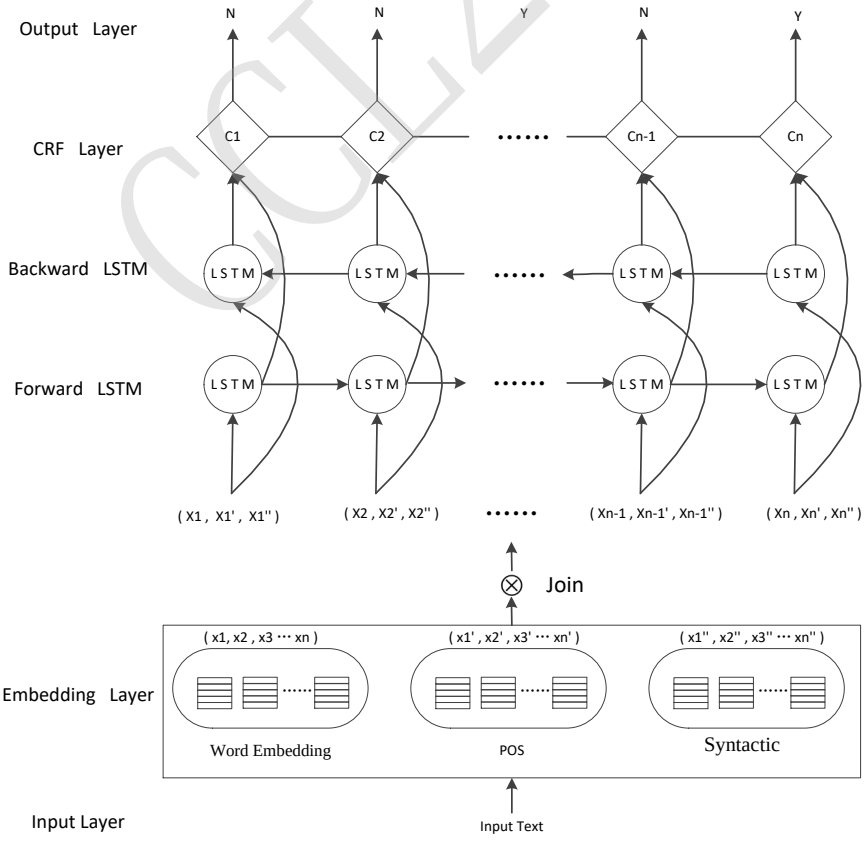


Figure 3: The framework of Bi-LSTM-CRF Model

Bi-LSTM-CRF expands the CRF layer on the basis of Bi-LSTM. The performance of CRF model in sequence annotation tasks has been verified. In this model, the Bi-LSTM through Bi-LSTM layer makes full use of past and future information, and CRF layer make use of tag in-formation. So, this model can predict the current tag by incorporate the advantage of Bi-LSTM and CRF.

4 Experiment Results

4.1 Experiment Setting

The input layers of our models are the input text. We give the vector representations of words, Part of speech (POS) and Syntactic. Word embeddings are pretrained using skip-n-gram, a variation of word2vec (Mikolov et al.,2013) that sensitive to the order of word. These embeddings are adjusted during training. We find improvements using pretrained word embeddings. For English, the embedding dimension we used is 200. For Chinese, we use pre-trained vector files from People's Daily News, the embedding dimension is 300(Li et al,2018). We use dropout training to avoid the model depending on one representation too strongly, and find it is import to result.

POS is the process of marking a word as nouns, verbs, adjectives, adverbs, etc. POS is used in many NLP task and proved very useful. Syntactic is the component that takes input sentence and give the grammatical tree structure of sentence, which is widely used to understanding written language, discourse parsing et al. For example, syntactic can output the phrases tag of words. We use Stanford coreNLP (Mikolov et al.,2013) to get POS and syntactic feature, it can give the POS and syntactic tag of each word. In this paper, we use parent phrase tag as syntactic feature simplify.

The task of EDUs recognition is giving a tag to every word in a sentence. A single EDU could span several words in a sentence. Sentences can represent in the Y(Yes) or N (No) format, where each word is labeled as Y label if the word is the end of EDU, and as N label if it is the beginning or inside of EDU word.

We exploit standard training methods for our model. Using AdaGrad(Duchi,2011) as stochastic gradient decent. Calculate derivatives from standard back propagation (Goller and Kuchler, 2002). In order to prevent over fitting, we regularize our model using dropout method (Srivastava et al.,2014), and fixed rate 0.5 for dropout layer. We obtain improvements after using dropout.

We set the initial AdaGrad learning rate as 0.01. The dimension of pre-trained word embed-ding is set as 200. The dimension of LSTM hidden state as 200.The W and b are randomly initialized with a uniform distribution in the range (-0.01, 0.01). We use publicly available 200-dimensional embeddings trained for English, there are total 40000 words. We use 300-dimensional embeddings for Chinese, there are total 355989 words. The Bi-LSTM units set 256, epoch set 200. In our experiment, for Chinese EDUs recognition, there are total 12 581 words, 32 POS tags and 29 syntactic tags. For English, there are total 4 106 words, 47 POS tags and 29 syntactic tags.

4.2 Experiment Results

In this section, EDU recognition is carried out in our Chinese-English alignment corpus. There are total of 100 documents, 513 paragraphs and 2 153 EDUs were involved. The recognition of English word is 42 122 in total, among which there are 2 153 positive labels. The ratio of positive and negative examples is 19.6:1 for English. Overall, the average length of the English EDUs is about 20 words, while the Chinese EDUs is 11 words. The experiment splits instances into 10 parts, and use 8 parts for training, 1 part for verification and 1 part for testing. The features we used are word embedding, POS tag and syntactic tag. The recognition results of Chinese words boundary are indicated in Table 2.

In Table 2, the best results are highlighted bold for each metric. From Table 2, we can see that by combining Bi-LSTM, pretrained embedding, and CRF on the top of the framework, our Bi-LSTM-CRF model outperforms best of all. We obtain the satisfactory results with the F1 93.4 % and R 94.4 % by using Bi-LSTM-CRF model.

Table 3 shows the English EDUs recognition result. For the purpose of comparison, we list Li's (Li et al, 2012) Maxent model results together with ours CRF and Bi-LSTM, especially our Bi-LSTM-CRF model results for comparison. The best F1 is 94.4% using Bi-LSTM-CRF model.

Model	P	R	F1
Li’ s Maxent	87.4	93.6	90.4
CRF	86.7	96	91.1
Bi-LSTM	**95.4**	89.8	92.5
Bi-LSTM-CRF	92.3	**94.4**	**93.4**

Table 2: Chinese EDU words boundary recognition results

Model	P	R	F1
Li’ s Maxent	86.5	78.7	82.4
CRF	87.4	91	89.1
Bi-LSTM	94.0	91.9	92.9
Bi-LSTM-CRF	**95.5**	**93.4**	**94.4**

Table 3: English EDU words boundary recognition results

Figure 4 comprise the result of F1 between Chinese and English for different models. We can see the best model is Bi-LSTM-CRF model, by joint decoding label sequence can benefit the final performance of neural network models, followed by Bi-LSTM and CRF. The reason is that EDUs recognition is sequence tag task, Bi-LSTM and CRF classifier perform better than traditional Maxent classifier.

Figure 4 shows that English EDUs recognition result is higher than Chinese using Bi-LSTM or Bi-LSTM-CRF, the reason is that the pretrained embedding of Chinese words are more than English, with Chinese 35 598 where English 4 000, the two is 10 times difference. But for using Maxent or CRF model, Chinese EDUs identification F1 is higher than English.

4.3 The contribution of features

In order to investigate the contribution of the features, we give experiments specifically targeted at features for EDUs recognition. Table 4 shows the performance of P, R, F1 for Chinese separately using different feature, and Table 5 gives the results of English.

Features	P	R	F1
Word Embedding	65.2	88.6	75.1
POS	70.1	80.2	74.8
Syntactic	81.1	82.1	81.6
Word Embedding +POS	76.7	90.7	83.1
Word Embedding +POS+ Syntactic	92.3	**94.4**	**93.4**

Table 4: The different feature result for Chinese

Table 4 and Table 5 show that syntactic feature outperform than other features, the F1 can reach 81.6% and 81.8% for Chinese and English. The reason is that both in Chinese and English, most EDU word syntactic labels contain IP and VP syntactic, while word with syntactic NP, PP and LCP are not EDU boundary. Syntactic information is highly related with EDUs recognition than other information. The combine of all features performance best both in Chinese and English, that means the more information used, the better the results.

POS is the commonly used in NLP task, from the results, we find it is also useful for EDU recognition. As shown in Table 5, only using word embedding feature, we can get F1 80.4% for English. We also find that word embedding feature is useful than syntactic feature for English, mainly because Chinese word is sparing. And Chinese EDUs boundary usually have punctuation, which have IP tag, so syntactic feature is useful than word embedding feature for Chinese.

According to the results, we know that using word embedding, POS and syntactic feature together, we can get best result, it proves the effectiveness of our features.

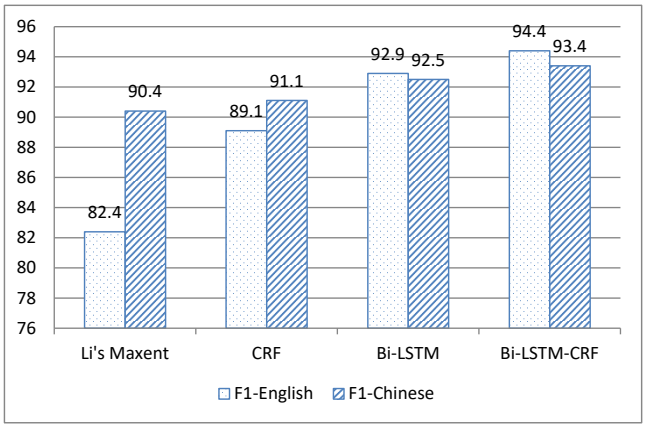


Figure 4: Comparison of F1 between Chinese and English for different models

Features	P	R	F1
Word Embedding	87.4	74.5	80.4
POS	71.2	79.8	75.3
Syntactic	80.2	83.5	81.8
Word Embedding +POS	90.4	87.1	88.7
Word Embedding +POS+ Syntactic	**95.5**	93.4	**94.4**

Table 5: The different feature result for English

4.4 Discussion

There are about 6% EDUs recognition error, and we discuss the reason as follows. There are two cases of errors: one is negative instances are recognized as positive instances. The other is positive instances are recognized as negative instances. From the recognition consistency compute method of section 2.3, we notice the punctuation plays an important role in EDUs recognition, especially in Chinese. For example, if the front words of comma are the subject of the sentence, therefore the position of this comma is not EDU boundary. But when using our model, the syntactic of the words is IP, which may lead to mistake recognition.

In EDUs recognition, it is difficult to distinguish EDUs from complex sentence structure. For example, if you believe that "在…以后，终于(In...After that, finally) "is a connective that expresses the relation of succession. It can be considered as an EDU. However, traditional grammar generally analyzes it as an adverbial, a part of the syntactic structure. This is transition between textual structure and syntactic structure. We currently follow the traditional grammar, leaving the analysis of this situation to the syntactic structure. For the automatic alignment of Chinese and English EDUs, we found that most of EDUs are sequence alignment, only about 4% of EDUs adjusted sequentially when from Chinese to English. So, for EDUs alignment, the main problem is EDUs recognition, which is influence on the result of automatic alignment EDUs. The difficulty of EDUs alignment is that EDUs does not correspond and adjust in order, which needs further research.

This paper only does Chinese and English EDUs recognition respectively, but does not do Chinese-English EDUs alignment. Once EDUs are identified, the next step is to align, and since EDUs are basically one-to-one, EDUs alignment can be turned into a machine translation or classification problem.

5 Related Work

Due to the emergence of discourse corpus, there have been a lot of researches on the recognition of English discourse. One of the corpora which are widely used is Rhetorical Structure Theory Discourse Treebank (RSTDT) building by Carlson et al. (2003), the other is Penn Discourse Treebank (PDTB) annotated by PDTB Research Group (2007). The RST represents a discourse as a tree, with phrases or clauses as EDU. PDTB adopts the predicate-argument view, with two spans as its arguments.

Due to the EDUs in RST consecutive annotation, the EDUs automatic identification on RSTDT is also called EDUs recognition, and now there is much research on it and the results are ideal, more representative research results include: Soricut and Marcus (2003) adopt statistics method for recognition, the F1 of EDUs recognition on the automatic syntax tree and standard syntax tree are 83.1% and 84.7%. Hernault et al. (2010) give a discourse recognition model based on sequential data annotation. They use lexical and syntactic features get the F1 94%, which is close to 98% of the F1 of manually. According to the above we can know that recogni-tion accuracy of EDUs on RSTDT is relatively high, and there is little room for further improvement. For the un-sequential annotation of arguments on PDTB, not all the discourse is covered. So, some researchers propose to replace the whole argument with the argument center in the recognition of argument (Wellner B. and Pustejovsky J,2007; Elwell R. and Baldridge J.,2008; Wellner B,2009). And other researches put forward to the point of identifying sentences that contain arguments (Prasad et al.,2010), the recognition accuracy of Arg1 and Arg2 are 65% and 85% (Xu F.,2013). Braund et al.(2017) research whether syntax help discourse segmentation, the results show that dependency information is less useful than expected, but they provide a fully scalable, robust model that only relies on part-of-speech information, and show that it performs well across languages in the absence of any gold-standard annotation.

Deep learning method has made breakthroughs in many NLP tasks in recent years. Among them, Cyclic Neural Network (RNN) is a typical sequence marking model, and it is proposed by Goller and Kuchler(1996). However, RNN is limited by gradient disappearance and gradient ex-plosion, Hochreiter and Schmidhuber (1997) come up with the variation of RNN which is named Long Short-Term Memory (LSTM). Because it only gets one-way contextual information, Graves and Schmidhuber (2005) raise the Bi-directional Long Short-Term Memory (Bi-LSTM), and applied it to speech identification. Bi-LSTM can effectively utilize past and future features in a specific time range. On the other hand, Conditional Random Field (CRF) algorithm which is put forward by Lafferty et al.(2001) has been widely applied in NLP recent years. In sequence marking tasks, CRF can take into account the anteroposterior dependence between adjacent la-bels of output. Considering the above reasons, there are some studies attempting to combine Bi-LSTM and CRF to build model for sequence data (Ji Me et al.,2018). Bi-LSTM and CRF hybrid model were first applied to the sequence labeling task of NLP by Huang et al. (2015), Ma and Hovy(2016) focus Bi-LSTM, CRF and CNN models and apply them to sequence marking tasks. Bi-LSTM-CRF model is applied in identifying biomedicine named entity (Greenberg et al.,2018), The effectiveness of the model in sequence marking tasks is gradually verified.

There are few discourse corpora in Chinese to mark EDU information (Zhang et al.2014;Li et al.,2014). At present, the task of EDU recognition is few referred. Zhang et al (2014) only identified the relation, but no relevant result about argument identification. Li et al. (2014) research on Chinese EDUs recognition based on comma, and Chinese EDUs recognition result can reach 90%. Ge Haizhu et al. (2019) proposes a Chinese EDU recognition approach based on theme-rheme theory, which can pay more attention on the internal structure of EDU, and the F1 score is 89.96%. However, limited by bilingual corpus, there is no EDUs recognition of both Chinese and English research.

6 Conclusion

The discourse alignment corpus of Chinese-English is annotated in this paper. The corpus has a complete EDU definition, annotation method, quality assurance and available scale. The corpus we annotated in this paper is the basic task of EDUs recognition. Then we developed an EDUs recognition system using Bi-LSTM-CRF model. Our neural model achieved satisfactory results for Chinese and English EDU recognition. To our knowledge, we are among the first to develop an effective neural network-

based approach to recognize EDUs for both Chinese and English. We input word embedding, POS and syntactic feature to our model in order to improve the result. By incorporating these features, our model can extract EDUs automatically and high quality. The F1 can reach 93.4% and 94.4% for Chinese and English separately, which is reaching the practical using. This model can also be generalized to solve other problems. In the future, we will improve the effect of recognition Chinese and English EDUs, then try to automatic align them.

Acknowledgements

This paper is supported by the National Natural Science Foundation of China (61502149), by the Open Research Fund of Key Laboratory of Advanced Theory and Application in Statistics and Data Science （East China Normal University）, Ministry of Education(KLATASDS1806), as well as the high-level talent research project of Henan Institute of Science and Technology (2017039).

References

Braud C., Lacroix O., and Anders S. 2017. *Does syntax help discourse segmentation? Not so much.* Conference on Empirical Methods in Natural Language Processing, 2432 – 2442.

Carlson, L.， Marcu, D.， and Okurowski, M. E. 2003. *Building a discourse-tagged corpus in the framework of Rhetorical Structure Theory.* Current and New Directions in Discourse and Dialogue. Springer Netherlands.

Duchi J., Hazan E. , and Singer Y. 2011. *Adaptive Subgradient Methods for Online Learning and Stochastic Optimization.* Journal of Machine Learning Research, 12(7):257-269.

Elwell R. and Baldridge J. 2008. *Discourse connective argument identification with connective specific rankers.* In IEEE International Conference on Semantic Computing, 198-205.

Feng W.H. 2013. *Alignment and Annotation of Chinese-English Discourse Structure Parallel Corpus.* Journal of Chinese Information Processing, 27(6):158-165.

Ge H.Z., Kong F., and Zhou G.D. 2019. *Chinese Elementary Discourse Unit Recognition Based on Theme-Rheme Theory.* Journal of Chinese Information Processing,33(8):20-27.

Goller C., Kuchler A. 1996. *Learning Task-Dependent Distributed Representations by Backpropagation Through Structure.* IEEE International Conference on Neural Networks,347 – 352.

Graves A., Schmidhuber J. 2005. *Framewise phoneme classification with bidirectional LSTM and other neural network architectures.* Neural Networks, 18(5):602-610.

Greenberg N., Bansal T., Verga P., and McCallum A. 2018. *Marginal Likelihood Training of BiLSTM-CRF for Biomedical Named Entity Recognition from Disjoint Label Sets.* In Proceedings of the Conference on Em-pirical Methods in Natural Language Processing, 2824 – 2829.

Hernault H., Bollegala D., and Ishizuka M. 2010. *A Sequential Model for Discourse Recognition.* In Computational Linguistics and Intelligent Text Processing, Springer, Berlin, Heidelberg, 2010, 315-326.

Hochreiter S., Schmidhuber J. 1997. *Long Short-Term Memory.* Neural Computation, 9(8):1735-1780.

Huang Z., Xu W., and Yu K. 2015. *Bidirectional LSTM-CRF Models for Sequence Tagging.* Computation and Language, 2015.

Ji M., Kuzman G. and David W. 2018. *State-of-the-art Chinese Word Recognition with Bi-LSTMs.* In Proceedings of the Conference on Empirical Methods in Natural Language Processing,4902 – 4908.

Lafferty J., Mccallum A., and Pereira F. 2001. *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data.* In Proceedings of the Eighteenth International Conference on Machine Learning , Morgan Kaufmann Publishers Inc, 282-289.

Li S.,Zhao Z. Hu R. et al. 2018. *Analogical Reasoning on Chinese Morphological and Semantic Relations.* In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, 138-143.

Li Y.C., Feng W.H., Sun J., et al. 2014. *Building Chinese Discourse Corpus with Connective-driven Dependency Tree Structure.* In proceedings of Empirical Methods in Natural Language Processing, 2105-2114.

Li Y.C., Feng W.H., Zhou G.D., et al. 2013. *Research of Chinese Clause Identification Based on Comma.* Acta Scientiarum Naturalium Universitatis Pekinensis, 49(1):7-14.

Ma X. and Hovy E. 2016 . *End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF.* In Proceedings of the Meeting of the Association for Computational Linguistics, 1064-1074.

Manning C. D., Mihai S., John B. et al. 2014. *The Stanford CoreNLP Natural Language Processing Toolkit.* In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 55-60.

Mikolov T., Sutskever I., Chen K., et al. 2013. *Distributed representations of words and phrases and their compositionality.* Advances in Neural Information Processing Systems, 26:3111-3119.

PDTB Research Group. 2007. *The Penn discourse Treebank 2.0 annotation manual.* IRCS Technical Reports Series.

Prasad R., Joshi A. K., and Webber B. L. 2010. *Exploiting Scope for Shallow Discourse Parsing.* In Proceedings of the Seventh International Conference on Language Resources and their Evaluation,Valletta, Malta, 2076-2083.

Soricut R. and Marcus D. 2003. *Sentence Level Discourse Parsing using Syntactic and Lexical Information.* In Proceedings of the 2003 Conference of the North American,149-156.

Wellner B. and Pustejovsky J. 2007. *Automatically Identifying the Arguments of Discourse Connectives.* In EMNLP-CoNLL, 92-101.

Srivastava N., Hinton G., Krizhevsky A., et al. 2014. *Dropout: A simple way to prevent neural networks from overfitting.* The Journal of Machine Learning Research, 15(1):1929 – 1958.

Wellner B. 2009. *Sequence models and ranking methods for discourse parsing.* Faculty of the Graduate School of Arts and Sciences Brandeis University Computer Science James Pustejovsky, Brandeis University.

Xu F. 2013. *Research of Key Issues in English Discourse Structure Analysis.* Soochow university.

Zhang M.Y., Qin B., and Liu T. 2014. *Chinese Discourse Relation Semantic Taxonomy and Annotation.* Journal of Chinese Information Processing, 28(2):28-36.

Better Queries for Aspect-Category Sentiment Classification

Yuncong Li [1,*], **Cunxiang Yin**[1,*], **Sheng-hua Zhong**[2,†], **Huiqiang Zhong**[1], **Jinchang Luo**[1], **Siqi Xu**[1] and **Xiaohui Wu**[1]
[1]Baidu Inc., Beijing, China
[2]College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China
[1]{liyuncong,yincunxiang,zhonghuiqiang,luojinchang,xusiqi01,wuxiaohui02}@baidu.com
[2]csshzhong@szu.edu.cn

Abstract

Aspect-category sentiment classification (ACSC) aims to identify the sentiment polarities towards the aspect categories mentioned in a sentence. Because a sentence often mentions more than one aspect category and expresses different sentiment polarities to them, finding aspect category-related information from the sentence is the key challenge to accurately recognize the sentiment polarity. Most previous models take both sentence and aspect category as input and query aspect category-related information based on the aspect category. However, these models represent the aspect category as a context-independent vector called aspect embedding, which may not be effective enough as a query. In this paper, we propose two contextualized aspect category representations, Contextualized Aspect Vector (CAV) and Contextualized Aspect Matrix (CAM). Specifically, we use the coarse aspect category-related information found by the aspect category detection task to generate CAV or CAM. Then the CAV or CAM as queries are used to search for fine-grained aspect category-related information like aspect embedding by aspect-category sentiment classification models. In experiments, we integrate the proposed CAV and CAM into several representative aspect embedding-based aspect-category sentiment classification models. Experimental results on the SemEval-2014 Restaurant Review dataset and the Multi-Aspect Multi-Sentiment dataset demonstrate the effectiveness of CAV and CAM.

1 Introduction

Sentiment analysis (Pang and Lee, 2008; Liu, 2012) is an important task in Natural Language Processing (NLP). It deals with the computational treatment of opinion, sentiment, and subjectivity in text. Aspect-based sentiment analysis (Pontiki et al., 2014; Pontiki et al., 2015; Pontiki et al., 2016) is a branch of sentiment analysis and aspect-category sentiment analysis (ACSA) is a subtask of it. In ACSA, there are a predefined set of aspect categories, and a predefined set of sentiment polarities. Given a sentence, the task aims to predict the aspect categories mentioned in the sentence and the corresponding sentiments. Therefore, ACSA contains two subtasks: aspect category detection (ACD) that detects aspect categories in a sentence and aspect-category sentiment classification (ACSC) that categorizes the sentiment polarities with respect to the detected aspect categories. Figure 1 shows an example, "Staffs are not that friendly, but the taste covers all.". ACD detects the sentence mentions two aspect categories: *service* and *food*, and ACSC predicts the sentiment polarities to them: negative and positive respectively. In this work, we focus on ACSC, while ACD as an auxiliary task is used to find coarse aspect category-related information for the ACSC task.

Because a sentence often mentions more than one aspect category and expresses different sentiment polarities to them, to accurately recognize the sentiment polarities, most previous models (Ruder et al., 2016; Wang et al., 2016; Cheng et al., 2017; Li et al., 2017; Tay et al., 2018; Xue and Li, 2018; Xing et al., 2019; Liang et al., 2019b; Jiang et al., 2019; Hu et al., 2019; Wang et al., 2019) take both sentence and aspect category as input and query aspect category-related information based on the aspect

[*]Equal contribution.
[†]Corresponding author.

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1079-1088, Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

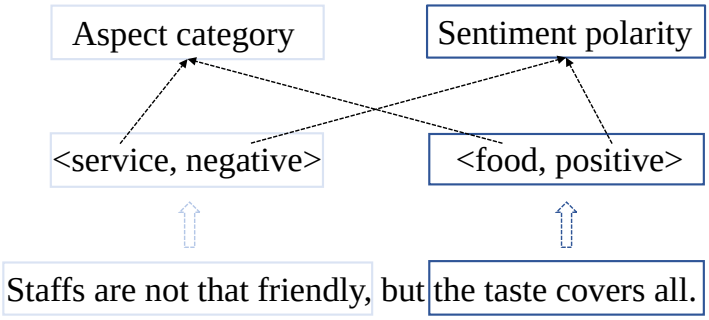


Figure 1: An example of aspect-category sentiment analysis.

category, then generate aspect category-specific representations for aspect-category sentiment classification. However, these models represent the aspect category as a context-independent vector called aspect embedding (AE). These models can be called aspect embedding-based models. Since aspect embedding only contains the global information of aspect category and loses the context-dependent information, it is semantically far away from the words in the sentence, and may not be effective enough as a query to search for aspect category-related information for the ACSC task. These models may be improved by replacing the aspect embedding with context-dependent aspect category representations.

The HiErarchical ATtention (HEAT) network (Cheng et al., 2017) used context-dependent aspect category representations to search for aspect category-related information for the ACSC task and obtained better performance. The context-dependent aspect category representations are generated by concatenating the aspect embedding and the aspect term representation in a sentence. An aspect term is a word or phrase that appears in the sentence explicitly indicating an aspect category. For the example in Figure 1, the aspect terms are "Staffs" and "taste" indicating aspect category *service* and *food* respectively. However, the HEAT network requires aspect term annotation information that the data for ACSC usually does not have. Moreover, the HEAT network ignores the situation where the aspect category is mentioned implicitly in sentences without any aspect term, making aspect category representations degenerate to context-independent representations in this situation.

In this paper, we propose two novel contextualized aspect category representations, Contextualized Aspect Vector (CAV) and Contextualized Aspect Matrix (CAM). CAV or CAM contain context-dependent information even though there are no aspect terms in sentences, and aspect term annotation information is not required to generate them. Concretely, we use the coarse aspect category-related information found by the ACD task to generate CAV or CAM. Then CAV or CAM as queries are used to search for fine-grained aspect category-related information like aspect embedding by aspect-category sentiment classification models. Specifically, we first use an attention-based aspect category classifier to obtain the weights of the words in a sentence, which indicate the degree of correlation between the aspect categories and the words. Then, we get CAV by combining the weighted sum of the word representations with corresponding aspect embedding. That is to say, CAV contains two kinds of representations of an aspect category: context-independent representation and context-dependent representation, which capture global information and local information respectively. Since CAV may lose details of the words, we also propose an aspect category matrix representation, called Contextualized Aspect Matrix (CAM), which is a not-sum version of CAV.

In summary, the main contributions of our work can be summarized as follows:

- We propose two novel contextualized aspect category representations, Contextualized Aspect Vector (CAV) and Contextualized Aspect Matrix (CAM). They include the global information and local information about the aspect category and are better queries to search for aspect category-related information for aspect category sentiment classification (ACSC). To the best of our knowledge, it is the first time to represent aspect category as matrix.

- We experiment with several representative aspect embedding-based models by replacing the as-

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1079-1088,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

pect embedding with CAV or CAM. Experimental results on the SemEval-2014 Restaurant Review dataset and the Multi-Aspect Multi-Sentiment dataset demonstrate the effectiveness of CAV and CAM.

2 Related Work

In this section, we first present a brief review about aspect-category sentiment classification. Then, we show the related study on context-aware aspect embedding that is a kind of context-dependent aspect category representation for targeted aspect based sentiment analysis (TABSA).

2.1 Aspect-Category Sentiment Classification

Many models (Wang et al., 2016; Ruder et al., 2016; Cheng et al., 2017; Li et al., 2017; Tay et al., 2018; Schmitt et al., 2018; Xue and Li, 2018; Xing et al., 2019; Liang et al., 2019b; Jiang et al., 2019; Hu et al., 2019; Wang et al., 2019; Sun et al., 2019) have been proposed for the aspect-category sentiment classification (ACSC) task. Wang et al. (2016) proposed an attention-based LSTM network for aspect-level sentiment classification. Tay et al. (2018) introduced a word-aspect fusion attention layer to attend based on associative relationships between sentence words and aspect categories. Xue et al. (2018) proposed to extract sentiment features with convolutional neural networks and selectively output aspect category related features for classification with gating mechanisms. Xing et al. (2019) proposed a novel variant of LSTM, which incorporates aspect information into LSTM cells in the context modeling stage. Liang et al. (2019b) proposed a novel Aspect-Guided Deep Transition model, which utilizes the given aspect category to guide the sentence encoding from scratch. Jiang et al. (2019) proposed new capsule networks to model the complicated relationship between aspects and contexts. To force the orthogonality among aspect categories, Hu et al. (2019) proposed constrained attention networks (CAN) for multi-aspect sentiment analysis. To avoid error propagation, some joint models (Li et al., 2017; Schmitt et al., 2018; Wang et al., 2019) have been proposed, which perform aspect category detection (ACD) and aspect-category sentiment classification (ACSC) jointly. Li et al. (2017) proposed an end-to-end machine learning architecture, in which the ACD task and the ACSC task are interleaved by a deep memory network. Wang et al. (2019) proposed the aspect-level sentiment capsules model (AS-Capsules), which utilizes the correlation between aspect and sentiment through shared components including capsule embedding, shared encoders, and shared attentions. The capsule embedding is similar to the aspect embedding. All these models represented aspect category as context-independent representations, which may benefit from CAV or CAM.

Closely related to our method is the HiErarchical Attention (HEAT) network proposed by Cheng et al. (2017), in which an aspect attention extracts the aspect term information, and then a context-dependent aspect category representation generated based on the aspect term information is used to guide the sentiment attention to better allocate aspect-specific sentiment words of the text. However, extracting aspect term information requires additional aspect term annotation information. In addition, HEAT ignores the situation where the aspect category is mentioned implicitly in texts. There are also some models that don't rely on aspect embedding. Schmitt et al. (2018) also proposed a joint model, in which different aspect categories have different sentiment classifiers to generate aspect category-specific representations. Sun et al. (2019) constructed an auxiliary sentence from the aspect and converted ABSA to a sentence-pair classification task.

2.2 Context-aware Aspect Embedding

Context-aware aspect embedding is a kind of context-dependent aspect category representation (Liang et al., 2019a). Liang et al. (2019a) proposed an embedding refinement method to generate context-aware target embedding and aspect embedding for targeted aspect based sentiment analysis (TABSA) (Saeidi et al., 2016), which utilizes a sparse coefficient vector to adjust the embeddings of target and aspect from the context and yields the state-of-the-art performance in this task. However, their method relies on context-aware target embedding to generate aspect embedding, and can't be applied in the ACSC task directly.

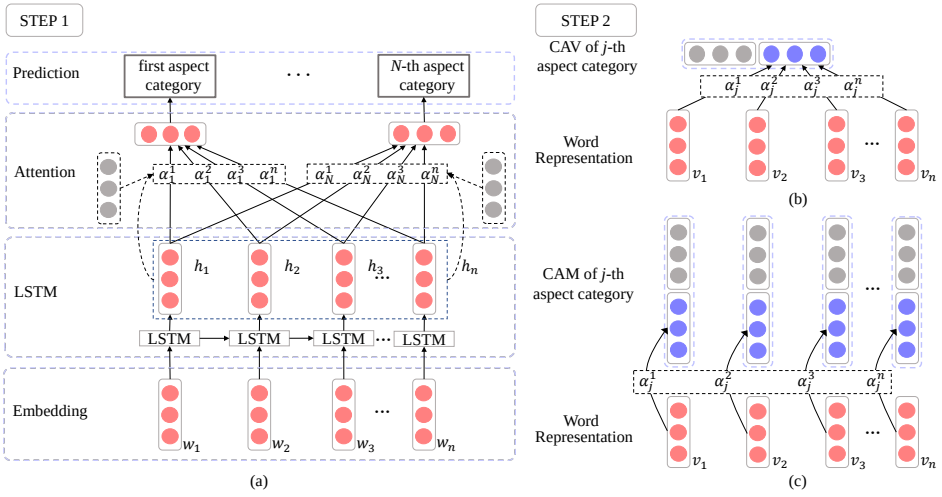


Figure 2: (a) shows the attention-based aspect category classifier, which generates the weights of the words in a sentence about all predefined aspect categories. (b) and (c) show how to generate CAV and CAM based on the weights and the original representations of the words respectively.

3 Method

In this section, we describe our proposed two contextualized aspect category representations, Contextualized Aspect Vector (CAV) and Contextualized Aspect Matrix (CAM), in detail.

Motivated by the process that people search for information through search engines: before finding the result they want, they usually try different words and adjust their queries based on previous results, the process to generate CAV or CAM consists of two steps. In the first step, the ACD task as an auxiliary task is used to find coarse aspect category-related information. In the second step, the coarse aspect category-related information is used to optimize original query (e.g. aspect embedding). Specifically, an attention-based aspect category classifier generates the weights of the words in a sentence about all predefined categories. Then the weights are used to generate CAV and CAM. The framework of our proposed method is demonstrated in Figure 2.

3.1 Coarse Aspect Category-related Information

In this step, the ACD task is used to find coarse aspect category-related information. It is a multi-label classification problem, and can be formulated as follows. There are N predefined aspect categories $A = \{A_1, A_2, ..., A_N\}$ in the dataset. Given a sentence, denoted by $S = \{w_1, w_2, ..., w_n\}$, the task checks each aspect $A_j \in A$ to see whether the sentence S mentions it.

An attention-based aspect category classifier is used for this task, because it can offer the weights of the words in a sentence about all predefined categories indicating which word is related to which aspect category. The overall architecture of the model is illustrated in Figure 2 (a). The model contains four modules: embedding layer, LSTM layer, attention layer, and aspect category prediction layer. All aspect categories share the embedding layer and the LSTM layer, and different aspect categories have different attention layers and prediction layers.

Embedding Layer: The input of this layer is a sentence consisting of n words $\{w_1, w_2, ..., w_n\}$. With an embedding matrix U, the input sentence is converted to a sequence of vectors $X = \{x_1, x_2, ..., x_n\}$, where $U \in R^{d \times |V|}$, d is the dimension of the word embeddings, and $|V|$ is the vocabulary size.

LSTM Layer: The word embeddings of the sentence are then fed into a LSTM (Hochreiter and Schmidhuber, 1997) layer, which outputs hidden states $H = \{h_1, h_2, ..., h_n\}$. At each time step i, the hidden state h_i is computed by:

$$h_i = LSTM(h_{i-1}, x_i) \quad (1)$$

The size of the hidden state is also set to be d.

Attention Layer: This layer takes the output of the LSTM layer as input, and produce an attention (Yang et al., 2016) weight vector for each predefined aspect category. Formally, for the j-th aspect category:

$$M_j = tanh(W_j H + b_j) \tag{2}$$

$$\alpha_j = softmax(u_j^T M_j) \tag{3}$$

where $W_j \in R^{d \times d}$,$b_j \in R^d$,$u_j \in R^d$ are learnable parameters, and $\alpha_j \in R^n$ is the attention weight vector. **We can see u_j as aspect embedding, which is the initial query for aspect category-related information.**

Aspect Category Prediction Layer: We use the weighted hidden state as the sentence representation for ACD prediction. For the j-th category:

$$r_j = H\alpha_j^T \tag{4}$$

$$\hat{y}_j = sigmoid(W_j r_j + b_j) \tag{5}$$

where $W_j \in R^{d \times 1}$ and $b_j \in R$.

Loss: As each prediction is a binary classification problem, the loss function for the N aspect categories of the sentence is defined by:

$$L(\theta) = -\sum_{j=1}^{N} y_j log\hat{y}_j + (1 - y_j) log(1 - y_j) + \lambda ||\theta||_2^2 \tag{6}$$

where y_j is the correct label, λ is the L_2 regularization factor, N is the number of total aspect categories and θ contains all the parameters.

3.2 Context-dependent Aspect Category Representations

In this step, the attention weight vectors offered by the ACD task is used to generate contextualized Aspect Vector (CAV) and Contextualized Aspect Matrix (CAM). **They are the results of optimizing the initial query based on context-dependent information**. Figure 2 (b) and Figure 2 (c) show how to generate CAV and CAM respectively. Given a sentence representation $V = \{v_1, v_2, \ldots, v_n\}$ from an ACSC model and the attention weight vectors of all predefined aspect categories offered by the ACD task, CAV of the j-th aspect category is computed by:

$$v_{CAV_j} = [v_{CAVG_j}; v_{CAVL_j}] \tag{7}$$

$$v_{CAVL_j} = \sum_{i=1}^{n} v_i \alpha_j^i \tag{8}$$

where $v_i \in R^{d_l}$ and d_l is the dimension of the word representations, $v_{CAVG_j} \in R^{d_g}$ and $v_{CAVL_j} \in R^{d_l}$ are the global representation and the local representation respectively, d_g is the dimension of the global aspect category representation, v_{CAVG_j} is initialized randomly and learned during training ACSC models like aspect embedding, and α_j^i indicates the weight of the i-th word about the j-th aspect category. V can be the output of the embedding layer or the sentence encoder in ACSC models.

Because the aspect category representation vectors, such as aspect embedding, often are repeated as many times as there are words in the sentence and concatenated to the word representations of the sentence, we also propose the Contextualized Aspect Matrix (CAM), which can be directly concatenated to the word representations and retains more details of the words. For the j-th aspect category, M_{CAM_j} is computed by:

$$M_{CAM_j} = \{[v_{CAVG_j}; v_1\alpha_j^1], [v_{CAVG_j}; v_2\alpha_j^2], ..., [v_{CAVG_j}; v_n\alpha_j^n]\} \tag{9}$$

where v_{CAVG_j} is the same as it in CAV.

Then the CAV or CAM as queries are used to search for fine-grained aspect category-related information like aspect embedding by ACSC models. Figure 3 shows how to integrate CAV and CAM into AT-LSTM (Wang et al., 2016).

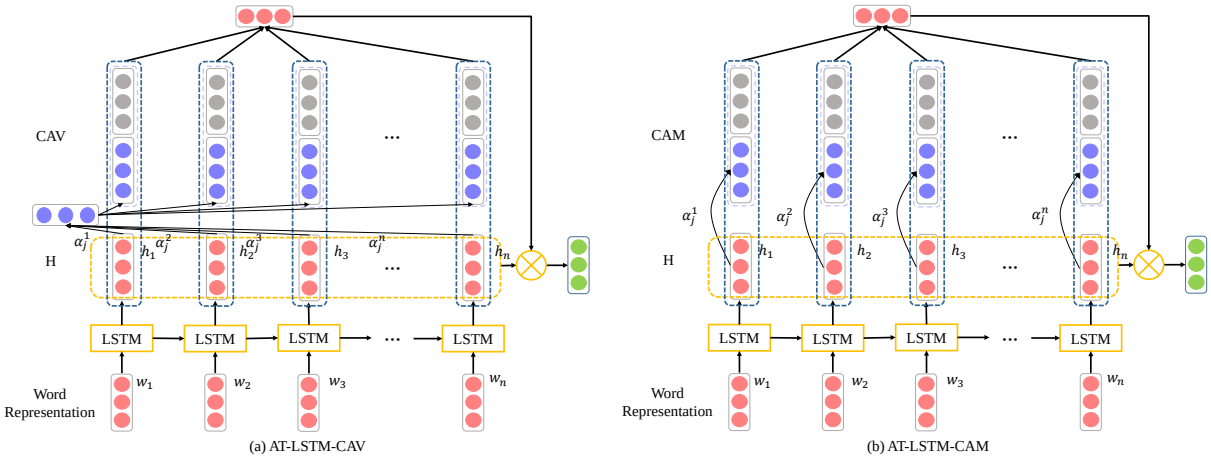


Figure 3: AT-LSTM-CAV and AT-LSTM-CAM, which are obtained by replacing the aspect embedding in AT-LSTM (Wang et al., 2016) with CAV and CAM respectively.

4 Experiments

4.1 Datasets

In order to evaluate the effectiveness of our methods, we conduct experiments on the SemEval-2014 Restaurant Review (Restaurant-2014) dataset (Pontiki et al., 2014) and the Multi-Aspect Multi-Sentiment for Aspect Category Sentiment Analysis (MAMS-ACSA) dataset (Jiang et al., 2019). The Restaurant-2014 is a widely used dataset. However, most sentences in Restaurant-2014 contain only one aspect category or multiple aspect categories with the same sentiment polarity, which makes ABSA task degenerate to sentence-level sentiment analysis. To mitigate the problem, Jiang et al. (2016) released the MAMS-ACSA dataset, all sentences in which contain multiple aspects with different sentiment polarities. Since there is no official development set for the Restaurant-2014 dataset, we use the split offered by Xue et al. (2018). Statistics of these two datasets are given in Table 1.

Dataset		Positive	Negative	Neutral	Total
Restaurant-2014	Train	1855	733	430	3018
	Validation	324	106	70	500
	Test	657	222	94	973
MAMS-ACSA	Train	1929	2084	3077	7090
	Validation	241	259	388	888
	Test	245	263	393	901

Table 1: Statistics of the datasets.

4.2 Implementation Details

We implement our models in PyTorch (Paszke et al., 2017). For all models, including the aspect category classifier and the aspect-category sentiment classification models, we use the pre-trained 300d Glove embeddings (Pennington et al., 2014) to initialize word embeddings, which is fixed in all models. We use Adam optimizer (Kingma and Ba, 2014) with learning rate 0.001 to train all models. We set L_2 regularization factor $\lambda = 0.00001$. The batch sizes are set to 32 and 64 for the Restaurant-2014 dataset and the MAMS-ACSA dataset respectively. For CAV and CAM, d_g is equivalent to d_l. For the aspect category sentiment classification models, we replace the aspect embedding with the CAV or CAM, just adjust the parameters to make the dimensions matching, and use hyper-parameter settings described in original papers. The aspect category classifier and the aspect-category sentiment classification models are trained in a pipeline manner. That is to say, the aspect category classifier is first trained, then the

aspect-category sentiment classification models are trained, where the attention weights offered by the aspect category classifier are used to generate CAV or CAM. We fine-tune the hyper-parameters for all baselines on the validation set. We run all models for 5 times and report the average results on the test datasets.

4.3 Comparison Methods

We select the following methods as baseline models:

AE-LSTM (Wang et al., 2016) first get the aspect-aware sentence embedding by concatenating the aspect embedding with each word embedding. Then the aspect-aware sentence embedding is fed into a LSTM layer. The final sentence representation is the last hidden state of the LSTM layer.

AT-LSTM (Wang et al., 2016) models the sentence via a LSTM model. Then it combines the hidden states from the LSTM with the aspect embedding to generate the attention vector. The final sentence representation is the weighted sum of the hidden states.

ATAE-LSTM (Wang et al., 2016) further extends AT-LSTM by taking the aspect-aware sentence embedding as input.

CapsNet (Jiang et al., 2019) is a capsule network that can model the complicated relationship between aspect categories and contexts and obtains state-of-the-art performance on the MAMS-ACSA dataset. It also takes the aspect-aware sentence embedding as input.

Our methods:

***-CAV** replace the aspect embedding in the baseline models with CAV.

***-CAM** replace the aspect embedding in the baseline models with CAM

Method	Restaurant-2014	MAMS-ACSA
AE-LSTM	76.876 (±2.037)	63.019 (±2.318)
AE-LSTM-CAV	76.711 (±0.963)	66.970 (±0.824)
AE-LSTM-CAM	80.493 (±1.422)	70.721 (±0.717)
AT-LSTM	77.9∗	66.436†
AT-LSTM-CAV	**81.891 (±0.493)**	73.052 (±1.551)
AT-LSTM-CAM	80.740 (±0.681)	**75.539 (±0.657)**
ATAE-LSTM	77.8∗	70.634†
ATAE-LSTM-CAV	81.172 (±0.398)	73.141 (±1.499)
ATAE-LSTM-CAM	81.829 (±0.784)	73.452 (±1.217)
CapsNet	81.110 (±0.492)	73.986†
CapsNet-CAV	77.246 (±0.696)	69.700 (±0.659)
CapsNeT-CAM	80.417 (±0.558)	75.117 (±0.203)

Table 2: Results of the ACSC task in terms of accuracy (%). "∗" refers to citing from Tay et al. (2018). "†" refers to citing from Jiang et al.(2019). Best scores are marked in bold.

4.4 Results and Analysis

Experimental results are illustrated in Table 2. From Table 2 we draw the following conclusions. First, we observe that most models with CAV obtain better performance. Specifically, by replacing the aspect embedding with CAV, our proposed methods outperform their counterparts in 5 of 8 results. compared original models, AT-LSTM-CAV and ATAE-LSTM-CAV improves the performance by 3.9% and 3.4% on the Restaurant-2014 dataset respectively. AE-LSTM-CAV, AT-LSTM-CAV and ATAE-LSTM-CAV improves the performance by 3.9%, 6.6% and 2.5% on the MAMS-ACSA dataset respectively. In addition, AT-LSTM-CAV obtains the best performance on Restaurant-2014. Second, most models with CAM also obtain better performance. Specifically, by replacing the aspect embedding with CAM, most of our proposed methods outperform their counterparts. AE-LSTM-CAM, AT-LSTM-CAM and ATAE-LSTM-CAM improves the performance by 3.6%, 2.8% and 4% on the Restaurant-2014 dataset, by 7.7%, 9.1% and 2.8% on the MAMS-ACSA dataset, respectively. AT-LSTM-CAM and CapsNeT-CAM surpass the

Sentence id	Aspect category	Attention weights
1	food	0.06 0.07 0.63 0.22 I go to Sushi Rose for fresh **sushi** and great **portions** all at a reasonable price
	price	1.00 I go to Sushi Rose for fresh sushi and great portions all at a reasonable **price**
2	food	0.04 0.93 Staffs are not that friendly, but the **taste** covers all.
	service	0.99 **Staffs** are not that friendly, but the taste covers all
3	price	0.99 I thought the food isn't cheap at all compared to Chinatown

Figure 4: Visualization of attention weights of different aspect categories in the ACD task. The numbers on the top of words are the attention weights of the words. The weights greater than 0.01 are labeled. The bold words are the labeled aspect terms. The color depth expresses the important degree of the word.

state-of-the-art baseline mode CapsNeT (+1.6% and +1.1% respectively) on the MAMS-ACSA dataset. Third, CAM outperform CAV in 7 of 8 results. This is because CAM retains more details of the words. Finally, we observe that, in 4 of 6 results, CAV leads to performance drop when aspect category sentiment classification models use it to get aspect-aware sentence embedding by concatenating it with each word embedding. Specifically, compared to AE-LSTM, AT-LSTM-CAV and CapsNet, AE-LSTM-CAV, ATAE-LSTM-CAV and CapsNet-CAV reduce by 0.2%, 0.7% and 4.6% on the Rest14 dataset. Compared to CapsNet, CapsNet-CAV reduces by 4.2% on the MAMS-ACSA dataset. The possible reason is that, in this situation, every word representation contains all aspect category-related information of the sentence, which leads to the sentence encoder, such as LSTM (Hochreiter and Schmidhuber, 1997), to concentrate on the aspect category-related information and discard the aspect category-related sentiment information. It suggests that CAV be best used in attention mechanisms.

4.5 Attention Visualizations

Figure 4 displays the performance of the attention to find aspect category-related words for the ACSC task. Sentence 1 shows that the attention can find the aspect terms for different aspect categories obviously. In sentence 2, while the aspect term for the aspect category *service* is "taste", the attention finds "friendly" that is more useful than "taste" for the ACSC task. The sentence 3 don't have any aspect term for the aspect category *price*, however, the attention also finds the useful word "cheap".

5 Conclusion

In this paper, we propose two novel contextualized aspect category representations, Contextualized Aspect Vector (CAV) and Contextualized Aspect Matrix (CAM). They include both the global information and local information about the aspect category and are better queries to search for aspect category-related information for the ACSC task. Moreover, CAV or CAM contain context-dependent information even though there are no aspect terms in sentences, and aspect term annotation information is not required to generate them. We experiment with several representative aspect embedding-based models by replacing the aspect embedding with CAV or CAM. Experimental results on the SemEval-2014 Restaurant dataset and the Multi-Aspect Multi-Sentiment (MAMS) dataset show that the variants with CAV or CAM obtain better performance. In future works, we will explore the performance of CAV and CAM with knowledge from open knowledge graphs on the ACSC task.

References

Jiajun Cheng, Shenglin Zhao, Jiani Zhang, Irwin King, Xin Zhang, and Hui Wang. 2017. Aspect-level sentiment classification with heat (hierarchical attention) network. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 97–106.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Mengting Hu, Shiwan Zhao, Li Zhang, Keke Cai, Zhong Su, Renhong Cheng, and Xiaowei Shen. 2019. Can: Constrained attention networks for multi-aspect sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4593–4602.

Xiaotian Jiang, Quan Wang, Peng Li, and Bin Wang. 2016. Relation extraction with multi-instance multi-label convolutional neural networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1471–1480.

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6281–6286.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Cheng Li, Xiaoxiao Guo, and Qiaozhu Mei. 2017. Deep memory networks for attitude identification. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 671–680.

Bin Liang, Jiachen Du, Ruifeng Xu, Binyang Li, and Hejiao Huang. 2019a. Context-aware embedding for targeted aspect-based sentiment analysis. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4678–4683, Florence, Italy, July. Association for Computational Linguistics.

Yunlong Liang, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2019b. A novel aspect-guided deep transition model for aspect based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5572–5584.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, August. Association for Computational Linguistics.

Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 486–495.

Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, AL-Smadi Mohammad, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphee De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30.

Sebastian Ruder, Parsa Ghaffari, and John G Breslin. 2016. A hierarchical model of reviews for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 999–1005.

Marzieh Saeidi, Guillaume Bouchard, Maria Liakata, and Sebastian Riedel. 2016. SentiHood: Targeted aspect based sentiment analysis dataset for urban neighbourhoods. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1546–1556, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Martin Schmitt, Simon Steinheber, Konrad Schreiber, and Benjamin Roth. 2018. Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1109–1114.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 380–385.

Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615.

Yequan Wang, Aixin Sun, Minlie Huang, and Xiaoyan Zhu. 2019. Aspect-level sentiment analysis using as-capsules. In *The World Wide Web Conference*, pages 2033–2044.

Bowen Xing, Lejian Liao, Dandan Song, Jingang Wang, Fuzheng Zhang, Zhongyuan Wang, and Heyan Huang. 2019. Earlier attention? aspect-aware lstm for aspect sentiment analysis. *arXiv preprint arXiv:1905.07719*.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2514–2523.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

Multimodal Sentiment Analysis with Multi-perspective Fusion Network Focusing on Sense Attentive Language

Xia Li* , Minping Chen
Guangzhou Key Laboratory of Multilingual Intelligent Processing,
School of Information Science and Technology,
Guangdong University of Foreign Studies,Guangzhou,China
{xiali,minpingchen}@gdufs.edu.cn

Abstract

Multimodal sentiment analysis aims to learn a joint representation of multiple features. As demonstrated by previous studies, it is shown that the language modality may contain more semantic information than that of other modalities. Based on this observation, we propose a Multi-perspective Fusion Network(MPFN) focusing on Sense Attentive Language for multimodal sentiment analysis. Different from previous studies, we use the language modality as the main part of the final joint representation, and propose a multi-stage and uni-stage fusion strategy to get the fusion representation of the multiple modalities to assist the final language-dominated multimodal representation. In our model, a Sense-Level Attention Network is proposed to dynamically learn the word representation which is guided by the fusion of the multiple modalities. As in turn, the learned language representation can also help the multi-stage and uni-stage fusion of the different modalities. In this way, the model can jointly learn a well integrated final representation focusing on the language and the interactions between the multiple modalities both on multi-stage and uni-stage. Several experiments are carried on the CMU-MOSI, the CMU-MOSEI and the YouTube public datasets. The experiments show that our model performs better or competitive results compared with the baseline models.

1 Introduction

Multimodal sentiment analysis is a task of predicting sentiment of a video, an image or a text based on multiple modal features. With the increase of short videos on the internet, such as Douyin, YouTube, etc., multimoal sentiment analysis can be used to analyze the opinions of the public based on the speaker's language, facial gestures and acoustic behaviors.

Based on the successes in video, image, audio and language processing, multimodal sentiment analysis has been studied extensively and produced impressive results in recent years (Liang et al., 2018; Liu et al., 2018; Mai et al., 2019; Wang et al., 2019; Zadeh et al., 2018c). The core of the multimodal sentiment analysis is to capture a better fusion of different modalities. Different methods are proposed to fuse the multimodal features and help to capture the interactions of the modalities. Tensor Fusion Network (Zadeh et al., 2017) is proposed to obtain raw unimodal representations, bimodal interactions and trimodal interactions in the form of 2D-tensor and 3D-tensor simultaneously. Low-rank Fusion Network (Liu et al., 2018) is then proposed to alleviate the drawback of the large amount of parameters by low-rank factor. Although the above methods achieved good results, they treat all modalities equally and fuse the modalities in the same contribution. We find that language modality always contain more semantic information for sentiment analysis, that's why most of ablation experiments of previous studies (Mai et al., 2019; Poria et al., 2017a; Zadeh et al., 2017) show that when using features from only one modality, the model using language features performs much better than using vision features or acoustic features.

In this paper, we take the assumption that the language modality contains more information than that of the vision and acoustic modalities. We regard language as the major modality and hope to use

*Corresponding author: xiali@gdufs.edu.cn

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1089-1100, Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

other modalities to assist the language modality to produce better performance for multimodal sentiment analysis. To this end, we propose a multi-perspective fusion network for multimodal sentiment analysis focusing on sense attentive language. Our model focuses on two aspects: (1)getting rich semantic language representation through the fusion of the sense level attention of language guided by other modalities. (2)learning comprehensive multimodal fusion from multiple perspectives, as well as keeping the enhanced language representation.

In order to get rich semantic information of the language modality, we incorporate a sense-level attention network into the model to obtain a more elaborate representation of the language. Generally speaking, there are many words which have more than one sense and their different senses may lead to different sentiment of a text in different context. Previous studies try to distinguish the ambiguities of a word from the text modality (Xie et al., 2017; Zeng et al., 2018) using HowNet (Dong, 1988) and LIWC(Pennebaker et al., 2007), while we hope the sense of a word can be distinguished not only by the context of the text but also by fusion of other modalities(video and acoustic). As an example shown in Figure1, we hope to predict the sentiment of the language "It would make sense". As can be seen, the word "sense" in the language modality has a higher attention weight which could be guided by the "smile face of the vision modality" and "high sound audio modality", and also by the "common sense" of the word "sense", which expresses more positive sentiment.

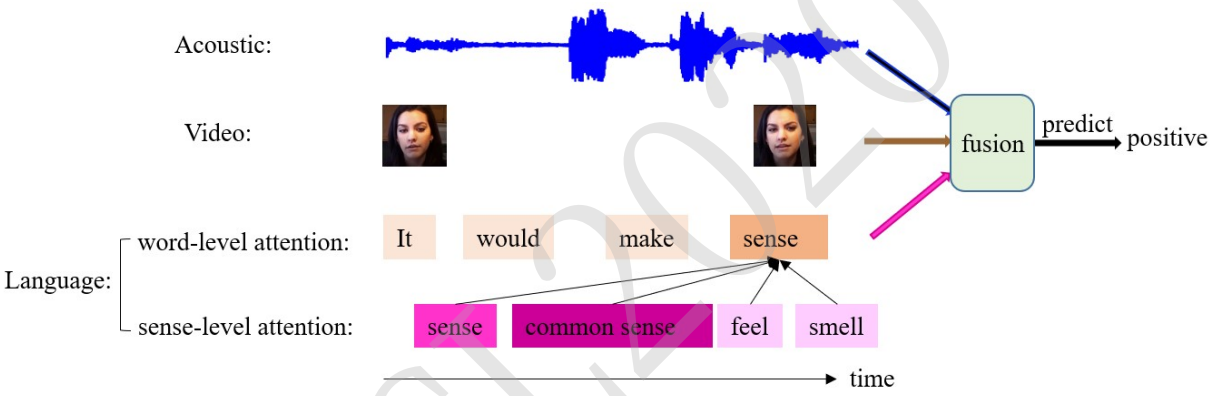


Figure 1: The sense-level attention and word-level attention of the text "It would make sense" learned by our model. The first line is the acoustic modality, the second line is the video modality. The third line and the last line are the language modality, in which the third line is the original sentence, and the last line presents the senses of word "sense". Darker color means greater weight.

For the effectiveness of modal fusion, the key problem is to model the gap between different modalities and to learn a better multimodal fusion. In this paper, we propose a multi-stage and uni-stage strategy to fuse the multiple modalities in order to capture the interactions between multi-stage sharing information and global information integrated from uni-stage fusion. For multi-stage fusion, we use CNN with different window sizes to capture the multimodal fusion of consecutive temporals within different windows respectively. As for uni-stage fusion, we first perform a projection operation on the concatenation of the LSTM outputs of three modalities, then attention mechanism is applied to learn the different contributions of the multimodal features at each temporal and produce a summary, which is regarded as the global multimodal fusion. The main contributions of our work are as follows:

1) To the best of our knowledge, this is the first time to use WordNet to reduce ambiguity for the task of multimodal sentiment analysis, which dynamically learn the different weights of the sense words to produce sense-attentive language presentation.

2) We propose to take language as the major modality and learn multimodal fusion from multi-stage and uni-stage perspective. Our final representation not only contains multimodal fusion, but also keeps the language representation, which is helpful in multimodal sentiment analysis.

3) Our model outperforms the baseline models on the CMU-MOSI, the CMU-MOSEI and the YouTube datasets and the ablation study shows the effectiveness of each components in our model.

2 Related Work

Compared with conventional text-based sentiment analysis, sentiment analysis with multiple modalities achieves significant improvements (Baltrusaitis et al., 2019). One of the most challenging task in multimodal sentiment analysis is to learn a joint representation of multiple modalities.

Earlier work uses fusion approaches such as concatenation of multi-modality features (Lazaridou et al., 2015; Ngiam et al., 2011), while recent studies propose more sophisticated fusion approaches. Poria et al. (2017a) propose a LSTM-based model to capture contextual information. Zadeh et al. (2017) propose a Tensor Fusion Network to explicitly aggregate unimodal, bimodal and trimodal interactions. Liu et al. (2018) propose a Low-rank Fusion Network to alleviate the drawback of the large amount of parameters by low-rank factor. Chen et al. (2017) propose a Gated Multimodal Embedding model to learn an on-off switch to filter noisy or contradictory modalities.

As the modalities can have interactions between different timestamps, several models are proposed to fuse the multiple modals from different views. Zadeh et al. (2018c) propose a Multi-attention Recurrent Network (MARN) to capture the interaction between modalities at different timestamps. Zadeh et al. (2018a) propose a Memory Fusion Network to learn view-specific interactions and use an attention mechanism called the Delta-memory Attention Network (DMAN) to identify the cross-view interactions. Liang et al. (2018) propose a Recurrent Multistage Fusion Network (RMFN) to model cross-modal interactions using multi-stage fusion approach, in which each stage of fusion focuses on a different subset of multimodal signals, learning increasingly discriminative multimodal representations.

Recently, Pham et al. (2019) propose to learn joint representations based on translations between modalities. They use a cycle consistency loss to ensure that the joint representations retain maximal information from all modalities. Instead of directly fusing features at holistic level, Mai et al. (2019) propose a strategy named 'divide, conquer and combine' for multimodal fusion. Their model performs fusion hierarchically to consider both local and global interactions. Wang et al. (2019) propose a Recurrent Attended Variation Embedding Network (RAVEN) to model expressive nonverbal representations by analyzing the fine-grained visual and acoustic patterns. Tsai et al. (2019) introduce a model that factorizes representations into two sets of independent factors: multimodal discriminative and modality-specific generative factors to optimize for a joint generative-discriminative objective across multimodal data and labels.

Although previous studies propose many effective approaches, most of them treat all modalities equally during the learning of multimodal fusion, which are different from our approach. In our model, we propose a sense-level attention network to learn different word representation under different senses. With the sense-attentive word representation, we can learn enhanced language representation. In addition, we try to learn sufficient multiomodal fusion through multi-stage fusion and uni-stage fusion, as well as keeping the language representation to form our final representation.

3 Our Model

Our model consists of three components: sense attentive language representation which is regared as the main representation of the multimodal fusion; multi-stage multimodal fusion which is designed to capture the interactions between the sharing information on the multi-stage; uni-stage multimodal fusion which is used to capture the global fusion information. The whole architecture of our model is shown in Figure 2. In the following sections, we will introduce the sense-level attention network in section 3.1, and describe the multi-stage multimodal fusion and the uni-stage multimodal fusion strategy in section 3.2. Section 3.3 describes the final representation and model trainning.

3.1 Sense-level Attention Network

As language has rich semantic information, a word may has different senses in different contexts, which may make the sentiment of a sentence totally different. However, the word's embedding representation is unique in the pretrained embeddings. In order to let the model to better distinguish different meanings of a same word, similar to the work of (Xie et al., 2017; Zeng et al., 2018), we use WordNet to get k number of different senses of a word into the model. If a word don't have any sense in WordNet, we input

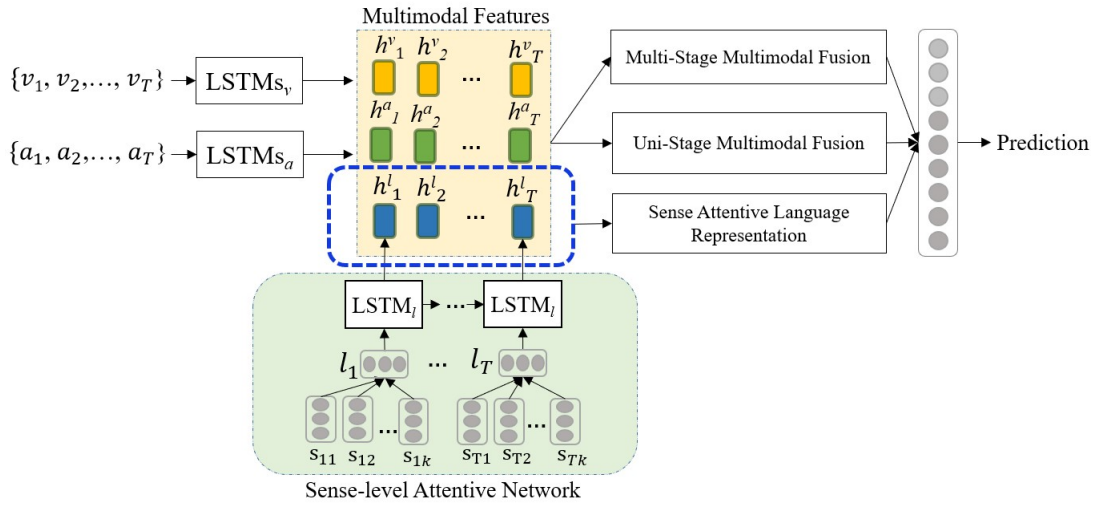


Figure 2: The whole architecture of our model. The sense-level attention is used to learn the different importance of the sense words of each word in the language modality and produce a sense-attentive representation of language. LSTM layers are then used to model the features from language, vision and acoustic modalities. Three blocks are used to learn multi-stage multimodal fusion, uni-stage multimodal fusion and language representation respectively, which are concatenated to form the final representation.

k number of original words into the model. If there are more than k number of senses for the word, we take the first k number of senses in order and pad the sense sequence with the original word if the number of senses of the word is less than k. We denote the sense sequence of the i-th word in the sentence as $S_i = \{s_{i1}, s_{i2}, \ldots, s_{ik}\}$. The word senses and the original word are converted into embeddings to be input into the model. Then attention mechanism is used to learn the importance weight of different senses of a word and the weighted sum of the embeddings of different senses forms the new representation l_i of the word, as shown in equations (1-3), where W_i and u_i are the trainable weights, b_i is the bias.

$$o_{ij} = relu\left(W_i s_{ij} + b_i\right) \tag{1}$$

$$\alpha_{ij} = softmax\left(u_i o_{ij}\right) \tag{2}$$

$$l_i = \sum_{j=1}^{k} a_{ij} s_{ij} \tag{3}$$

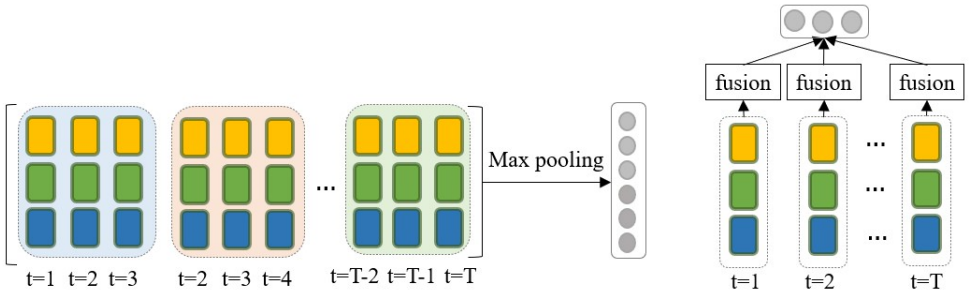
3.2 Multi-stage and Uni-stage Multimodal Fusion

In order to obtain comprehensive multimodal fusion, we propose two strategies to learn the relationship and interactive information between multiple modal features, which are multi-stage fusion and uni-stage fusion. The two strategies are shown in Figure 3.

After getting the new representation of language modality and the original features of acoustic and vision modality, denoted as $L = \{l_1,\ l_2,\ \ldots, l_T\}, A = \{a_1, a_2,\ .., a_T\}$ and $V = \{v_1, v_2,\ .., v_T\}$ respectively. We use three LSTM layers for modeling the features, aiming to consider the inter-relationship of the individual modality in different timestamps. The outputs of LSTM of acoustic, vision and language modality are denoted as $H_A = \{h_1^a,\ h_2^a, \ldots, h_T^a\}, H_V = \{h_1^v,\ h_2^v, \ldots, h_T^v\}$ and $H_L = \{h_1^l,\ h_2^l, \ldots, h_T^l\}$ respectively.

Multi-stage Multimodal Fusion. First we concatenate h_i^a, h_i^v and h_i^l, then we use different CNN layers with different window sizes to learn the multi-stage shared fusion. For CNN with window size 1, we aim to model the relationship between the three modalities timestamp by timestamp, through which we can get the fusion about the word, facial expression and speech tone of the speaker at the same timestamp. For CNN with window size bigger than 1, we aim to model the relationship between the three modalities within several timestamps. We perform maxpooling operation on top of the CNNs

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1089-1100, Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China



(a) Multi-Stage Fusion (slide window=3) (b) Uni-Stage Fusion

Figure 3: The strategies of multimodal fusion proposed in our model. The multi-stage fusion aims to capture the interactions of the shared multimodal information in different time steps. The uni-stage fusion aims to capture the global interactions of multimodal features fused within the same timestep.

respectively and concatenate the results, getting the multi-stage shared multimodal fusion $h_{multi-stage}$. The convolution operation of CNN is shown in equation(4-6), where W_z and b_z are trainable weights and bias respectively, w is the window size, f is activation function which is $relu$ in our implementation and [ ] denotes for concatenation.

$$h_i = \left[h_i^a, h_i^v, h_i^l\right] \tag{4}$$

$$z_i = f\left(W_z\left[h_i : h_{i+w-1}\right] + b_z\right) \tag{5}$$

$$Z_w = maxpooling\left(\left[z_1, z_2, ..., z_T\right]\right) \tag{6}$$

As stated above, we use different CNN layers with different window sizes following maxpooling operation, getting Z_w representation ($w = 1, 2, ...$), finally Z_w are concatenated to form the multi-stage fusion $h_{multi-stage}$.

Uni-stage Multimodal Fusion. The uni-stage fusion is applied to learn the different contributions of the multimodal feature at each temporal and produce a summary, which is regarded as the global multimodal fusion. We use another block to learn uni-stage multimodal fusion. Specifically, as shown in equation (7), we use a non-linear projection layer to project features of three modalities into the same space.

$$h_i^{'} = f\left(W_f\left[h_i^a, h_i^v, h_i^l\right] + b_f\right) \tag{7}$$

Where W_f is the trainable weights, b_i is the bias, f is $relu$ activation function and [ ] denotes for concatenation. Then we perform attention operation on the projected results $h_i^{'}$ to get a summary about of which stages the multimodal features are most important for sentiment analysis, as shown in equations (8-10).

$$o_i = tanh\left(W_a h_i^{'} + b_i\right) \tag{8}$$

$$\alpha_i = \ softmax\left(u_a o_i\right) \tag{9}$$

$$h_{uni-stage} = \sum_{i=1}^{T} a_i h_i^{'} \tag{10}$$

Where α_i is the attention weight of timestamp i.We use the attention weights to perform weighted sum on $h_i^{'}$, getting the uni-stage multimodal fusion $h_{uni-stage}$.

3.3 Final Representation and Model Training

As mentioned before, we believe that language modality contains richer information than other modalities, thus we perform attention operation on H_L to get the final language representation h_l. At last

we concatenate h_l, the multi-stage multimodal fusion $h_{multi-stage}$ and uni-stage multimodal fusion $h_{uni-stage}$ to form the final representation h_{final}. The final representation is input to a fully-connected layer and a prediction layer to get the output, as shown in equations (11-12):

$$h'_{final} = relu\left(W_1 h_{final} + b_1\right) \tag{11}$$

$$y = f(W_2 h'_{final} + b_2) \tag{12}$$

Where W_1 and W_2 are trainable weights, b_1 and b_2 are biases. f is $softmax$ function for classification task. For regression task, we don't need activation function. y is the prediction.

4 Experiments

4.1 Dataset

We conduct several experiments on the CMU-MOSI(Zadeh et al., 2016) dataset, the CMU-MOSEI(Zadeh et al., 2018b) dataset and the YouTube(Morency et al., 2011) dataset. The CMU-MOSI dataset contains 93 videos from the social media website, each of which comes from a different speaker who is expressing his or her opinions towards a movie. The videos in CMU-MOSI dataset are split into 2199 video clips, and each clip has a sentiment label $y \in [-3, 3]$, which represents strongly positive(labeled as +3), positive(+2), weakly positive(+1), neutral(0), weakly negative(-1), negative(-2), strongly negative(-3) respectively. The CMU-MOSEI dataset is a made up of 23,043 movie review video clips taken from YouTube. Following (Mai et al., 2019), we consider positive, negative and neutral sentiments in the paper. The YouTube dataset is collected from YouTube which contains 269 video clips. The statistical information of the three datasets is shown in Table 1.

Dataset	CMU-MOSI	CMU-MOSEI	YouTube
#Train	1284	15920	173
#Valid	229	2291	36
#Test	686	4832	60

Table 1: The statistical information of the experimental dataset.

4.2 Evaluation Metrix

Following previous work, we use different evaluation metrix on differetn datasets. For CMU-MOSI, we conduct experiments on binary classification task, multi-class classification task and regression task. For binary classification, we report accuracy and F1 score, whereas for multi-class classification we only report accuracy. For regression task, we report Mean Absolute Error (MAE) and Pearson's Correlation (Corr). For all the metrics, higher values denote better performance, except MAE where lower values denote better performance. For CMU-MOSEI and YouTube datasets, we conduct 3 classification task and report accuracy and F1 score.

4.3 Experimental Details

For all datasets, 300-dimensional GloVe embeddings(Pennington et al., 2014) are used to represent the language features; Facet[0] library is used to extract a set of visual features and COVAREP(Degottex et al., 2014) is used to extract acoustic features. We use WordNet to get 4 sense words for each word. Note that we add a constraint that the sense words should contain the orginal word. Besides, in WordNet, sense may contains more than one word, if this happen we use the average embedding of the words in the sense as the representation of the sense. The sizes of hidden states of LSTMs encoding language features, vision features and acoustic features are 100, 10 and 30 respectively. We use CNNs with window size 1 and 3 respectively to learn the multi-stage multimodal fusion and the filter number of CNN is set to 50. The batch size is set to 32, 16 and 16 for CMU-MOSI, CMU-MOSEI, YouTube datasets respectively,

[0]https://imotions.com/biosensor/fea-facial-expression-analysis/

and the initial learning rate is set to 0.0008, 0.0003 and 0.0001 for the three datasets respectively. For CMU-MOSI dataset, we use L1 loss as training loss, for other two datasets, we use cross entropy loss as training loss. We report the experimental results predicted by the model which performs best on the validation set.

4.4 Baseline Models

We use several models as our baselines to compare with our model. Firstly,we use THMM(Morency et al., 2011) and MV-HCRF(Song et al., 2012) as the traditional baseline models. THMM(Morency et al., 2011) concatenates language, acoustic and vision features and then uses HMM for classification. MV-HCRF(Song et al., 2012) is an extension of the HCRF for Multi-view data, explicitly capturing view-shared and view specific sub-structures. Secondly, we use MV-LSTM(Rajagopalan et al., 2016), BC-LSTM(Poria et al., 2017a), CAT-LSTM(Poria et al., 2017b), GME-LSTM(Chen et al., 2017), TFN(Zadeh et al., 2017),CHFusion(Majumder et al., 2018),LMF(Liu et al., 2018),MFN(Zadeh et al., 2018a), RMFN(Liang et al., 2018) and MARN(Zadeh et al., 2018c) as the early neural network based compared models. Lastly, we use several previous state of the art models as our baseline models. MCTN(Pham et al., 2019) learns joint representations of multi-modalities by cyclic translations between modalities. HFFN(Mai et al., 2019) proposes a hierarchical feature fusion network, named 'divide, conquer and combine' to explore both local and global interactions in multiple stages. MFM (Tsai et al., 2019) is proposed to optimize for a joint generative-discriminative objective across multimodal data and labels.

4.5 Experimental Results

Experimental Results on the CMU-MOSI Dataset. The results of our model and baseline models on the CMU-MOSI dataset is shown in Table 2. As is shown, the neural network based models outperform traditional machine learning models with a large margin. Among all models, our model achieves the second best performance on accuracy and F1 score of binary classification and accuracy of 7 classification, and our model achieves the best performance on MAE and Pearson's correlation of regression task compared with the baseline models. Specifically, our model achieves competitive results compared with HFFN on binary classification task, and outperforms MCTN, which is the best model on MAE among the baseline models by 4.5% on MAE. For Pearson's correlation (Corr), our model outperforms RMFN which achieves the best performance on Corr among the baselines by 3.9%. As for seven classification task, we achieve the second best performance. The overall experimental results on the CMU-MOSI dataset show the effectiveness of our model.

Experimental Results on YouTube Dataset. Table 3 shows the experimental results of our model and the baseline models on YouTube dataset. YouTube is a very small dataset, as shown in Table 1, we can see that not all neural network based models outperform traditional machine learning models both on accuracy and F1 score. However, compared with the baseline models, our model achieves the best performance on both accuracy and F1 score, which outperforms the previous state-of-the-art model MFM by 1.7% on accuracy and 3.5% on F1 score. Although YouTube dataset is very small, our model can achieve the best performance among the baseline models.

Experimental Results on CMU-MOSEI Dataset. For CMU-MOSEI dataset, we conduct experiments on 3 classification tasks. We present the experimental results of different models in Table 4. As we can see, our model achieves the best performance on both accuracy and F1 score, which outperforms HFFN by 0.93% on accuracy and 0.6% on F1 score, and outperforms BC-LSTM by 0.53% on accuracy and 0.63% on F1 score. Note that CMU-MOSEI is the largest dataset in this paper. In addition, we can see that although CAT-LSTM and LMF achieve relative good performance on accuracy, their performance on F1 score is much worse than that on accuracy. Our model can achieve both good performance on accuracy and F1 score. Experimental results on CMU-MOSEI dataset and YouTube dataset show that our model can adapt to both small data and large data.

Model	Binary		Regression		7-class
	Acc	F1	MAE	Corr	Acc
THMM(Morency et al., 2011)	50.7	45.4	-	-	17.8
MV-HCRF(Song et al., 2012)	65.6	65.7	-	-	24.6
MV-LSTM(Rajagopalan et al., 2016)	73.9	74.0	1.019	0.601	33.2
BC-LSTM (Poria et al., 2017a)	73.9	73.9	1.079	0.581	28.7
GME-LSTM (Chen et al., 2017)	76.5	73.4	0.955	-	-
TFN (Zadeh et al., 2017)	74.6	74.5	1.040	0.587	28.7
LMF (Liu et al., 2018)	76.4	75.7	0.912	0.668	32.8
RMFN(Liang et al., 2018)	78.4	78.0	0.922	0.681	**38.3**
MARN (Zadeh et al., 2018c)	77.1	77.0	0.968	0.625	34.7
MFN (Zadeh et al., 2018a)	77.4	77.3	0.965	0.632	34.1
MFM (Tsai et al., 2019)	78.1	78.1	0.951	0.662	36.2
MCTN (Pham et al., 2019)	79.3	79.1	0.909	0.676	-
HFFN (Mai et al., 2019)	**80.2**	**80.3**	-	-	-
MPFN(Ours)	80.0	80.0	**0.864**	**0.720**	37.0

Table 2: Experimental results of different models on CMU-MOSI dataset.

Model	Acc	F1
THMM (Morency et al., 2011)	42.4	27.9
MV-HCRF (Song et al., 2012)	44.1	44.0
MV-LSTM (Rajagopalan et al., 2016)	45.8	43.3
BC-LSTM (Poria et al., 2017a)	45.0	45.1
TFN(Zadeh et al., 2017)	45.0	41.0
MARN (Zadeh et al., 2018c)	48.3	44.9
MFN (Zadeh et al., 2018a)	51.7	51,6
MCTN (Pham et al., 2019)	51.7	52.4
MFM (Tsai et al., 2019)	53.3	52.4
MPFN(Ours)	**55.0**	**55.9**

Table 3: Experimental results of different models on YouTube dataset

Model	Acc	F1
BC-LSTM (Poria et al., 2017a)	60.77	59.04
TFN (Zadeh et al., 2017)	59.40	57.33
CAT-LSTM (Poria et al., 2017b)	60.72	58.83
CHFusion (Majumder et al., 2018)	58.45	56.90
LMF (Liu et al., 2018)	60.27	53.87
HFFN (Mai et al., 2019)	60.37	59.07
MPFN(Ours)	**61.30**	**59.67**

Table 4: Experimental results of different models on CMU-MOSEI dataset.

4.6 Ablation Studies

In order to investigate the impact of various components in our model, we conduct several ablation experiments on the CMU-MOSI dataset, which are shown in Table 5. In the experiment, we remove one kind of component of our full model each time. Specifically, we remove the sense-level attention (denoted as MPFN-no-sense-att), the multi-stage multimodal fusion (denoted as MPFN-no-multi-stage-fusion), the uni-stage multimodal fusion (denoted as MPFN-no-uni-stage-fusion) and final language representation (denoted as MPFN-no-language-final) respectively.

As shown in Table 5, once we remove any component of our model, the performance will decline.

For example, if we remove the sense-level attention and use the original word embedding as word representation, the performance of our model will drop by 1.0% on accuracy, 1.4% on F1 score of binary classification task, 3.8% on MAE, 2.5% on Corr, and 2.9% on accuracy of 7 classification task on the CMU-MOSI dataset. This observation suggests that using WordNet and sense-level attention to dynamically learn the word representation is effective.

In terms of multimodal fusion, we can see that if we remove the multi-stage fusion block or the uni-stage fusion block, the performance of our model will also drop, which indicates that both multi-stage fusion and uni-stage fusion are important for multimodal sentiment analysis. Furthermore, it seems that the multi-stage multimodal fusion plays a more important role than uni-stage multimodal fusion on the CMU-MOSI dataset.

Last but not least, we remove the final language representation which is concatenated with the multimodal fusion representation to see whether this operation is useful. The experimental results prove our early assumption. As we mentioned, ablation studies of previous researches show that if only using features of one modality as input, the model which use language modality features as input performs best. If only using multimodal fusion representation to form the final representation, some intra-modality information of language will be lost during fusion process. Concatenating the final language representation with the multimodal fusion representation to form the final representation can address this problem.

Model	**Binary**		**Regression**		**7-class**
	Acc	F1	MAE	Corr	Acc
MPFN-no-sense-att	79.0	78.6	0.902	0.695	34.1
MPFN-no-multi-stage-fusion	79.0	79.0	0.882	0.698	36.9
MPFN-no-uni-stage-fusion	79.3	79.3	0.888	0.711	33.5
MPFN-no-language-final	79.3	79.3	0.899	0.714	34.4
MPFN(Ours)	**80.0**	**80.0**	**0.864**	**0.720**	**37.0**

Table 5: Ablation study on CMU-MOSI dataset.

4.7 Discussion

In order to investigate how each modality effects the performance of our model, we conduct several experiments to compare the performance of our model using unimodal, bimodal and multimodal features, as shown in Table 6.

For unimodal features, we can see that our model only using sense attentive language representation outperforms the model that only using audio features or video features with significant margin, which is consistent with our early assumption that language modality is dominant. For bimodal features, we can infer that when integrating language modaltity with acoustic modality or vision modality, the performance of the model outperforms that of only using language representation, which indicates that acoustic and vision modalities play auxiliary roles and the multi-perspective multimodal fusion can improve the performance of the model. However, when using audio features and video features as input, the performance of the model is still much worse than that of only using language modality, which again proves that language modality is the most important modality in this task.

When cooperating three modalities, our full model MPFN achieves the best performance among the different combinations, which demonstrates the effectiveness of multi-perspective multimodal fusion proposed in this paper.

5 Conclusion

In this paper, we propose a novel multi-perspective fusion network focusing on sense attentive language for multimodal sentiment analysis. Evaluations show that using our proposed multi-stage and uni-stage fusion strategies and using sense attentive language representation can improve performance on multimodal sentiment analysis for the CMU-MOSI, CMU-MOSEI and YouTube data. Our model also achieves a new state-of-the-art in the YouTube and CMU-MOSEI dataset on accuracy and F1 measure

Modality	Source	Binary		Regression		7-class
		Acc	F1	MAE	Corr	Acc
Unimodal	Audio	57.1	56.2	1.396	0.196	16.0
	Video	57.3	57.3	1.431	0.137	16.2
	Sense attentive language	79.0	79.1	0.922	0.689	34.0
Bimodal	Sense attentive language +Audio	79.7	79.6	0.881	0.701	34.7
	Sense attentive language +Video	79.6	79.6	0.915	0.714	32.9
	Audio+Video	59.0	59.0	1.391	0.176	19.7
Multimodal	**Sense attentive language +Audio+Video**	**80.0**	**80.0**	**0.864**	**0.720**	**37.0**

Table 6: The performance of our model using unimodal, bimodal and multimodal features.

metrics compared with the baseline models. The experimental results using different modal combinations also show that the proposed sense attentive language modal achieves the most significant performance improvement on the CMU-MOSI dataset, especially on the 7-classification results, indicating that the sense attentive language modal plays an important role in multimodal sentiment analysis task. Like most of other models, our approach also focuses on the multimodal data with the same length of stamp. In the future, we will investigate a novel fusion of multimodal data with different length of stamp.

Acknowledgements

This work is supported by National Natural Science Foundation of China (No. 61976062) and the Science and Technology Program of Guangzhou (No. 201904010303).

References

Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency. 2019. Multimodal machine learning: A survey and taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(2):423–443.

Minghai Chen, Sen Wang, Paul Pu Liang, Tadas Baltrusaitis, Amir Zadeh, and Louis-Philippe Morency. 2017. Multimodal sentiment analysis with word-level fusion and reinforcement learning. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction, ICMI 2017*, pages 163–171.

Gilles Degottex, John Kane, Thomas Drugman, Tuomo Raitio, and Stefan Scherer. 2014. COVAREP - A collaborative voice analysis repository for speech technologies. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014*, pages 960–964.

Zhendong Dong. 1988. Knowledge description: what, how and who. In *Proceedings of International Symposium on Electronic Dictionary*, volume 18.

Angeliki Lazaridou, Nghia The Pham, and Marco Baroni. 2015. Combining language and vision with a multimodal skip-gram model. In *The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2015*, pages 153–163.

Paul Pu Liang, Ziyin Liu, Amir Zadeh, and Louis-Philippe Morency. 2018. Multimodal language analysis with recurrent multistage fusion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, pages 150–161.

Zhun Liu, Ying Shen, Varun Bharadhwaj Lakshminarasimhan, Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. 2018. Efficient low-rank multimodal fusion with modality-specific factors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018*, pages 2247–2256.

Sijie Mai, Haifeng Hu, and Songlong Xing. 2019. Divide, conquer and combine: Hierarchical feature fusion network with local and global perspectives for multimodal affective computing. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019*, pages 481–492.

Navonil Majumder, Devamanyu Hazarika, Alexander F. Gelbukh, Erik Cambria, and Soujanya Poria. 2018. Multimodal sentiment analysis using hierarchical fusion with context modeling. *Knowl. Based Syst.*, 161:124–133.

Louis-Philippe Morency, Rada Mihalcea, and Payal Doshi. 2011. Towards multimodal sentiment analysis: harvesting opinions from the web. In *Proceedings of the 13th International Conference on Multimodal Interfaces,ICMI 2011*, pages 169–176.

Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. 2011. Multimodal deep learning. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, pages 689–696.

James W Pennebaker, Roger J Booth, and Martha E Francis. 2007. Linguistic inquiry and word count: Liwc [computer software]. *Austin, TX: liwc. net*, 135.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014*, pages 1532–1543.

Hai Pham, Paul Pu Liang, Thomas Manzini, Louis-Philippe Morency, and Barnabás Póczos. 2019. Found in translation: Learning robust joint representations by cyclic translations between modalities. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019*, pages 6892–6899.

Soujanya Poria, Erik Cambria, Devamanyu Hazarika, Navonil Majumder, Amir Zadeh, and Louis-Philippe Morency. 2017a. Context-dependent sentiment analysis in user-generated videos. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017*, pages 873–883.

Soujanya Poria, Erik Cambria, Devamanyu Hazarika, Navonil Majumder, Amir Zadeh, and Louis-Philippe Morency. 2017b. Multi-level multiple attentions for contextual multimodal sentiment analysis. In *2017 IEEE International Conference on Data Mining, ICDM 2017*, pages 1033–1038.

Shyam Sundar Rajagopalan, Louis-Philippe Morency, Tadas Baltrusaitis, and Roland Goecke. 2016. Extending long short-term memory for multi-view structured learning. In *Computer Vision - ECCV 2016 - 14th European Conference*, volume 9911 of *Lecture Notes in Computer Science*, pages 338–353.

Yale Song, Louis-Philippe Morency, and Randall Davis. 2012. Multi-view latent variable discriminative models for action recognition. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2120–2127.

Yao-Hung Hubert Tsai, Paul Pu Liang, Amir Zadeh, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2019. Learning factorized multimodal representations. In *7th International Conference on Learning Representations, ICLR 2019*.

Yansen Wang, Ying Shen, Zhun Liu, Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. 2019. Words can shift: Dynamically adjusting word representations using nonverbal behaviors. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019*, pages 7216–7223.

Ruobing Xie, Xingchi Yuan, Zhiyuan Liu, and Maosong Sun. 2017. Lexical sememe prediction via word embeddings and matrix factorization. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017*, pages 4200–4206.

Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. 2016. MOSI: multimodal corpus of sentiment intensity and subjectivity analysis in online opinion videos. *Computing Research Repository*, arXiv:1606.06259.

Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2017. Tensor fusion network for multimodal sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017*, pages 1103–1114.

Amir Zadeh, Paul Pu Liang, Navonil Mazumder, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2018a. Memory fusion network for multi-view sequential learning. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI 2018*, pages 5634–5641.

Amir Zadeh, Paul Pu Liang, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2018b. Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018*, pages 2236–2246.

Amir Zadeh, Paul Pu Liang, Soujanya Poria, Prateek Vij, Erik Cambria, and Louis-Philippe Morency. 2018c. Multi-attention recurrent network for human communication comprehension. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI 2018*, pages 5642–5649.

Xiangkai Zeng, Cheng Yang, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2018. Chinese LIWC lexicon expansion via hierarchical classification of word embeddings with sememe attention. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI 2018*, pages 5650–5657.

CAN-GRU: a Hierarchical Model for Emotion Recognition in Dialogue

Ting Jiang **Bing Xu** **Tiejun Zhao** **Sheng Li**
Laboratory of Machine Intelligence and Translation
School of Computer Science and Technology, Harbin Institute of Technology
Harbin, China
jiangting_hit@163.com {hitxb,tjzhao,lisheng}@hit.edu.cn

Abstract

Emotion recognition in dialogue systems has gained attention in the field of natural language processing recent years, because it can be applied in opinion mining from public conversational data on social media. In this paper, we propose a hierarchical model to recognize emotions in the dialogue. In the first layer, in order to extract textual features of utterances, we propose a convolutional self-attention network(CAN). Convolution is used to capture n-gram information and attention mechanism is used to obtain the relevant semantic information among words in the utterance. In the second layer, a GRU-based network helps to capture contextual information in the conversation. Furthermore, we discuss the effects of unidirectional and bidirectional networks. We conduct experiments on Friends dataset and EmotionPush dataset. The results show that our proposed model(CAN-GRU) and its variants achieve better performance than baselines.

1 Introduction

As an important component of human intelligence, emotional intelligence is defined as the ability to perceive, integrate, understand and regulate emotions(Mayer et al., 2008). Emotion is the essential difference between human and machine, so emotion understanding is an important research direction of artificial intelligence. As the most common way for people to communicate in daily life, dialogue contains a wealth of emotions. Recognising the emotions in the conversation is of great significance in intelligent customer service, medical systems, education systems and other aspects.

According to (Poria et al., 2015), textual features usually contain more emotional information than video or audio features, so we focus on the emotion analysis of dialogue text and aims to recognize the emotion of each utterrance in dialogues.

There are some challenges in this task. First, the length of an utterance may be too long, making it difficult to capture contextual information. Furthermore, a dialogue usually contains lots of utterances, therefore, it's hard to grasp long-term contextual relations between utterances. Second, the same word may express different emotions in different contexts. For example, in Table 1, while in different dialogues, the word 'Yeah' can express three different emotions, that is , joy, neutral and suprise. To tackle these challenges, we propose a hierarchical model based on convolutional attention network and gated recurrent unit (CAN-GRU). Existing works pay little attention to the extraction of semantic information within an utterance. In this work, we focus on this problem. Our proposed model can extract n-gram information by CNNs and use self-attention to capture contextual information within an utterance in the first layer. Moreover, we utilize a GRU-based network to model the sequence of utterances in the second layer, which can fully combine the context when analyzing utterance emotion and solve the problem of long-term dependence between texts at the same time.

2 Related Work

Text emotion recognition is one of the most hot topic in natural language processing. Recent years,a lot of classical neural networks are used to tackle this problem. Such as Long Short-Term Memory Net-

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1101-1111, Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

speaker	utterance	emotion
Phoebe	Can I tell you a little secret?	neutral
Rachel	**Yeah!**	**joy**
Wayne	Hey Joey, I want to talk to you.	neutral
Joey	**Yeah?**	**neutral**
Gary	Hey Chandler, what are you doing here?	suprise
Chandler	Gary, I'm here to report a crime.	neutral
Gary	**Yeah?**	**suprise**
Chandler	It is a crime that you and I don't spend more time together.	neutral

Table 1: The word 'Yeah' expresses different emotions in the different contexts.

work(Hochreiter and Schmidhuber, 1997), Gated Recurrent Unit Network(Cho et al., 2014) and textual Convolutional Neural Network(Kim, 2014). However, these models don't perform well when the texts are too long, because it's hard to capture the long-range contextual information. Later, attention mechanism(Bahdanau et al., 2015) is proposed to solve this problem. Recently, self-attention(Vaswani et al., 2017) is widely used since it can solve the long-term dependence problem of text effectively.

Recent years, more and more researchers focus on emotion recognition in conversation. This task aims to recognize the emotion of each utterance in dialogues. bcLSTM(Poria et al., 2017) extracts textual features by CNN and model the sequence of utterances by LSTM. Considering inter-speaker dependency relations, conversational memory network(CMN)(Hazarika et al., 2018b) has been proposed to model the speaker-based emotion using memory network and summarize task-specific details by attention mechanisms. ICON(Hazarika et al., 2018a) improves the CMN, it hierarchically models the self-speaker emotion and inter-speaker emotion into global memories. DialogueRNN(Majumder et al., 2019) uses emotion GRU and global GRU to model inter-party relation, and uses party GRU to model relation between two sequential states of the same party. DialogueGCN(Ghosal et al., 2019) improves DialogueRNN by graph convolutional network, and it can hold richer context relevant to emotion. However, these models may be too complex for small textual dialogue datasets.

In this paper, we study on the EmotionX Challenge(Hsu and Ku, 2018), Dialogue Emotion Recognition Challenge, which aims to recognize the emotion of each utterance in dialogues. According to the overview of this task, the best team(Khosla, 2018) proposes a CNN-DCNN auto encoder based model, which includes a convolutional encoder and a deconvolutional decoder. The second place team(Luo et al., 2018) mainly uses BiLSTM with a self-attentive architecture on the top for the classiffication. The third place team(Saxena et al., 2018) proposes a hierarchical network based on attention models and conditional random fields(CRF). For a meaningful comparison, we use the same dataset and metric as the challenge in our study.

3 Method

3.1 Task Definition

Given a dialogue $dia = \{u_1, u_2, ..., u_N\}$, where N is the number of utterances in the dialogue, $u_i = \{w_1, w_2, ..., w_L\}$ represents the $ith(1 \leq i \leq N)$ utterance in the dialogue that consists of L words, our goal is to analyze the emotion of each utterance in the dialogue. To solve this task, we propose a hierarchical model CAN-GRU and extend three variants, CAN-GRUA, CAN-biGRU and CAN-biGRUA(illustrated in Fig. 1).

3.2 Text Feature Extraction

In this section, we discuss the first layer of the model. Like (Poria et al., 2017), we use convolutional neural network to extract the features of the utterance. Inspired by (Gao et al., 2018), in order to capture the contextual information of long text effectively, we use convolutional self-attention network(CAN) instead of traditional CNN network.

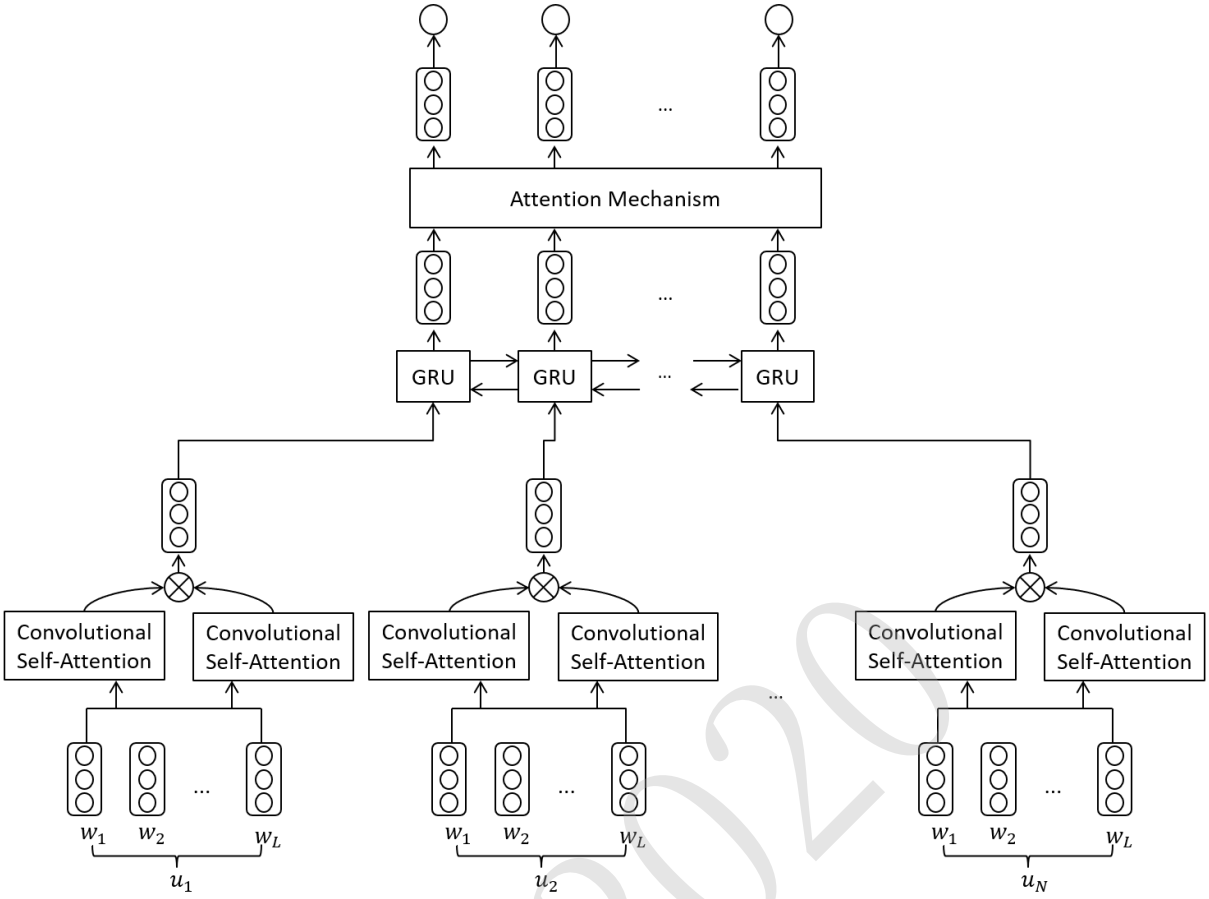


Figure 1: The architecture of our proposed CAN-biGRUA. In the first layer, convolutional neural network and self-attention mechanism are used to extract text features. In the second layer, biGRU with an attentive architecture on the top is used to model the sequence of utterances in the dialogue.

For query embedding Q, key embedding K and value embedding V involved in attention operation, they may need different effective features. And different effective information can be extracted in different convolution operations. So we obtain the Q, K and V embeddings by convolving the input word embeddings, instead of using the input word embeddings as the Q, K and V embeddings directly:

$$Q = f(conv(E, W_q) + b_q) \tag{1}$$

$$K = f(conv(E, W_k) + b_k) \tag{2}$$

$$V = f(conv(E, W_v) + b_v) \tag{3}$$

In the equations above, E is the input word embeddings, $\{E, Q, K, V\} \in \mathbb{R}^{l \times d}$, where l means the length of the sentence and d means the embedding dimension. $\{W_q, W_k, W_v\} \in \mathbb{R}^{w \times n \times d}$, where w is the window size of filters and n is the feature maps of filters. $\{b_q, b_k, b_v\} \in \mathbb{R}^d$. $conv(E, W)$ means convolution operation between E and W. And f is the activation function.

After getting Q, K, V embeddings, we calculate semantic relations among words within the utterance by the scaled dot product attention operation. More specifically, Q and K operate to get the weight matrix. Then we scale this weight matix by $\sqrt{d}$. After that, softmax operation is conducted to obtain the standardized weight matrix, which is used to express the degree of attention between words in the sentence, then the normalized weight matrix is mutiplied with V to get the result $Z \in \mathbb{R}^{l \times d}$ of attention operation:

$$Z = softmax(\frac{QK^T}{\sqrt{d}})V \tag{4}$$

As mentioned in (Gao et al., 2018), attention mechanisms cannot capture complex interactions, because it is designed for creating weighted averages. So we do the equations(1)-(3) twice to create Q_a, K_a, V_a and Q_b, K_b, V_b respectively, and get Z_a, Z_b by operate equation(4) respectively, then perform elementwise multiplication on Z_a and Z_b to get $U \in \mathbb{R}^{l \times d}$:

$$U = Z_a \otimes Z_b \tag{5}$$

Finally, for each $U_i (1 \leq i \leq N)$ in $dia = \{U_1, U_2, ..., U_N\}$, we get the individual embedding e_{u_i} by max-pooling on the contextual word embeddings within the U_i. In this way, we obtain a set of utterance embeddings $\{e_{u_1}, e_{u_2}, ..., e_{u_N}\}$ in one dialogue.

3.3 Dialogue Modeling

In this section, we discuss the sencond layer of the model. Considering different networks, we propose the hierarchical model CAN-GRU and its three progressive variants.

CAN-GRU: In real life, when we analyze the emotion of the current utterance, we can only refer to the historical information of the past utterances in the conversation. So in our model, we use GRU to model the sequence of utterances in the dialogue, because it can memory and transmit historical information. GRU(Cho et al., 2014) is an improved model for the original recurrent neural networks and it performs well with simple calculation. At timestep t, it use reset gate R_t and update gate Z_t to calculate current hidden state S_t with input utterance embedding e_{u_t} and hidden state s_{t-1} at the previous time step.

$$R_t = \sigma(e_{u_t} W_{ur} + S_{t-1} W_{sr} + b_r) \tag{6}$$

$$Z_t = \sigma(e_{u_t} W_{uz} + S_{t-1} W_{sz} + b_z) \tag{7}$$

$$H_t = tanh(e_{u_t} W_{uh} + (R_t \otimes S_{t-1}) W_{sh} + b_h) \tag{8}$$

$$S_t = Z_t \otimes S_{t-1} + (1 - Z_t) \otimes H_t \tag{9}$$

where W, b are trainable parameters and $\otimes$ means elementwise mutilication.

CAN-GRUA: However, it is difficult to grasp long-term dependence between sentences when there are too many sentences in a conversation. That is, it is hard for the current utterance to capture the historical information contained in the distant utterance. To solve this problem, we connect an attention layer upon the GRU to obtain the influence degree of historical information on the emotion of the current utterance. If the weight calculated by attention mechanism tends to be large, it indicates that the preceding utterance have an important influence on the current utterance, so this preceding utterance should be given more attention.

$$\tilde{S_t} = \sum_{i=1}^{t-1} S_i \alpha_i \tag{10}$$

$$\alpha_i = \frac{exp(S_t S_i)}{\sum_{i=1}^{t-1} exp(S_t S_i)} \tag{11}$$

Here, $S_t \in \mathbb{R}^m$ is the current hidden state, where m is the dimension of the hidden state. $S_i \in \mathbb{R}^m$ is the preceding hidden state at time step i, $\tilde{S_t} \in \mathbb{R}^m$ is the attention result at time step t.

CAN-biGRU: In fact, when analyzing the emotion of the utterance, we can not only use the historical information before the utterance, but also the future information after the current utterance. This is because emotional tone is usually maintained and does not shift frequently within a conversation in a short time. If we only pay attention to the historical information, it may be difficult to analyze the emotion of the current utterance, while the future information can be helpful in the analysis. Therefore, using both historical and future information can help to capture a richer context. Bidirectional GRU(biGRU) is used to model the sequence of utterances abstracting contextual features forward and backward, which can provides context for emotion classification more effectively.

CAN-biGRUA: As mentioned before, biGRU also suffers from the difficulty of obtaining semantic connections between long sequences. So we connect a self-attention layer on the top of the hidden states of biGRU to take full advantage of global contextual information.

$$\tilde{S}_t = \sum_{i=1}^{N} S_i \alpha_i \tag{12}$$

$$\alpha_i = \frac{exp(S_t S_i)}{\sum_{i=1}^{N} exp(S_t S_i)} \tag{13}$$

Here, $S_t \in \mathbb{R}^m$ is the current hidden state, $S_i \in \mathbb{R}^m$ is the hidden state at time step i, N is the number of sentences in the dialogue, $\tilde{S}_t \in \mathbb{R}^m$ is the attention result at time step t.

3.4 Emotion Classification

As mentioned above, we get final representations of utterances $\{\tilde{S}_1, \tilde{S}_2, ..., \tilde{S}_t, ..., \tilde{S_N}\}$. Then we utilize a fully-connected layer and a softmax layer to get the emotion class of each utterance in a dialogue.

$$f_t = tanh(W_f \tilde{S}_t + b_f) \tag{14}$$

$$o_t = softmax(W_o \tilde{f}_t + b_o) \tag{15}$$

$$\hat{y}_t = \underset{i}{argmax}(o_t[i]), i \in [1, c] \tag{16}$$

Where $W_f \in \mathbb{R}^{m \times m}$, $b_f \in \mathbb{R}^m$. $W_o \in \mathbb{R}^{m \times c}$, c is the number of emotion class, $b_o \in \mathbb{R}^c$, $o_t \in \mathbb{R}^c$, $\hat{y}_t$ is the predicted class for utterance u_t.

3.5 Training

Like(Khosla, 2018), in order to solve the problem of emotion class imbalance, we use a weigted cross entropy loss as a minimization target to optimize the parameters in the model. We give higher weight to the loss of minority class data sample in the dataset.

$$Loss = \frac{1}{K} \sum_{k=1}^{K} weight_k loss_k \tag{17}$$

$$loss_k = -[y_k log(p_k) + (1 - y_k) log(1 - p_k)] \tag{18}$$

$$\frac{1}{weight_k} = \frac{count_i}{\sum_{i=1}^{c} count_i} \tag{19}$$

Where K is the total number of samples, y_k is the ground-truth, p_k is the probability calculated in softmax layer, $count_i$ is the total number of samples in the same class as sample k.

4 Experiments

4.1 Datasets

We conduct experiments on two datasets provided by the EmotionX Challenge(Hsu and Ku, 2018).

Friends[0] : The conversations in this dataset are from the Friends TV show transcripts. The dataset contains eight emotion categories: joy, anger, sadness, surprise, fear, disgust, neutral, and non-neutral.

EmotionPush[1] : The conversations in this dataset are from the facebook messenger logs after processing the private information. Emotion categories are the same as Friends dataset.

[0] http://doraemon.iis.sinica.edu.tw/emotionlines

[1] http://doraemon.iis.sinica.edu.tw/emotionlines

In the challenge(Hsu and Ku, 2018), each dataset is divided into the training set with 720 dilogues, the validation set with 80 dialogues and the test set with 200 dialogues. Since there are few utterances for some emotions, the challenge only evaluate the performence of recognition for four emotions: joy, anger, sadness and neutral. Table 2 shows the distributions of train, validation, test samples and the distributions of the emotions for both datasets respectively.

dataset	Dialogue(Utterance)			Emotion				
	train	validation	test	anger	joy	sadness	neutral	others
Friend	720(10561)	80(1178)	200(2764)	759	1710	498	6530	5006
EmotionPush	720 (10,733)	80(1202)	200(2807)	140	2100	514	9855	2133

Table 2: Statistics of the datasets.

4.2 Evaluation Metric

We use the unweighted accuracy(UWA) as the evaluation metric instead of the weighted accuracy(WA), the same as the challenge. This is because WA is easily influced by the large proportion of neutral emotion and UWA can help to make a meaningful comparision.

$$UWA = \frac{1}{c}\sum_{i=1}^{c} a_i, WA = \sum_{i=1}^{c} weight_i a_i \tag{20}$$

Where a_i is the accuracy of class i and $weight_i$ is the percentage of the class i.

4.3 Experimental Setting

We use 300-dimensional pre-trained GloVe[2] (Pennington et al., 2014) word-embeddings which is trained from web data. We use three distinct convolution filters of sizes 3, 4, and 5 respectively, each having 100 feature maps. The dimension of the hidden states of the GRU is set to 300. We use adam(Kingma and Ba, 2015) optimizer and set the initial learning rate as 1.0×10^{-4}. The learning rate is halved every 20 epochs during training. Dropout probability is set to 0.3.

4.4 Baselines

In experiments, we compare our proposed model with the following models.

CNN-DCNN: The winner of EmotionX Challenge(Khosla, 2018). The model contains a convolutional encoder and a deconvolutional decoder. The linguistic features enhance the latent feature of the model.

SA-LSTM: The second place of the challenge(Luo et al., 2018). A self-attentive biLSTM network can provide information between utterances and the word dependency in each utterance.

HAN: The third place of the challenge(Saxena et al., 2018). LSTM with attention mechanism gets the sentence embedding. Another LSTM and CRF layer model the context dependency between sentence embeddings of the dialogue.

scGRU: We implement the basic model proposed by(Poria et al., 2017), but with a few changes. The same as (Poria et al., 2017), CNN is used to extract text features, but we use a contextual GRU network instead of a contextual LSTM network to model the sequences.

bcGRU: We implement the variant model proposed by(Poria et al., 2017), CNN is also used to obtain utterance features, but the biLSTM network used in the author's work is replaced by biGRU network.

4.5 Main Results

Table 3 presents the performance of baselines and CAN-GRU along with its variants.

Baselines: Our implemented bcGRU model performes better than scGRU on both datasets. On the Emotionpush dataset, bcGRU's performance has surpassed CNN-DCNN, and it is the best model in baselines. On the Friend dataset, CNN-DCNN remains the best baseline.

[2] http://nlp.stanford.edu/projects/glove/

model	Friend					EmotionPush				
	anger	joy	sadness	neutral	UWA	anger	joy	sadness	neutral	UWA
CNN-DCNN	55.3	71.1	55.3	68.3	62.5	45.9	**76.0**	51.7	76.3	62.5
SA-BiLSTM	49.1	68.8	30.6	**90.1**	59.6	24.3	70.5	31.0	**94.2**	55.0
HAN	39.8	57.6	50.6	73.5	55.4	21.6	63.1	54.0	88.2	56.7
scGRU	51.6	68.7	44.8	72.6	59.4	49.8	68.2	57.1	75.4	62.6
bcGRU	54.1	69.8	43.5	73.4	60.2	50.1	71.4	61.6	71.8	63.7
CAN-GRU	56.2	67.0	**55.9**	71.4	62.6	52.4	70.6	59.8	74.5	64.3
CAN-biGRU	54.8	68.1	52.9	76.3	63.0	**55.7**	71.8	60.1	74.9	65.6
CAN-GRUA	**57.6**	70.2	53.7	76.2	64.4	53.2	72.1	61.5	78.3	66.3
CAN-biGRUA	56.4	**72.6**	54.4	77.8	**65.3**	54.3	73.8	**62.9**	77.4	**67.1**

Table 3: Experimental results on Friend dataset and EmotionPush dataset.

CAN-GRU: In the first layer, it uses the convolutional self-attention mechanism to extract utterance features, and in the second layer, GRU is used to model the sequence of utterances. Compared with scGRU, it attains 3.2% and 1.7% improvement on the Friend dataset and EmotionPush dataset.

CAN-biGRU: Compared with CAN-GRU, it uses biGRU at the second layer and get improvements on the two datasets. CAN-biGRU achieves 2.8% and 1.9% improvements over bcGRU on the Friend dataset and the Emotionpush dataset respectively. Both the improvements of CAN-GRU and CAN-biGRU over baselines illustrate that the convolutional self-attention mechanism can capture contextual information in long text effectively.

CAN-GRUA: Compared with CAN-GRU, an attention mechanism is connected upon the GRU layer, which can help the model better capture the historical information of utterance and give high weight to important historical information. It gets 1.8% and 2.0% improvements over CAN-GRU on the two datasets.

CAN-biGRUA: At the top of biGRU, a self-attention mechanism is added to help calculate the importance of contextual information by using historical and future information when analyzing the current utterance emotion. This model achieves the best results, it improves 2.8% and 4.6% over baseline on the two datasets respectively.

model	Friend					EmotionPush				
	anger	joy	sadness	neutral	UWA	anger	joy	sadness	neutral	UWA
BERT	78.1	86.5	74.3	**90.3**	82.3	79.4	**89.7**	85.3	92.4	86.7
CAN-biGRU(*)	**81.2**	**87.4**	**78.7**	89.1	**84.1**	**82.8**	88.3	**87.6**	**94.1**	**88.2**

Table 4: Experimental results for BERT and CAN-biGRU(*).

In addition, we use the pretrained model BERT(Devlin et al., 2019) to get the word embeddings and input the pre-trained word embeddings into our CAN-biGRU, the experimental results are shown as the CAN-biGRU(*) in Table 4. As we can see, while BERT achieves a high degree of accuracy, our model can be further improved on the basis of BERT. CAN-biGRU(*) gets 1.8% and 1.5% improvements over BERT on the Friend dataset and the Emotionpush dataset respectively.

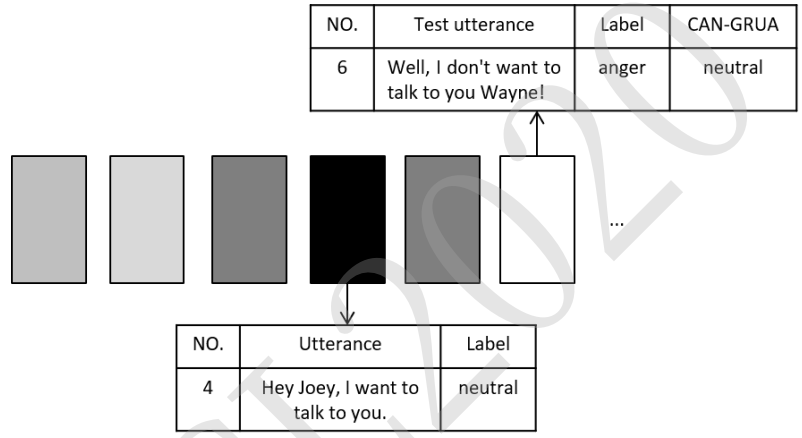
4.6 Case Study

In Table 5, we compare the emotion recognition results of bcGRU and CAN-biGRU. In the first case, 'bad' expresses strong emotion and both two model can recognize the sad emotion successfully. While there is no explicit emotion word in the third utterance, but the word 'ruined' delivers bad information, our CAN-biGRU can extract semantic information among words by CAN and gives the right prediction. In the second case, the word 'celebrating' in the third utterance express the joy emotion implicitly. Our model obtains the contextual information through the CAN, and makes the correct prediction. However,

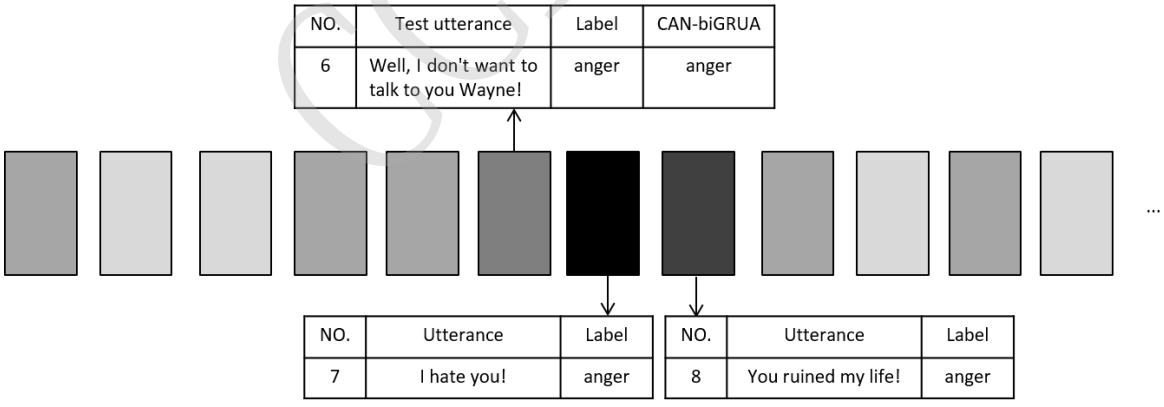
speaker	utterance	True label	bcGRU	CAN-biGRU
Phoebe	Oh, it's bad. It's really bad ... Which I do.	sadness	**sadness**	**sadness**
Chandler	How's your room Rach?	neutral	**neutral**	**neutral**
Rachel	Everything's ruined ... blue sweater.	sadness	neutral	**sadness**
Joey	Hey-hey-hey!	joy	**joy**	**joy**
Chandler	What are you doing?	neutral	**neutral**	**neutral**
Phoebe	We're just celebrating that Joey ... back.	joy	neutral	**joy**
Phoebe	I'm sorry ... Check this out.	neutral	sad	sad
Monica	No, Phoebe ... you play it at the wedding.	neutral	**neutral**	**neutral**

Table 5: Some case comparisons of emotion recognition results by bcGRU and CAN-biGRUA

in the third case, both two model make false predictions for the utterance said by Phoebe, since the word 'sorry' expresses strong sad emotion. This shows CAN is still limited in such complicated semantic environment.



(a) Recognition results and attention results of CAN-GRUA



(b) Recognition results and attention results of CAN-biGRUA

Figure 2: Comparison of recognition results and attention results between CAN-GRUA and CAN-biGRUA. Deeper color means higher attention.

As shown in Table 6, we analyse some cases of the results of emotion recognition by our CAN-biGRUA. In the first two cases, our model can successfully recognise the emotion category of utterances. In the conversation of Monica and Joey, 'Yeah' expresses neutral emotion, while in the conversation of Chloe and Ross, 'Yeah' with '!' expresses stronger emotion and our model analyses its joy emotion effectively. However, in the third case, our model makes wrong classification for the first,third and fourth utterances.

This dialogue contains three different emotions and emotion shifts frequently. The failure of our model indicates that although considering the context, model's ability to understand emotions in the complicated situation is limited and still needs improvement.

In Fig. 2, we compare the recognition results and attention results of CAN-GRUA and CAN-biGRUA for the sixth utterance in the dialogue. As we can see, CAN-GRUA only uses historical information and focuses on the fourth utterance, and it takes neutral emotion as a result. While CAN-biGRUA takes both historical and future information into account, and it mainly pays attention to the seventh and the eighth utterances which contain strongly anger emotion, so the model finnaly classifies the test utterance as anger emotion. This case shows that considering both historical information and future information can help model make better classifications.

speaker	utterance	True label	Predicted label
Monica	Hey, Joey, could you pass the cheese?	**neutral**	**neutral**
Joey	Yeah.	**neutral**	**neutral**
Chloe	That's so great for you guys!	**joy**	**joy**
Ross	Yeah!	**joy**	**joy**
Chloe	Good luck, with your girlfriend.	**neutral**	**neutral**
Monica	Ross, we can handle this.	neutral	joy
Ross	Well,... be hurt over something that is so silly.	**sadness**	**sadness**
Ross	I mean, enough of the silliness!	anger	sadness
Chandler	Well, why don't you tell her to stop being silly!	anger	sadness

Table 6: Some cases of emotion recognition results by CAN-biGRUA

5 Conclusion

In the paper, we propose a hierarchical model(CAN-GRU) to tackle emotion recognition in dialogues. Unlike existing works, we focus on semantic information extraction within utterance in the dialogue. N-gram features and relevant semantic information among words in the utterance are learned by the convolutional self-attention network in the first layer and the sequence of utterances is modeled by the GRU-based network in the second layer. We improve CAN-GRU to three variants, CAN-biGRU, CAN-GRUA and CAN-biGRUA. Experimental results show that attention mechanism can help to grasp long-term dependency in the contexts effectively. CAN-biGRUA achieves better results than CAN-GRUA demonstrates that it is necessary to consider both past and future information of the utterance.In the future, we will try to explore deeper semantic information in the context and focus more on emotion shift to solve the problem of poor performance of the model in complex situations.

Acknowledgements

The work of this paper is funded by the project of National key research and development program of China (No. 2018YFC0830700).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations*.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1724–1734. ACL.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American*

Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers).

Shang Gao, Arvind Ramanathan, and Georgia D. Tourassi. 2018. Hierarchical convolutional attention networks for text classification. In *Proceedings of The Third Workshop on Representation Learning for NLP, Rep4NLP@ACL*, pages 11–23. Association for Computational Linguistics.

Deepanway Ghosal, Navonil Majumder, Soujanya Poria, Niyati Chhaya, and Alexander F. Gelbukh. 2019. Dialoguegcn: A graph convolutional neural network for emotion recognition in conversation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 154–164. Association for Computational Linguistics.

Devamanyu Hazarika, Soujanya Poria, Rada Mihalcea, Erik Cambria, and Roger Zimmermann. 2018a. ICON: interactive conversational memory network for multimodal emotion detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2594–2604. Association for Computational Linguistics.

Devamanyu Hazarika, Soujanya Poria, Amir Zadeh, Erik Cambria, Louis-Philippe Morency, and Roger Zimmermann. 2018b. Conversational memory network for emotion recognition in dyadic dialogue videos. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2122–2132. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Chao-Chun Hsu and Lun-Wei Ku. 2018. Socialnlp 2018 emotionx challenge overview: Recognizing emotions in dialogues. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media, SocialNLP@ACL*, pages 27–31. Association for Computational Linguistics.

Sopan Khosla. 2018. Emotionx-ar: CNN-DCNN autoencoder based emotion classifier. In Lun-Wei Ku and Cheng-Te Li, editors, *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media, SocialNLP@ACL*, pages 37–44. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751. ACL.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*.

Linkai Luo, Haiqing Yang, and Francis Y. L. Chin. 2018. Emotionx-dlc: Self-attentive bilstm for detecting sequential emotions in dialogues. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media, SocialNLP@ACL*, pages 32–36. Association for Computational Linguistics.

Navonil Majumder, Soujanya Poria, Devamanyu Hazarika, Rada Mihalcea, Alexander F. Gelbukh, and Erik Cambria. 2019. Dialoguernn: An attentive RNN for emotion detection in conversations. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6818–6825. AAAI Press.

John D Mayer, Richard D Roberts, and Sigal G Barsade. 2008. Human abilities: emotional intelligence. *Annual review of psychology*, 59:507—536.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543. ACL.

Soujanya Poria, Erik Cambria, and Alexander F. Gelbukh. 2015. Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2539–2544. ACL.

Soujanya Poria, Erik Cambria, Devamanyu Hazarika, Navonil Majumder, Amir Zadeh, and Louis-Philippe Morency. 2017. Context-dependent sentiment analysis in user-generated videos. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers*, pages 873–883. Association for Computational Linguistics.

Rohit Saxena, Savita Bhat, and Niranjan Pedanekar. 2018. Emotionx-area66: Predicting emotions in dialogues using hierarchical attention network with sequence labeling. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media, SocialNLP@ACL*, pages 50–55. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems*, pages 5998–6008.

A Joint Model for Aspect-Category Sentiment Analysis with Shared Sentiment Prediction Layer

Yuncong Li [*], **Zhe Yang** [*], **Cunxiang Yin, Xu Pan** [†],
Lunan Cui, Qiang Huang and **Ting Wei**
Baidu Inc., Beijing, China
{liyuncong,yangzhe08,yincunxiang,panxu,cuilunan,huangqiang03,
weiting}@baidu.com

Abstract

Aspect-category sentiment analysis (ACSA) aims to predict the aspect categories mentioned in texts and their corresponding sentiment polarities. Some joint models have been proposed to address this task. Given a text, these joint models detect all the aspect categories mentioned in the text and predict the sentiment polarities toward them at once. Although these joint models obtain promising performances, they train separate parameters for each aspect category and therefore suffer from data deficiency of some aspect categories. To solve this problem, we propose a novel joint model which contains a shared sentiment prediction layer. The shared sentiment prediction layer transfers sentiment knowledge between aspect categories and alleviates the problem caused by data deficiency. Experiments conducted on SemEval-2016 Datasets demonstrate the effectiveness of our model.

1 Introduction

Aspect-category sentiment analysis (ACSA) is a subtask of aspect-based sentiment analysis (ABSA) (Pontiki et al., 2014; Pontiki et al., 2015; Pontiki et al., 2016). ACSA aims to identify all the aspect categories mentioned in texts and their corresponding sentiment polarities. An aspect category (or simply aspect) is an entity E and attribute A pair, denoted by E#A. For example, in the text "The place is small and cramped but the food is fantastic.", the aspect categories mentioned in the text are *AMBIENCE#GENERAL* and *FOOD#QUALITY*, and their sentiment polarities are negative and negative, respectively.

Many methods have been proposed to address the ACSA task. However, most existing methods (Zhou et al., 2015; Movahedi et al., 2019; Wang et al., 2016; Ruder et al., 2016; Cheng et al., 2017; Xue and Li, 2018; Tay et al., 2018) divide the ACSA task into two subtasks: aspect category detection (ACD) which detects aspect categories in a text and sentiment classification (SC) which categorizes the sentiment polarities with respect to the detected aspect categories, and perform these two tasks separately. Such two-stage approaches lead to error propagation, that is, errors caused by aspect category detection would affect sentiment classification. To avoid error propagation, previous studies (Schmitt et al., 2018; Hu et al., 2019; Wang et al., 2019) have proposed some joint models, which jointly model the detection of aspect categories and the classification of their polarities. Further more, given a text, these joint models detect all the aspect categories mentioned in the text and predict the sentiment polarities toward them at once. Although these joint models obtain promising performances, they train separate parameters for each aspect category and therefore suffer from data deficiency of some aspect categories. For example, the english laptops domain dataset from SemEval-2016 task 5: Aspect-based Sentiment Analysis (Pontiki et al., 2016) has a quarter of the aspect categories whose sample size are less than or equal to 2 (see Table 2). Previous joint models will under-fit on the aspect categories with deficient samples.

To solve the problem caused by data deficiency mentioned above, we propose a novel joint model, which contains a shared sentiment prediction layer. Our model is based on the observation that

[*]Equal contribution.
[†]Corresponding author.

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1112-1121, Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

Aspect Category	Text	Polarity
LAPTOP#QUALITY	...I was **surprised** at the overall quality and the price...	positive
LAPTOP#PRICE		positive
LAPTOP#OPERATION_PERFORMANCE	...I was **surprised** with the performance and quality of this HP Laptop...	positive
LAPTOP#QUALITY		positive

Table 1: Different aspect categories have the same sentiment word which have the same sentiment polarity.

the sentiment expressions and their polarities of different aspect categories are transferable. For instance, in Table 1, the three aspect categories *LAPTOP#QUALITY*, *LAPTOP#PRICE*, and *LAPTOP#OPERATION_PERFORMANCE* have the same sentiment word "surprised" and the consistent polarity. The shared sentiment prediction layer transfers sentiment knowledge between aspect categories and alleviates the problem caused by data deficiency.

In summary, the main contributions of our work are as follows:

- We propose a novel joint model for the aspect category sentiment analysis (ACSA) task, which contains a shared sentiment prediction layer. The shared sentiment prediction layer transfers sentiment knowledge between aspect categories and alleviates the problem caused by data deficiency.
- Experiments conducted on SemEval-2016 Datasets demonstrate the effectiveness of our model.

2 Related Work

Existing methods for Aspect-Category Sentiment Analysis (ACSA) can be divided into two categories: two-stage methods and joint models.

Two-stage methods perform the ACD task and the SC task separately. Zhou et al. (2015) and Movahedi et al. (2019) perform the ACD task. Zhou et al. (2015) propose a semi-supervised word embedding algorithm to obtain word embeddings on a large set of reviews, which are then used to generate deeper and hybrid features to predict the aspect category. Movahedi et al. (2019) utilize topic attention to attend to different aspects of a given text. Many methods (Wang et al., 2016; Ruder et al., 2016; Cheng et al., 2017; Xue and Li, 2018; Tay et al., 2018; Liang et al., 2019; Jiang et al., 2019; Xing et al., 2019; Sun et al., 2019; Zhu et al., 2019; Lei et al., 2019) have been proposed for the SC task. Wang et al. (2016) first propose aspect embedding (AE) and use an Attention-based Long Short-Term Memory Network (AT-LSTM) to generate aspect-specific text representations for sentiment classification based on aspect embedding. Ruder et al. (2016) propose a hierarchical bidirectional LSTM (H-LSTM) to modeling the interdependencies of sentences in a review. Tay et al. (2018) propose a method named Aspect Fusion LSTM (AF-LSTM) to model word-aspect relationships. Xue and Li (2018) propose a model, namely Gated Convolutional network with Aspect Embedding (GCAE), which incorporates aspect information into the neural model by gating mechanisms. Jiang et al. (2019) proposed new capsule networks to model the complicated relationship between aspects and contexts. All the two-stage methods have the problem of error propagation.

Joint models jointly model the detection of aspect categories and the classification of their polarities. Only a few joint models (Schmitt et al., 2018; Hu et al., 2019; Wang et al., 2019) have been proposed for ACSA. Schmitt et al. (Schmitt et al., 2018) propose two joint models: End-to-end LSTM and End-to-end CNN, which produce all the aspect categories and their corresponding sentiment polarities at once. Hu et al. (2019) propose constrained attention networks (CAN), which extends AT-LSTM to multi-task settings and introduces orthogonal and sparse regularizations to constrain the attention weight allocation. As a result, the CAN achieves better sentiment classification performance. However, to train the CAN, we need to annotate the multi-aspect sentences with overlapping or nonoverlapping. Wang et al. (2019) propose the aspect-level sentiment capsules model (AS-Capsules), which utilizes the correlation between aspect category and sentiment through shared components including capsule

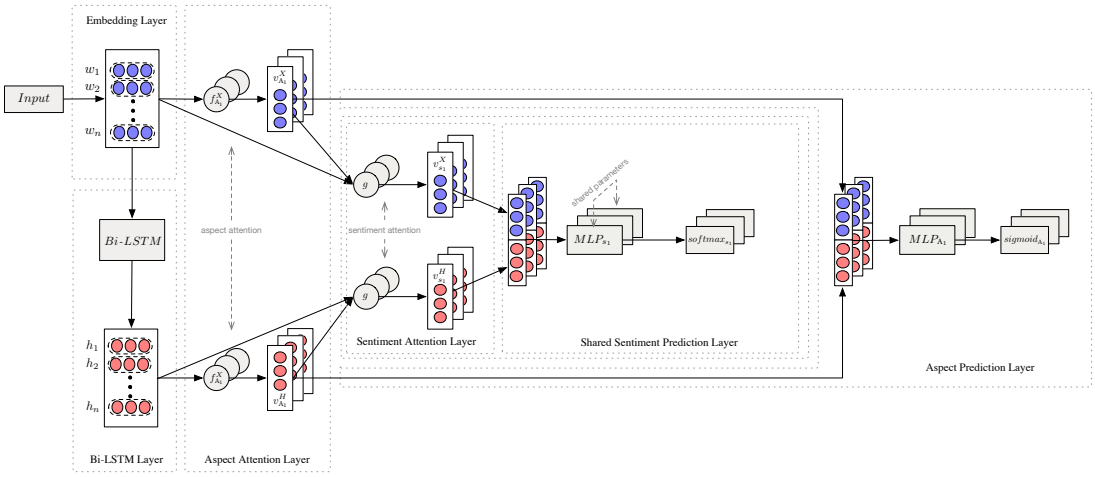


Figure 1: Overall architecture of the proposed method.

embedding, shared encoders, shared attentions and a shared recurrent neural network. These joint models train separate parameters for each aspect category, which results in that these models under-fit on the aspect categories with deficient samples.

3 Proposed Model

We first formulate the problem. There are N predefined aspect categories $A = \{A_1, A_2, \cdots, A_N\}$ and M predefined sentiment polarities $P = \{P_1, P_2, \cdots, P_M\}$ in the dataset. Given a sentence or a review, denoted by $S = \{w_1, w_2, \cdots, w_n\}$, the task aims to predict the aspect categories and the corresponding sentiment polarities, i.e., aspect-sentiment pairs$\{< A_j, P_k >\}$, expressed in the text. The overall model architecture is illustrated in Figure 1, which contains six modules: embedding layer, Bi-LSTM layer, aspect attention layer, sentiment attention layer, aspect category prediction layer, and shared sentiment prediction layer. Then, we display the details of each module and introduce the training objective function.

3.1 Embedding Layer

The input to our model is a text consisting of n words $\{w_1, w_2, ..., w_n\}$. With a embedding matrix U, the input text is converted to a sequence of vectors $X = \{x_1, x_2, ..., x_n\}$. Where $U \in R^{d_w \times |V|}$, d_w is the dimension of the word embeddings, and $|V|$is the vocabulary size.

3.2 Bidirectional LSTM Layer

The word embeddings of the text are then fed into a Bidirectional LSTM (Graves et al., 2013) network (Bi-LSTM) with two LSTM (Hochreiter and Schmidhuber, 1997) networks. We can obtain two hidden representations, and then concatenate the forward hidden state and backward hidden state of each word. Formally, given the sequence of vectors $X = \{x_1, x_2, \cdots, x_n\}$, Bi-LSTM outputs hidden states $H = \{h_1, h_2, \cdots, h_n\}$. At each time step $i = 1, 2, \cdots, n$, the hidden state h_i of the Bi-LSTM is computed by:

$$\overrightarrow{h_i} = \overrightarrow{LSTM}(\overrightarrow{h}_{i-1}, x_i) \tag{1}$$

$$\overleftarrow{h_i} = \overleftarrow{LSTM}(\overleftarrow{h}_{i+1}, x_i) \tag{2}$$

$$h_i = [\overrightarrow{h_i}, \overleftarrow{h_i}] \tag{3}$$

where $\overrightarrow{h_i} \in R^{d_s}$,$\overleftarrow{h_i} \in R^{d_s}$, $h_i \in R^{2d_s}$, and d_s denotes the size of the hidden state of LSTM.

3.3 Aspect Attention Layer

This layer applies an attention mechanism on the outputs of both the embedding layer and the Bi-LSTM layer and generates aspect-specific representations for the ACD task. Different aspect categories have different attention parameters. The process can be formulated as follows:

$$v_{A_j}^X = f_{A_j}^X(X), j = 1, \cdots, N \tag{4}$$

$$v_{A_j}^H = f_{A_j}^H(H), j = 1, \cdots, N \tag{5}$$

where f(·) is an attention mechanism (Yang et al., 2016) and can be defined as follows:

$$f(V) = v = \Sigma_{i=1}^n \alpha_i v_i \tag{6}$$

$$u_i = tanh(W_a v_i + b_a) \quad for \quad i = 1, 2, \cdots, n \tag{7}$$

$$\alpha_i = \frac{exp(u_i^T u_w)}{(\Sigma_{j=1}^n exp(u_j^T u_w)} \quad for \quad i = 1, 2, \cdots, n \tag{8}$$

where $V = \{v_1, \cdots, v_i, \cdots, v_n\}$ is a sequence of vectors and $v_i \in R^d$. $W_a \in R^{m \times d}$,$b_a \in R^m$, and $u_w \in R^m$ are the parameters of the attention mechanism. m is the dimensionality of the attention context vector, and d is the dimensionality of the input vector. Note that the vector v generated by $f(V)$ is a weighted sum of vectors in V and is in the same semantic space with them.

3.4 Aspect Category Prediction Layer

Aspect category prediction layer takes as input the concatenation of the aspect-specific representations at the embedding layer and the Bi-LSTM layer for the ACD task and predicts whether the text mentions the aspect categories. Formally, for the j-th aspect category:

$$v_{A_j} = [v_{A_j}^X, v_{A_j}^H] \tag{9}$$

$$\widehat{y}_{A_j} = \sigma(\widehat{W}_{A_j} ReLU(W_{A_j} v_{A_j} + b_{A_j}) + \widehat{b}_{A_j}) \tag{10}$$

$$\sigma(x) = \frac{1}{(1 + e^{-x})} \tag{11}$$

where W_{A_j},b_{A_j}, $\widehat{W}_{A_j}$, and $\widehat{b}_{A_j}$ are the parameters of the j-th aspect category. If $\widehat{y}_{A_j}$ is greater than the specified threshold τ, we judge that the j-th aspect category is mentioned by the text.

3.5 Sentiment Attention Layer

This layer generates aspect-specific text representations for the SC task based on aspect-specific text representations for the ACD task. For the j-th aspect category, its aspect-specific text representations for the SC task can be computed as follows:

$$v_{s_j}^X = g(X, v_{A_j}^X) \tag{12}$$

$$v_{s_j}^H = g(H, v_{A_j}^H) \tag{13}$$

where X and H are the outputs of the embedding layer and the Bi-LSTM layer, respectively. $v_{A_j}^X$ and $v_{A_j}^H$ are the aspect-specific text representations of the j-th aspect category for the ACD task at the outputs of

the embedding layer and the Bi-LSTM layer respectively. g(·) is an attention mechanism (Vaswani et al., 2017) and can be defined as follows:

$$v_s = g(X, v_q) \tag{14}$$

$$\beta_i = \frac{exp(x_i^T v_q)}{\Sigma_{j=1}^{n} x_j^T v_q} \quad for \quad i = 1, 2, \cdots, n \tag{15}$$

$$v_s = \Sigma_{i=1}^{n} \beta_i x_i \tag{16}$$

where X is a sequence vectors $\{x_1, x_2, \cdots, x_n\}$, and v_q is the query vector of the attention. We use the dot product to compute attention weights because it does not import extra aspect-specific parameters. Since the query vector and the key vector of the attention are in the same semantic space in our model, the dot product is reasonable.

3.6 Shared Sentiment Prediction Layer

The aspect-specific text representation of the j-th aspect for the SC task is generated by concatenating the aspect-specific text representations of the j-th aspect category for the SC task at the outputs of the embedding layer and the Bi-LSTM layer. The representation are then fed to a fully connected layer with the ReLU activation function and then the output of the fully connected layer is fed to another fully connected layer with the softmax activation function to generate sentiment probability distribution. Formally, for the j-th aspect category:

$$v_{s_j} = [v_{s_j}^X, v_{s_j}^H] \tag{17}$$

$$\widehat{y}_{s_j} = softmax(\widehat{W}_s ReLU(W_s v_{s_j} + b_s) + \widehat{b}_s) \tag{18}$$

$$softmax(x)_i = \frac{exp(x_i)}{\Sigma_{k=1}^{M} exp(x_k)} \tag{19}$$

where W_s , b_s , $\widehat{W}_s$, and $\widehat{b}_s$ are the shared parameters of all aspect categories.

3.7 Loss

For the aspect category detection task, as each prediction is a binary classification problem, the loss function is defined by:

$$L_A(\theta) = -\Sigma_{j=1}^{N} y_{A_j} log\widehat{y}_{A_j} + (1 - y_{A_j}) log(1 - \widehat{y}_{A_j}) \tag{20}$$

For the sentiment classification task, the loss function is defined by:

$$L_s(\theta) = -\Sigma_{j=1}^{N} \Sigma_{k=1}^{M} y_{s_{j_k}} log(\widehat{y}_{s_{j_k}}) \tag{21}$$

if the j-th aspect category is not mentioned in the text, $y_{s_{j_k}} = 0$ for $k = 1, 2, \cdots, M$.

We jointly train our model for the two tasks. The parameters in our model are then trained by minimizing the combined loss function:

$$L_s(\theta) = L_A(\theta) + \eta L_s(\theta) + \lambda ||\theta||_2^2 \tag{22}$$

where η is the weight of sentiment classification loss, λ is the L2 regularization factor and θ contains all the parameters except for Bi-LSTM layer's parameters. Furthermore, to avoid over-fitting, we adopt the dropout strategy to enhance our model.

Dataset	#aspect	#polarity	#train	#val	#test	#min	#Q1	#Q2	#Q3
CH-CAME-SB1	75	2	1090	169	481	1.0	1.0	6.0	13.0
CH-PHNS-SB1	81	2	1152	181	529	1.0	1.8	4.5	23.3
EN-LAPT-SB2	88	4	355	40	80	1.0	2.0	7.0	20.0
EN-REST-SB2	12	4	301	34	90	20.0	36.5	68.5	177.0

Table 2: Statistics of the datasets. #aspect and #polarity represent the number of predefined aspects and sentiment polarities, respectively. #train, #dev, and #test represent the sample size of training sets, validation sets, and test sets, respectively. #min indicates the minimum value of the aspect sample size. #Q1, #Q2, #Q3 are the first quartile, the second quartile, and the third quartile of the aspect sample size, respectively.

4 Experiments

4.1 Datasets

We conduct experiments on four public datasets from SemEval-2016 task 5: Aspect-based Sentiment Analysis (Pontiki et al., 2016):

CH-CAME-SB1 is a Chinese sentence-level dataset about digital cameras domain.

CH-PHNS-SB1 is a Chinese sentence-level dataset about mobile phones domain.

EN-REST-SB2 is an English review-level dataset about restaurants domain.

EN-LAPT-SB2 is an English review-level dataset about laptops domain.

We randomly split the original training set into training, validation sets in the ratio 9:1. We use quartiles to measure the distribution of the sample size of aspects in these datasets. Detailed statistics are summarized in Table 2. Particularly, for the three datasets, CH-CAME-SB1, CH-PHNS-SB1, and EN-LAPT-SB2, the sample size of 50% of the aspects are no more than 7.

4.2 Evaluation Metrics

We use micro-averaged F1-scores as the evaluation metric for both the ACSA and the ACD:

$$F_1 = \frac{2 * P * R}{P + R} \tag{23}$$

where precision and recall are defined as:

$$P = \frac{|S \cap G|}{|S|} \tag{24}$$

$$R = \frac{|S \cap G|}{|G|} \tag{25}$$

Here S is the set of aspect-sentiment pairs or aspect category annotations (in ACSA and ACD, respectively) that a model returns for all the test texts, and G is the set of the gold (correct) aspect-sentiment pairs or aspect category annotations. To evaluate the SC task, we use the gold aspect category annotations to select sentiment polarities model predicts and calculated the accuracy.

4.3 Comparison Methods

We select the following methods for comparison.

End-to-end LSTM (Schmitt et al., 2018) performs the ACSA task, which jointly models the detection of aspects and the classification of their polarities in an end-to-end trainable neural network.

End-to-end CNN (Schmitt et al., 2018) is an CNN version of End-to-end LSTM, which replaces the Bi-LSTM in End-to-end LSTM with a convolutional neural network (CNN) described in (Kim, 2014).

AS-Capsules (Wang et al., 2019) utilizes the correlation between aspect category and sentiment through shared components including capsule embedding, shared encoders, shared attentions and a shared recurrent neural network.

Models	CH-CAME-SB1	CH-PHNS-SB1	EN-LAPT-SB2	EN-REST-SB2
End-to-end cnn	34.96	19.87	36.52	66.03
End-to-end lstm	41.52	26.30	37.94	63.75
AS-Capsules	38.85	27.05	33.47	63.99
Our Model	**42.01**	**28.98**	**50.05**	68.24
– w/o Share	36.23	22.72	49.43	**68.28**

Table 3: Results of the ACSA task in terms of micro-averaged F1-scores(%).

Models	CH-CAME-SB1	CH-PHNS-SB1	EN-LAPT-SB2	EN-REST-SB2
SemEval-2016 Best	80.45	73.34	75.05	81.93
End-to-end cnn	70.55	64.15	69.42	80.78
End-to-end lstm	75.12	67.36	72.00	80.03
AS-Capsules	76.96	71.56	69.05	76.794
Our Model	**82.54**	**76.50**	**75.91**	82.43
– w/o Share	69.09	59.86	70.53	**83.33**

Table 4: Results of the SC task in terms of accuracy(%).

SemEval-2016 Best is the best model for each subtask of SemEval-2016 task 5: Aspect based Sentiment Analysis (Pontiki et al., 2016).

Our Model – w/o Share was added to show the effectiveness of the shared sentiment prediction layer, which trains a separate sentiment prediction layer for each aspect.

4.4 Implementation Details

We implement all models in Keras. We set $\lambda = 0.01$ and gradient clipping norm to 5. Adam (Kingma and Ba, 2014) optimizer is applied to minimize the loss. We apply a dropout of $p = 0.5$ after the embedding layer and the Bi-LSTM layer. Hidden layer size for Bi-LSTM is 100. We use 300-dimensional word embeddings. We use GloVe (Pennington et al., 2014) embeddings which are pre-trained on an unlabeled corpus whose size is about 840 billion for English and Skip-Gram (Mikolov et al., 2013) embeddings which are pre-trained on the Baidu Encyclopedia dataset for Chinese. If an aspect is not mentioned, its corresponding sentiment label is set to a zero vector. We set threshold $\tau = 0.25$ for aspect category detection. While batch size is 32 on CAME-SB1 and CH-PHNS-SB1, batch size is 10 on EN-LAPT-SB2 and EN-REST-SB2. The sentiment classification loss weight is 1 on CH-CAME-SB1, CH-PHNS-SB1, and EN-LAPT-SB2, and is 0.6 on EN-REST-SB2. To reduce the randomness of results, we train each model three times and report their averaged scores.

4.5 Results

Table 3, Table 4, and Table 5 show our experimental results on the ACSA, SC, and ACD tasks, respectively. The best results are marked in bold.

Models	CH-CAME-SB1	CH-PHNS-SB1	EN-LAPT-SB2	EN-REST-SB2
SemEval-2016 Best	36.3	22.5	60.4	**83.9**
End-to-end cnn	47.83	26.64	42.25	76.20
End-to-end lstm	**52.98**	33.81	43.81	76.24
AS-Capsules	48.72	33.66	40.99	78.29
Our Model	51.81	**36.67**	**62.91**	81.65
– w/o Share	52.16	36.54	62.72	81.45

Table 5: Results of the ACD task in terms of micro-averaged F1-scores(%).

	Text	Label	Our model	-w/o Share
Train set	...The only objection I have is that after you buy it the windows 7 system is a starter and charges for the upgrade...	negtive		
	...The flaws are, this computer is not for computer gamers because of the OS X...	negtive		
Test set	...The OS is easy, and offers all kinds of surprises...	positive	positive	negtive
	...The free upgrade to Mountain Lion FAILED...	negtive	negtive	negtive

Table 6: Impact of the shared sentiment prediction layer on the sentiment prediction of the aspect category *OS#MISCELLANEOUS*.

Table 3 shows the experimental results on the ACSA task, which show the overall performance of our joint model. We observe that our proposed joint model outperforms the baseline models on all datasets, which demonstrates the effectiveness of our model.

The experimental results on the SC task are in Table 4. First, we observe that our model surpasses all baseline models on all datasets, which indicates the effectiveness of our model predicting the sentiment polarities toward given aspect categories. Second, our model outperforms its variant (– w/o Share) by 13.45%, 16.64% and 5.38% on CH-CAME-SB1, CH-PHNS-SB1, and EN-LAPT-SB2 datasets, respectively. The reason is that the three datasets have many aspect categories which only have a few instances and benefit from parameter sharing. This shows that our shared parameter prediction layer can alleviate the problem caused by data deficiency. Meanwhile, our model obtains worse perfromance than its variant (– w/o Share) on the EN-REST-SB2 dataset. The possible reason is that the sample size of the aspect categories in the EN-REST-SB2 dataset is enough to train independent sentiment prediction parameters, and parameter sharing brings some noise between aspect categories.

Table 5 shows the results on the ACD task. Although we did not specifically optimize our model for the ACD task, our model still achieves competitive performance. Specifically, our model outperforms all baselines on the CH-PHNS-SB1 and EN-LAPT-SB2 datasets.

4.6 Case Studies

To have an intuitive understanding of our proposed shared sentiment prediction layer for the SC task, we use the EN-LAPT-SB2 dataset to illustrate the impact of knowledge transferring. The selected aspect category from the dataset is *OS#MISCELLANEOUS*. There are only two samples with both negative polarities in the training set, while there are two samples in the test set, whose polarities are negative and positive, respectively. Table 6 shows that our model can correctly predict the polarity of the sample with positive sentiment in the test set. After removing the shared sentiment prediction layer, – w/o Share fails to predict the polarity of the sample with positive sentiment, which confirms the importance of the shared sentiment prediction layer.

5 Conclusion

In this work, we propose a novel joint model which contains a shared sentiment prediction layer. The shared sentiment prediction layer transfers sentiment knowledge between aspect categories and alleviates the problem caused by data deficiency. Experiments conducted on four datasets from SemEval-2016 task 5 demonstrate the effectiveness of our model. Furture work could consider introducing extra component that prevents the shared sentiment prediction layer from transfering aspect-specific sentiment knowledge.

References

Jiajun Cheng, Shenglin Zhao, Jiani Zhang, Irwin King, Xin Zhang, and Hui Wang. 2017. Aspect-level sentiment classification with heat (hierarchical attention) network. In *Proceedings of the 2017 ACM on Conference on*

Information and Knowledge Management, pages 97–106.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Mengting Hu, Shiwan Zhao, Li Zhang, Keke Cai, Zhong Su, Renhong Cheng, and Xiaowei Shen. 2019. Can: Constrained attention networks for multi-aspect sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4593–4602.

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6281–6286.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Zeyang Lei, Yujiu Yang, Min Yang, Wei Zhao, Jun Guo, and Yi Liu. 2019. A human-like semantic cognition network for aspect-level sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6650–6657.

Yunlong Liang, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2019. A novel aspect-guided deep transition model for aspect based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5572–5584.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Sajad Movahedi, Erfan Ghadery, Heshaam Faili, and Azadeh Shakery. 2019. Aspect category detection via topic-attention network. *arXiv preprint arXiv:1901.01183*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, August. Association for Computational Linguistics.

Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 486–495.

Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, AL-Smadi Mohammad, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphee De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30.

Sebastian Ruder, Parsa Ghaffari, and John G Breslin. 2016. A hierarchical model of reviews for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 999–1005.

Martin Schmitt, Simon Steinheber, Konrad Schreiber, and Benjamin Roth. 2018. Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1109–1114.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 380–385.

Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 606–615.

Yequan Wang, Aixin Sun, Minlie Huang, and Xiaoyan Zhu. 2019. Aspect-level sentiment analysis using as-capsules. In *The World Wide Web Conference*, pages 2033–2044.

Bowen Xing, Lejian Liao, Dandan Song, Jingang Wang, Fuzheng Zhang, Zhongyuan Wang, and Heyan Huang. 2019. Earlier attention? aspect-aware lstm for aspect sentiment analysis. *arXiv preprint arXiv:1905.07719*.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2514–2523.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2015. Representation learning for aspect category detection in online reviews. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Peisong Zhu, Zhuang Chen, Haojie Zheng, and Tieyun Qian. 2019. Aspect aware learning for aspect category sentiment analysis. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 13(6):1–21.

Compress Polyphone Pronunciation Prediction Model with Shared Labels

Pengfei Chen, Lina Wang, Hui Di, Kazushige Ouchi and Lvhong Wang
Research & Development Center, Toshiba, China
{chenpengfei, wanglina, dihui}@toshiba.com.cn
kazushige.ouchi@toshiba.co.jp
wlv1990@gmail.com

Abstract

It is well known that deep learning model has huge parameters and is computationally expensive, especially for embedded and mobile devices. Polyphone pronunciations selection is a basic function for Chinese Text-to-Speech (TTS) application. Recurrent neural network (RNN) is a good sequence labeling solution for polyphone pronunciation selection. However, huge parameters and computation make compression needed to alleviate its disadvantages. Meanwhile, Large-scale-labels classification leads to more complicated network and heavy computation cost. In contrast to existing quantization with low precision data format and projection layer, we propose a novel method based on shared labels, which focuses on compressing the fully-connected layer before Softmax for models with a huge number of labels in TTS polyphone selection. The basic idea is to compress large number of target labels into a few label clusters, which will share the parameters of fully-connected layer. Furthermore, we combine it with other methods to further compress the polyphone pronunciation selection model. The experimental result shows that for Bi-LSTM (Bidirectional Long Short Term Memory) based polyphone selection, shared labels model decreases about 52% of original model size and accelerates prediction by 44% almost without performance loss.

It is worth mentioning that the proposed method can be applied for other tasks to compress model and accelerate calculation.

Keywords: Bi-LSTM, Polyphone Pronunciation Prediction, Model Compression, Shared Labels.

1 Introduction

Polyphone pronunciation prediction is a basic module of G2P (Grapheme-to-Phoneme) in Chinese Text-to-Speech (TTS) system, which provides the right pronunciation for Chinese character. The algorithms of polyphone pronunciation prediction include dictionary-based algorithm, statistical machine learning-based algorithm like Conditional Random Field (CRF) in (Lafferty et al., 2001), and deep learning-based algorithm like RNN. Dictionary-based method may fail for polyphone words problem, such as "朝阳" can be read as "chao2yang2" and "zhao1yang2", and Out of Vocabulary(OOV) problem. CRF and RNN perform well for polyphone pronunciation selection with context features. However, CRF needs manually designed context features. Neural network always has more parameters, larger model size and more expensive computation cost. In the application for embedded device, small model size and quick computation are necessary. So compression and acceleration of neural network is a hot research field in recent years.

There are several methods to compress deep learning model: low precision of data format, quantification, pruning, low rank factorization, and knowledge distillation. Low precision of data format replaces double or float with 16-bit float, which can reduce model size by quarter or half, but it cannot reduce running time when corresponding computations are not supported by existing instruction sets. Quantification and pruning methods pack some weights into one and prune the weights close to zero. It can

reduce the model size but cannot accelerate the computation. Low rank factorization changes the network structure by factorizing large matrix into small matrixes. It can reduce the model size and accelerate the computation.

Our interest focuses on how to compress model that contains huge number of labels, and each input has a fixed label set. For example, each polyphone Chinese character has several (less than 10) fixed pronunciations. If we take all pronunciations of polyphone characters as target labels, there are too many parameters in fully-connected layer before Softmax. Therefore, the computation of fully-connected layer and Softmax will be costly. One idea is to use 1 character-1 model method, which can reduce the network complexity and computation cost, but it needs larger memory because of too many polyphone characters. This inspired us how we can share parameters in a single model which has comparable size to one-character model.

We propose a method to share parameters by means of different characters sharing pronunciation labels. We randomly assign labels for each character's pronunciation under conditions of avoiding label conflicts and assuring that the target label set is small. This yields smaller model size and less computation cost. Then we train our Bi-LSTM model with newly tagged corpus, and compare it with other compression methods in model size, memory usage and decoding time. The experimental results show our method can compress the model to half size and accelerate computation speed while maintaining a comparable performance compared with original model, overcoming the problem of too many labels.

2 Related Work

2.1 Bi-LSTM for sequence labeling

LSTM (Hochreiter and Schmidhuber, 1997) is excellent in sequence labeling by learning contextual information. Bi-LSTM can use the future and history information to improve performance. LSTM encodes the embedding of input sequentially into a vector, which will keep the history information. Bi-LSTM will concatenate the vectors of forward LSTM and backward LSTM, which can utilize future and history information. Followed by fully-connected layer and Softmax, Bi-LSTM model can predict the label with the highest probability. (Lample et al., 2016) presented a neural architectures based on Bi-LSTM and CRF to predict name entity. (Cai et al. , 2019) described a system composed of Bi-LSTM acting as an encoder and a prediction network for Chinese polyphone pronunciation prediction. The output size equals the number of all possible pronunciations of polyphone character.

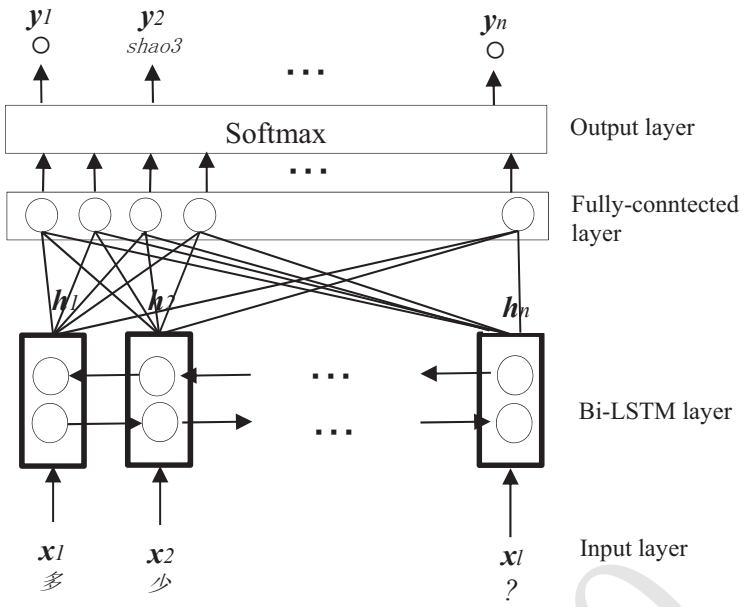


Figure 1: Polyphone pronunciation sequence labeling by Bi-LSTM

There is a long history of model compression in Nature Language Processing (NLP) tasks. Many compression approaches have been proposed, including low precision data format and quantization, network pruning and parameter sharing, tensor decomposition, knowledge distillation. We briefly review the most popular methods in this section.

- Low precision data format and Quantization

Low precision data format can reduce the model size exponentially. Quantization compresses value to a less bits' data to reduce the number of bytes of the weight parameter. For example, we can compress a float value to an 8-bit integer, one of 256 equally-sized intervals within the range. 16-bit, 8-bit, and even 1-bit quantization were proposed to compress network. (Gupta et al. , 2015) used 16-bit wide fixed-point number representation to train neural network, without degradation in the accuracy of classification tasks. (Courbariaux and Bengio, 2016) proposed a binarized Neural Networks (BNNs) with binary weights and activations, which can drastically reduce memory size, without any loss in classification accuracy.

- Network pruning and parameter sharing

Pruning can remove the parameters below threshold from network. Parameter sharing groups weights into hash brackets for sharing. Network pruning and parameter sharing not only can reduce the structure complexity of model, but also can improve generalization of network.

(Grachev et al ., 2017) introduced pruning to network compression for language modeling, and compared performance and model size with baseline model. (Han et al. , 2015) trained a network to learn which connections are important firstly. Then they pruned the unimportant connections and retrained the network to fine tune the weights of the remaining connections. Their method improved the energy efficiency and storage of neural networks without affecting accuracy. (Chen et al. , 2015) presented HashedNets which used a low-cost hash function to group weights into hash buckets, and all weights within the same hash bucket share a single parameter value.

- Tensor decomposition

Tensor decomposition approaches can factorize weight matrix into smaller matrixes to reduce the number of parameters, which can compress the size of model and accelerate the calculation.

(Grachev et al ., 2017) compared low-rank (LR) factorization and tensor train (TT) decomposition in LSTM compression in language modeling. In their result, LR LSTM 650-650 is the most useful model for practical application.

- Knowledge distillation

(Hinton et al. , 2015) first proposed knowledge distillation to transfer knowledge from teacher model to student model. This method can be used to improve the compressed model (student model) by exploiting original model (teacher model).

3 Shared Labels model

3.1 Framework

Next we will introduce how to implement our novel proposal. For polyphone Chinese characters, there are as many as 1304 pronunciations in our corpus. 1305 labels, including "O" for non-polyphone characters, are used in our original model, which leads to the facts that the fully-connected layer before Softmax contains a large proportion of the weights and Softmax computation is costly. Considering the number of any polyphone pronunciations is less than 10, only 10 shared labels are used in our novel proposal to reduce the memory usage and computational cost.

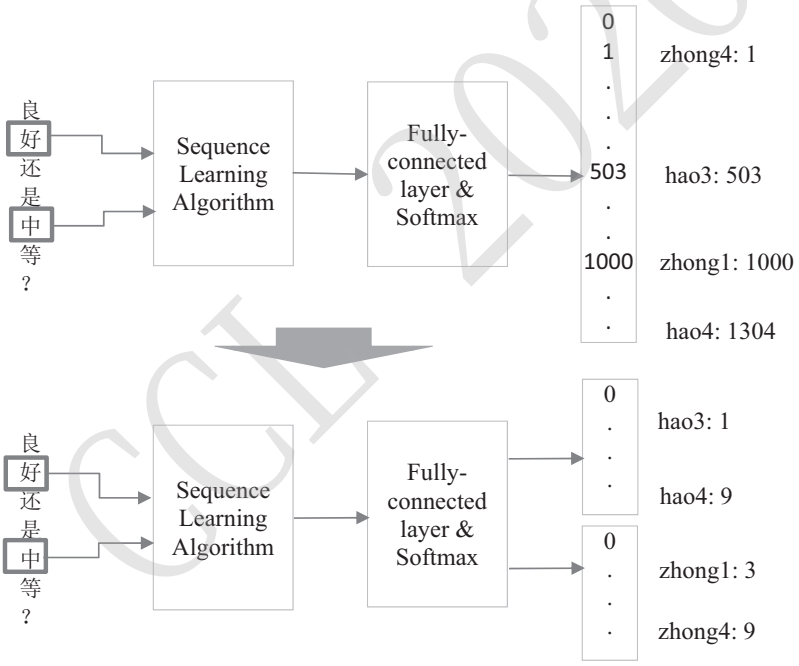


Figure 2: Illustration of shared labels model

In our proposal, we use a character-based model for polyphone pronunciation prediction. We take a sentence as the input, such as "良好还是中等?". Each character's embedding comes from pre-trained embedding model. The inputs are fed into sequence learning algorithm to learn semantics and features, and the output label is predicted with 1 fully-connected layer and Softmax. The sequence learning method includes but not limited to Bi-LSTM. For the output layer, the original 1305 labels are mapped to 10 digital shared labels. For example, labels "hao3" and "hao4" for "好" are mapped to label-1 and label-9 respectively. The prediction is based on the 10 shared labels.

The digital labels for different polyphone characters can be the same but their real pronunciations may be different. For example, in the above sentence, "好hao4" and "中zhong4" share the digital label "9". We can translate the digital shared labels into real pronunciations with a dictionary based on the method of the following section.

3.2 Theoretical Analysis

Theoretically, the framework can reduce parameter number and accelerate computation speed. We use Bi-LSTM as sequence learning algorithm. Given sequence length *L*, character embedding size *V*, LSTM hidden size *H*, target label number *N*, the parameter number of Bi-LSTM layer is

$$N_1 = 2 * 4 * (V * H + H * H + H) \tag{1}$$

The parameter number of Fully-connected layer is

$$N_2 = 2 * H * N \tag{2}$$

The parameter number of model is the sum of these two parts:

$$N_{para} = N_1 + N_2 \tag{3}$$

For our polyphone pronunciation prediction task, the model is always not complex. In our implementation, we set *L* as 100, *V* as 100, *H* as 200 and *N* as 1305. The total parameter number of baseline model is 1,003,600, and that of shared labels model is 485,600, which reduces about 52% of size.

We take multiply-accumulate operations (MACCs) as measure of computations. One MACC includes one multiplication and one addition.

For vector multiplication:

$$y = w_0 * x_0 + w_1 * x_1 + \cdots + w_{n-1} * x_{n-1} \tag{4}$$

w and *x* are two vectors, result *y* is a scalar.

A dot-product between two vectors of size *n* uses *n* MACCs. For a sequence of length *L*, the total MACCs of our model is

$$N_{MACCs} = L * [2 * 4 * (V + H) * H + 2 * H * N] \tag{5}$$

According the equation (5), the total MACCs of baseline model is 100.2 million. In our shared labels method, the total MACCs is 48.4 million, reducing about 52% compared with the baseline model.

If we take Floating Point Operations (FLOPs) as measure, there will be more reduction in computation because of less operations in Softmax.

3.3 Modules of polyphone pronunciation selection with shared labels

In the training phase, we need to convert pronunciation to digital label. As mentioned above, we map 1305 labels to 10 shared labels. The mapping relations are saved in two dictionaries consisting of character-pronunciation-label_ID and character-label_ID-pronunciation. Details will be described in the next section. Then we handle the corpus with shared labels. We train Bi-LSTM model with pre-processed corpus.

In inference phase, we get the digital label from the prediction of model. Then we replace the digital label with real pronunciation by looking up dictionary.

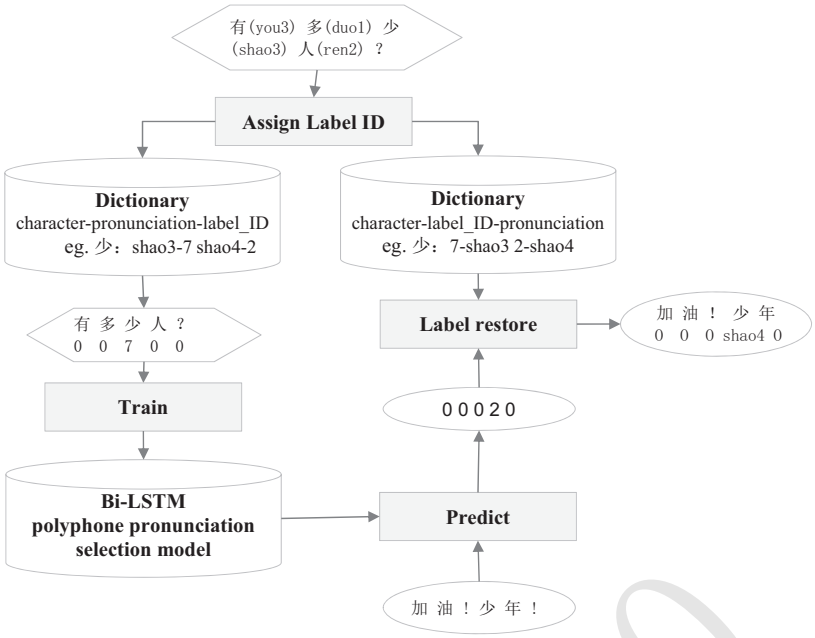


Figure 3: Modules of polyphone pronunciation selection with shared labels model

- Assign pronunciations to label clusters

The relation between polyphone characters and their pronunciations is *N*:*N*. So it is important to map 1305 pronunciations to 10 shared labels. We take ID 0 for non-polyphone characters, whose pronunciation can be determined by looking up dictionary. Meanwhile, keeping balanced data number in each shared label will benefit to training speed and performance of the model. We assume that the label IDs subject to random distribution.

The algorithm is as follow.

Algorithm 1 Assign Label ID for pronunciation

```
1:  for char in poly_chars do
2:    for pron in char's prons do
3:      rand_id = randint(1,9)
4:      for char in homophone chars with this pron do
5:        for c_pron in char's prons do
6:          if c_pron.label_id == rand_id then
7:            goto 3
8:          else
9:            continue
10:         end if
11:         return rand_id
12:       end for
13:     end for
14:   end for
15: end for
```

Firstly, traverse each character's pronunciations (line 1,2), and assign a label randomly (line 3). If the label is the same with that of other pronunciations of current character and related homophone characters (line 4-6), reassign it randomly (line 7). Repeat this process until all pronunciations are assigned to a certain label. Because of randomness of label assignment, the labels distribution keeps balance.

4 Experiments

In this section, we compare our proposal shared labels model with standard Bi-LSTM model. Besides, we also test shared labels model combining with other methods, such as low precision float, projection layer. We compare their performance, memory usage, and speed respectively.

4.1 Data

(Cai et al. , 2019) did their experiments with a public polyphonic character dataset, but it was unavailable when we tried to use it. So we use our own data as train and test sets. We have 188k sentences labeled with their pronunciations. We randomly select 1000 sentences as test set, and others as train set. There are 1127 polyphone characters in our corpus consisting of 1305 pronunciations (labels). Other Chinese characters are non-polyphone, which are labeled as "O", and their pronunciations are got by looking up dictionary.

4.2 Experimental settings

Our experiments are done in tensorflow-GPU[0] version (train) and tensorflow-CPU version (test). Our CPU is Intel Xeon E5, and it does not support AVX512.

- Baseline Bi-LSTM model:The structure of the network is the same as in section 3.2. In the training phase, we set the batch size to 16, learning rate to 0.1, and the dropout rate to 0.2. We adopt gradient descent optimization to learn the parameters.
- Shared labels model: we have the same setting with baseline model, except the output size is 10.
- Bi-LSTM with projection Layer

For further compression, we add a projection layer between input layer and Bi-LSTM layer. Projection layer can factorize the big matrix into small matrixes, which can save memory and reduce the number of parameters.

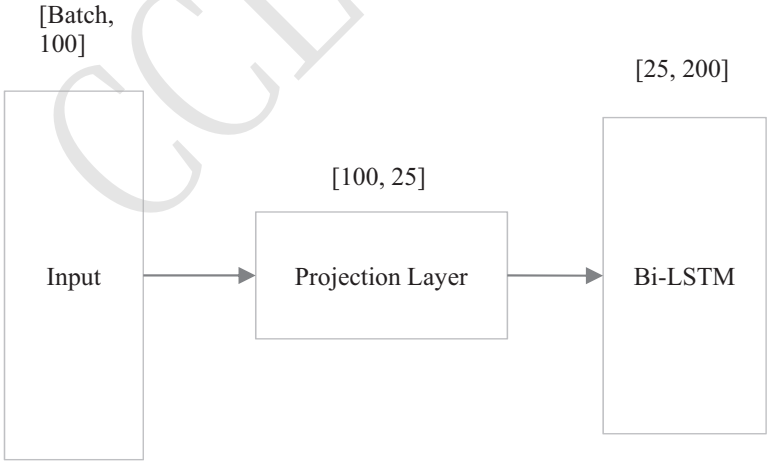


Figure 4: Bi-LSTM with Projection Layer

- Knowledge distillation

Transfer learning is a promising method to improve the performance of compressed and simplified networks. We take 16-bit model with projection layer as student model and baseline as teacher model. We adopt the fusion of soft target and hard target as learning object.

[0]https://github.com/tensorflow/tensorflow

- Shared labels in CRF

CRF is a statistical-based machine learning algorithm, which is popular used in sequence labeling problems. We use it to implement the polyphone selection with shared labels to check if it is workable.

4.3 Experiment results

- Model size and performance

We compare the models by F1-score and file size.

Model	F1-score	Model Size(Kb)
Baseline(Bi-LSTM)	96.86	3925
+ Shared	96.78	1897
+ 16bit	96.85	1963
+ 16bit + Shared	96.75	948
+ 16bit + PL	94.40	1733
+ 16bit + PL + KD	94.96	1733
+ 16bit + Shared + PL	94.55	719

Table 1: F1-score and model size for different models

Note: Shared means shared labels model, PL means Bi-LSTM model with projection layer, KD means knowledge distillation.

From the table, we can see shared labels model is compressed to 48% of baseline model with no obvious performance loss. Combined with 16-bit float and shared label, the size of baseline model is further compressed to 24%, and F1-score only drops 0.11 point.

Compared with projection and knowledge distillation, shared model shows good result in model size reduction and keeps high performance at the same time.

a) By adding projection layer, 16-bit model is compressed a little, but the perfor-mance dropped a lot.

b) Knowledge distillation is useful to improve performance of projection model.

c) Compared with knowledge distillation, shared labels model has a much smaller size and comparable performance.

d) In general, shared labels outperforms projection in both model size and perfor-mance.

- Memory usage

We test the memory usage with open source tool Valgrind[1]. From Figure-5, we can see that the memory usage of shared label model drops a lot compared with that of baseline model.

[1]https://sourceware.org/git/?p=valgrind.git;a=summary

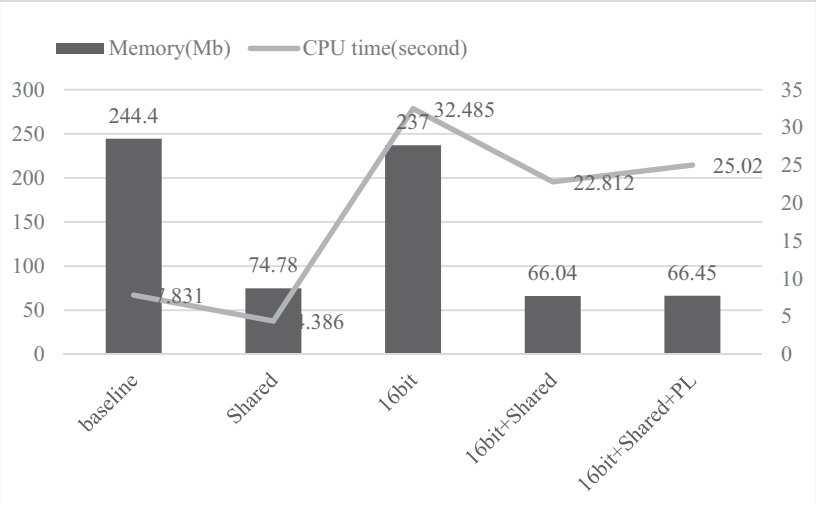


Figure 5: Memory usage and CPU time of different models

- Decoding time

It takes less time for shared labels model compared with its counterpart model, with accelerating by 44% in Shared vs. baseline and 30% in 16bit+Shared vs. 16bit model. But for 16-bit model, the decoding is much slower than baseline, because our experiment machine does not support 16-bit float instructions.

- Result on shared labels in CRF

For CRF with 1305 labels, the feature number is about 900 million which makes it unable to be loaded into memory as much as 256 GB. So the training cannot be continued. If we use 10 shared labels for CRF, the feature number is about 6,900,000, with an exponential reduction compared with previous model. The training speed is fast and its F1-score is as high as 0.9765.

Model	**F1-score**	**Model Size(Kb)**
CRF	NA (unable to train)	
Shared-CRF	97.65	50292

Table 2: Result on CRF for polyphone pronunciation selection

5 Conclusion

We propose a novel shared labels method to compress polyphone pronunciation selection model. It decreases size of models consisting of huge number of labels by mapping labels to small shared labels. Our proposed method reduces the model size and memory usage remarkably, and accelerate decoding speed without performance loss compared with other methods.

In the future, we will verify the compressed model on embedded device and investigate other tasks which can apply this method.

References

Artem M. Grachev, Dmitry I. Ignatov, Andrey V. Savchenko. 2017. *Neural Networks Compression for Language Modeling.* Pattern Recognition and Machine Intelligence, volume 10597.

Geoffrey Hinton, Jeff Dean, Oriol Vinyals. 2015. *Distilling the Knowledge in a Neural Network.* In NIPS Deep Learning and Representation Learning Workshop.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, Chris Dyer. 2016. *Neural Architectures for Named Entity Recognition.* Proceedings of NAACL-HLT, pages 503–512.

John Lafferty, Andrew McCallum, and Fernando CN Pereira 2001. *Conditional random fields: Probabilistic models for segmenting and labeling sequence data.* In Proc. ICML., 28(1):114–133.

M. Courbariaux and Y. Bengio. 2016. *Binarynet: Training deep neural networks with weights and activations constrained to +1 or -1.* CoRR, vol. abs/1602.02830.

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo,Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis,Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow,Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia,Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster,Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens,ABABenoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker,Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas,Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke,Yuan Yu, and Xiaoqiang Zheng. 2015. *TensorFlow: Large-scale machine learning on heterogeneous systems.* Software available from tensorflow.org.

Sepp Hochreiter and Jurgen Schmidhuber 1997. *Long short-term memory.* Neural Computation, 9(8):1735–1780.

S. Gupta, A. Agrawal, K. Gopalakrishnan, and P. Narayanan. 2015. *Deep learning with limited numerical precision.* In Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37, ser. ICML'15, pp. 1737–1746.

S. Han, J. Pool, J. Tran, and W. J. Dally. 2015. *Learning both weights and connections for efficient neural networks.* In Proceedings of the 28th International Conference on Neural Information Processing Systems, ser. NIPS'15.

W. Chen, J. Wilson, S. Tyree, K. Q. Weinberger, and Y. Chen. 2015. *Compressing neural networks with the hashing trick.* JMLR Workshop and Conference Proceedings.

Yu Cheng, Duo Wang, Pan Zhou and Tao Zhang. 2019. *A Survey of Model Compression and Acceleration for Deep Neural Networks.* IEEE Signal Processing Magazine, Special Issue on Deep Learning for Image Understanding.

Zexi Cai, Yaogen Yang, Chuxiong Zhang, Xiaoyi Qin, Ming Li. 2019. *Polyphone Disambiguation for Mandarin Chinese Using Conditional Neural Network with Multi-level Embedding Features.* INTERSPEECH.

Multi-task Legal Judgement Prediction Combining a Subtask of the Seriousness of Charges

Zhuopeng Xu, Xia Li*, Yinlin Li, Zihan Wang, Yujie Fanxu and Xiaoyan Lai
Guangzhou Key Laboratory of Multilingual Intelligent Processing
School of Information Science and Technology
Guangdong University of Foreign Studies,Guangzhou,China
zhuopengxu@126.com,ksyzformy@126.com,zihanwang0703@126.com,
yujiefanxu@126.com,avaxiaoyan@126.com
xiali@gdufs.edu.cn

Abstract

Legal Judgement Prediction has attracted more and more attention in recent years. One of the challenges is how to design a model with better interpretable prediction results. Previous studies have proposed different interpretable models based on the generation of court views and the extraction of charge keywords. Different from previous work, we propose a multi-task legal judgement prediction model which combines a subtask of the seriousness of charges. By introducing this subtask, our model can capture the attention weights of different terms of penalty corresponding to the charges and give more attention to the correct terms of penalty in the fact descriptions. Meanwhile, our model also incorporates the position of defendant making it capable of giving attention to the contextual information of the defendant. We carry several experiments on the public CAIL2018 dataset. Experimental results show that our model achieves better or comparable performance on three subtasks compared with the baseline models. Moreover, we also analyze the interpretable contribution of our model.

1 Introduction

Legal Judgement Prediction (LJP) aims to predict charge, law article and terms of penalty automatically based on the fact descriptions of the criminal cases. It can be used to help the court's judgement and provide legal guidance and assistance to the public.

In recent years, different methods have been proposed to improve the performance of legal judgement prediction task. Some previous studies need to design features manually (Katz et al., 2014; Lin et al., 2012; Liu and Hsieh, 2006; Liu et al., 2015) and some of neural network based models extract features automatically and achieve significant improvements (Liu et al., 2019; Ye et al., 2018; Zhong et al., 2018). However, there are still some challenging problems, including the improvement of the performance and the enhancement of the interpretability of the terms of penalty prediction.

For the improvement of the performance in terms of penalty prediction, previous studies use multi-task and joint learning to obtain the sharing information among different subtasks. Zhong et al. (2018) propose a Directed Acyclic Graph structure with topological relations to capture the information attribution among three subtasks, which effectively improve the problem of insufficient fine-grained in LJP. For the enhancement of the interpretability, different solutions are proposed to the problem. Ye et al. (2018) propose a Seq2Seq model to formulate legal judgement prediction task as a natural language generation problem. Their model take fact descriptions and charge labels as input and outputs the court's view. The outputs are used as an auxiliary information for practical judgement. Liu et al. (2019) propose a multi-task learning model to incorporate charge keywords extracted by TF-IDF and TextRank. Their model has a good interpretability by introduced the keyword information.

Although different methods are proposed for the above two problems, we argue that some of the knowledge are known in legal judgement prediction task and can be incorporated into the model for improving the performance and the interpretability of the prediction results.

*Corresponding author: xiali@gdufs.edu.cn

Figure 1: Attentions of different terms of penalty for charge of "murder" generated by the proposed subtask. As can be seen, term of death or life imprisonment and terms of 7-10 years are paid more attention for serious murder and less serious murder respectively.

The first one is the seriousness of charges. Actual judgement procedure tells us that the final decision of the terms of penalty is largely determined by the seriousness of the case, which depends on the case fact descriptions and the terms of penalty definition described in the article corresponding to the charge of the case. Inspired by the actual judgement procedure, we propose to design a subtask of the seriousness of charges which is determined by the charge and the terms of penalty for the task of legal judgement prediction. According to the scope of legal terms of penalty, we can easily divide a fact description into two categories: serious and less serious. Detailed descriptions and examples are given in Section 3.3. The new subtask is used to obtain attentions of different terms of penalty according to serious and less serious predicted by the subtask and let the model pay more attention to those important terms of penalty for the corresponding fact descriptions. As an example with predicted charge as "murder" which is shown in Fig.1., we can see that our model captures more attention on "Death or life Imprisonment" with predicted serious label and on "7-10 years" with predicted less serious label, which is useful for the model selecting the right terms of penalty.

The second one is the defendant information which is known in the case fact descriptions. Previous studies focuses on the fact descriptions only (eg., just using text words), ignoring the importance of the context information of the defendant. To this end, we propose to incorporate the position of the defendant into the model. By introducing the defendant position-aware embedding for the fact descriptions, we can capture more context information of the defendant which is helpful for the prediction of subtasks. The main contributions of our work are as follows:

1)We propose a multi-task legal judgement prediction model combining a subtask of the seriousness of charges. By introducing this subtask, our model improves the performance and the interpretability of the terms of penalty prediction in LJP.

2)Based on the importance of defendant in the fact descriptions, we propose to incorporate the position information of the defendant into the model, making it capable of giving attention to the relevant context information of the defendant.

3)We carry several experiments on the CAIL2018 dataset. We will show that our proposed model achieves a better or comparable performance in all subtasks than the baseline models. We also give a discussion of our model's interpretability in terms of penalty prediction.

2 Related work

Legal judgement prediction task usually includes three subtasks: charges prediction, law articles recommendation and terms of penalty prediction. We will review the work of legal judgement prediction from single-task based models and multi-task based models.

2.1 Single-task based Legal Judgement Prediction Models.

In the models of single-task based legal judgement prediction, the core perspective is to use different encoding method to represent the fact descriptions more correctly. Luo et al. (2017) propose an attention-based neural network with two hierarchical encoding structures to jointly model the fact descriptions and the top k relevant law articles. Their model achieves good performance for those simple cases, which indicates that the hierarchical encoding structure and introducing of law articles effectively improve the result of charge prediction. Hu et al. (2018) propose an attribute-attentive charge prediction model. They incorporate the fact descriptions attributed by attention mechanism with the original text. Their model performs well in few-shot charges and confusing charge pairs. Ye et al. (2018) propose a label-

conditioned Seq2Seq model with attention mechanism. The model take the fact descriptions and charge labels as input and formulates legal judgement prediction as a natural language generation problem. Their model can automatically generate court views and give a better interpretability of the prediction. In order to improve the terms of penalty prediction, Chen et al. (2019) regard term prediction as a kind of regression problem. By introducing charge labels and using a structure of Deep Gating Network (DGN), their model achieves good results for the terms of penalty prediction.

2.2 Multi-task based Legal Judgement Prediction Models.

Most of above models are proposed for single task such as charge prediction or terms of penalty prediction. However, judge's actual judgement procedure tells us that different subtasks are often related with each other, like charge is related with law and charge is also related to the terms of penalty. To this end, different multitask based learning models are proposed to obtain the relationship information of different subtasks. Zhong et al. (2018) propose a topological multitask learning framework for three subtasks of law articles, charges, and the terms of penalty. They formalized the dependencies among these subtasks as a Directed Acyclic Graph for neural network learning. Their model improves the problem of insufficient fine-grained of legal judgement prediction task. Yang et al. (2019) propose a multi-perspective bi-feedback network with the word collocation attention mechanism. Liu et al. (2019) propose a multi-task learning framework for legal judgement prediction. They use charge keywords extracted by TF-IDF and Text Rank as auxiliary information and use a hierarchical structure to decode the fact descriptions. Their model shows good interpretability because of the introduced charge keywords. Wang et al. (2019) propose a hybrid attention model which combines the improved hierarchical attention network (iHAN) and the deep pyramid convolutional neural network (DPCNN) by ResNet. Their model achieves a good performance for the subtask of the terms of penalty. Xu et al. (2020) take advantages of a novel graph neural network to distinguish confusing law articles and improve the capacity of the encoding of the fact descriptions. Zhong et al. (2020) propose a model based on reinforcement learning, which can visualize the prediction process and give interpretable judgements by giving a process of QA judgement. Their model greatly improves the interpretability of legal judgement prediction task.

This paper focuses on multi-task legal judgement prediction. Different from previous studies, our work focuses on the scope of legal terms of penalty of the different seriousness in the law. We introduce the seriousness of charges as a subtask into the model. By introducing this subtask, it is expected that the prediction of the terms of penalty can obtain improvements not only on the performance but also on the interpretability of the prediction. In addition, in order to make a better judgement to the defendant, our model also combines the defendant's position information in the model.

3 Proposed Model

3.1 Architecture of Our Model

In this paper, we propose a multi-task legal judgement prediction model combining a subtask of the seriousness of charges, which consists of two parts. The first part is encoding layer, in which a defendant position-aware context information is incorporated into the fact descriptions representation. The second part is decoding layer, in which a subtask of the seriousness of charges is introduced to help obtain the attention of different terms of penalty corresponding to the predicted charge. Our model is shown in Fig.2. In the following sections, we will introduce embedding of the defendant's position information in section 3.2 and describe our design of subtask of the seriousness of charges in section 3.3. Section 3.4 will describe model training and prediction.

3.2 Embedding of the Defendant's Position Information

3.2.1 Design of the Defendant's Position Information.

In order to obtain the context information of defendant in the fact descriptions, we use the relative position of each word to the defendant as an indicator to represent the context information of the defendant

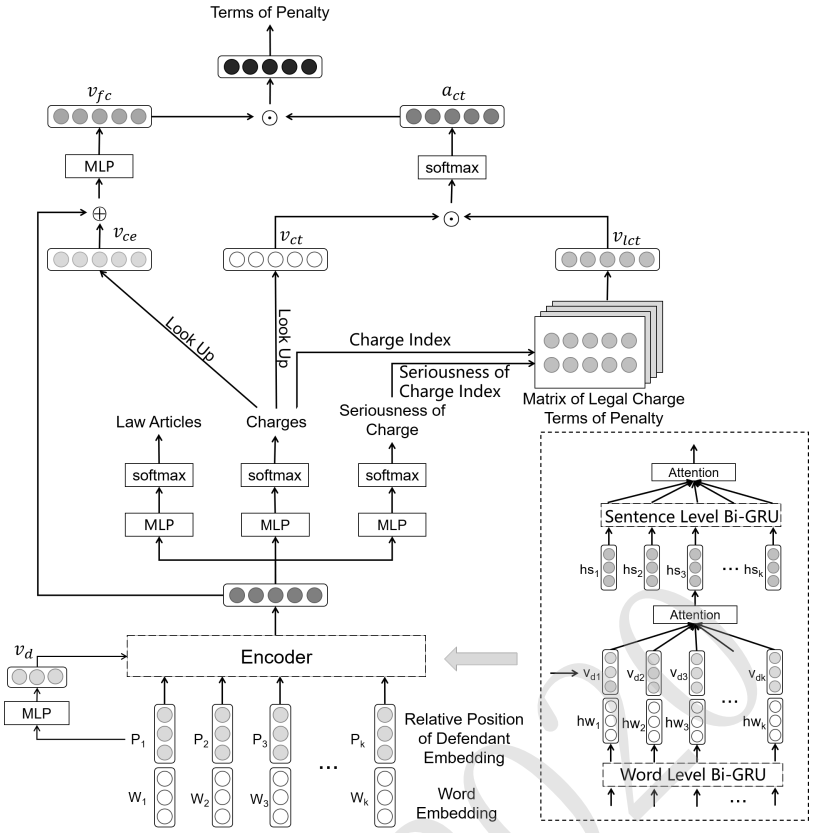


Figure 2: The whole architecture of our model.

He Mou	hold	chopper	run after and cut	the hurt	Zheng MouMou	the defendant	He MouMou	in charge of	drive	pick up
贺某	持	砍刀	追砍	被害人	郑某某	被告人	贺某某	负责	驾车	接应
7	6	5	4	3	2	1	0	1	2	3

Figure 3: An example of relative position of each word to the defendant in a sentence.

in a sentence. For example, as show in Fig.3., the defendant is "贺某某(He MouMou)" whose position is set to 0, the word "驾车(drive)" whose position is 2 and the word "追砍(run after and cut)" whose position is 4. We can see that the action "驾车(drive)" is more relative to the defendant than that of "追砍(run after and cut)". By incorporating the position, the model can learn to focus more on the action "驾车(drive)" than the action "追砍(run after and cut)". This kind of defendant position-aware fact descriptions representation has a better expression of the context information of defendant.

3.2.2 Defendant's Position-aware Fact Descriptions Encoding.

For a given fact descriptions, we formulate it as $d = \{s_1, ..., s_n\}$, in which $s_i \in \mathbb{R}^{L_w \times m}$ is the representation of vectorization of i-th sentence, m is the dimension of the word vector, L_w is the maximum length of a sentence. For sentence s_i, it is formulated as $s_i = \{w_{i1}, \ldots, w_{ik}\}$ represented by k words, in which $w_{ik} \in \mathbb{R}^m$ is the representation of word embedding vector. The relative position of defendant in document d is formulated as $p = \{sp_1, \ldots, sp_n\}$, in which $sp_i \in \mathbb{R}^{L_s \times n}$ is the representation of vectorization of i-th relative position of defendant of sentence formulated as $sp_i = \{wp_{i1}, \ldots, wp_{ik}\}$, $wp_{ik} \in \mathbb{R}^n$ is the representation of vectorization of k-th position in i-th sentence, n is the dimension of the vector of relative position of defendant.

As shown in Fig.2., we employ a structure of hierarchical attention network (Yang et al., 2016) to encode the fact descriptions. Firstly, we encode each word in each sentence on word level by employing Bi-GRU network with attention. We then obtain the hidden representation of each sentence. Secondly, we encode each sentence of a document on sentence level by employing Bi-GRU network with attention,

and then obtain the hidden representation of the document.

For word level encoding, the new word representation is obtained by concatenating the word embedding vector and the relative position of defendant vector. We formulate sentence consist of the new word representation as $s = \{x_1, \ldots, x_k\}$, in which x_k is obtained by w_k and wp_k formulated as $x_k = [w_k; wp_k]$. The w_k and wp_k represent the representation of k-th word in sentence s and the representation of the relative position of defendant of k-th word respectively. Then, we input the representation of sentence s into a word level Bi-GRU network, and then obtain the hidden output of sentence s formulated as $hw = \{hw_1, hw_2, \ldots, hw_k\}$. At t time stamp, we concatenate the hidden output of the forward and backward GRU unit formulated as $hw_t = [\overrightarrow{h_t}, \overleftarrow{h_t}]$.

3.2.3 Defendant's Position-aware Attention Enhancing.

We combine the relative position of defendant vector into word level attention so that the hidden output of each GRU unit in sentence s can better capture the information of position of defendant. Firstly, we employ a multilayer perceptron to obtain the vector v_{dj} which represent the information of position of defendant in each unit in sentence s. Then, we concatenate the hw and v_{dj}. Employing a one-layer MLP, we obtain the new hidden output u_h. Finally, we obtain the hidden representation H_s of sentence s after obtaining the attention aw_t of the new hidden output u_h via softmax function. The W_w and b_w are the parameter of hidden layer projection, u_w is word level context vector. The calculation formula is shown in equations $(1) \sim (4)$.

$$v_{dj} = MLP(sp_j) \tag{1}$$

$$u_h = tanh(W_w[hw, v_{dj}] + b_w) \tag{2}$$

$$aw = softmax(u_h^T u_w) \tag{3}$$

$$H_s = \sum_t aw_t hw_t \tag{4}$$

For sentence level encoding, we input each representation H_s of sentences into a sentence level Bi-GRU network with attention, and then obtain the final hidden representation v_f of fact descriptions.

3.3 Design of the Subtask of the Seriousness of Charges

Based on the definition of terms of penalty, we divide each charge into two categories: serious and less serious. Then we annotate each charge with two legal terms of penalty vectors, which have the same dimension with the prediction of terms of penalty subtask. We also annotate all the samples with the seriousness of chargs, then we can carry a new subtask of the seriousness of charges in the model.

3.3.1 Tagging Rules.

First of all, we manually annotate the legal terms of penalty vectors of the two categories with serious and less serious. The tagging rules are as follows: when the legal terms of penalty is less serious, according to the actual terms of penalty described in law articles, we set the vector of less serious category of the corresponding charge. If there is no distinction between the seriousness of the charge of legal terms of penalty, the vectors of the corresponding serious and less serious legal terms of penalty are set as the same. When a charge includes several seriousness such as less serious, serious, very serious, etc, we combine the serious and more serious parts as the serious category.

Then, we annotate each sample with the label of seriousness. Given a sample, we can determine its corresponding range of legal terms of penalty based on the charge label, if the corresponding range is serious, the seriousness label of the sample is annotated 'serious'; if the corresponding range is less serious, then the seriousness label is annotated 'less serious'. A special case is that if the terms of penalty label is not within the scope of the serious and less serious, we will still annotate it as 'serious'.

3.3.2 Example Demonstration.

In order to better illustrate our annotation rules, we give an example of tagging for the legal terms of penalty vector tagging of a specific charge. As shown in Fig.4., take "故意伤害罪(intentional assault)"

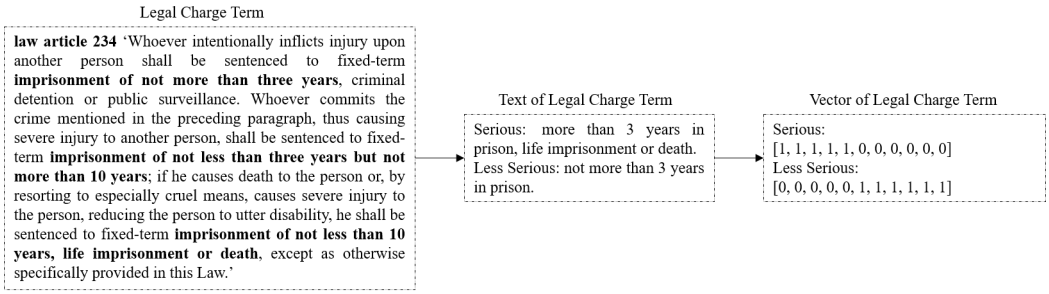


Figure 4: An example of legal charge terms of penalty.

as an example, according to the definition of the corresponding law article 234, we firstly divide the legal terms of penalty into the following three categories: 'less serious': fixed-term imprisonment of not more than three years, criminal detention or public surveillance; 'serious': fixed-term imprisonment of not less than three years but not more than 10 years; 'very serious': fixed-term imprisonment of not less than 10 years, life imprisonment or death. Based on our classification of seriousness, we combine the corresponding legal terms of penalty range of "serious" and "very serious", and the final "serious" legal terms of penalty text is: "fixed-term imprisonment of not less than 3 years, life imprisonment or death"; the "less serious" legal terms of penalty text is "fixed-term imprisonment of not more than three years". Then according to the 11 categories of the subtask of terms of penalty prediction, the corresponding legal terms of penalty range vectors are generated, the serious category vector of the legal terms of penalty of 'intentional assault' is [1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0], and the less serious category vector is [0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1].

3.4 Model Training and Prediction

As shown in section 3.2, after getting the final hidden representation of the fact descriptions v_f ,we employ three different multilayer perceptrons to obtain the decoding vector of law articles, charges, seriousness of charge respectively. Then, as shown in Equation (5), we input them into softmax function to get prediction results $\hat{y}_1$, $\hat{y}_2$ and $\hat{y}_3$ of three subtasks. As shown in Equations (6) and (7), the index vector of the corresponding charges and seriousness of charge was obtained by the prediction results of charges and seriousness of charge respectively.

$$\hat{y}_k = softmax(MLP_k(v_f)), k = 1, 2, 3 \tag{5}$$

$$i_{charge} = argmax(\hat{y}_2) \tag{6}$$

$$i_{seriousness} = argmax(\hat{y}_3) \tag{7}$$

According to the index of the prediction results of charges and seriousness of charge, we obtain legal charge term vector v_{lct}. Similar to word embedding, we obtain charge term vector v_{ct} and charge embedding vector v_{ce} in weight matrices W_{ct} and W_{ce} respectively which have different dimensions and perform joint learning in the model. Then, as shown in Equation (8), we calculate charge term attention weight a_{ct} via charge term vector v_{ct} and legal charge term vector v_{lct}.

$$a_{ct} = softmax(v_{ct} \odot v_{lct}) \tag{8}$$

After concatenating the final hidden representation v_f and charge embedding vector v_{ce}, we input it into a multilayer perceptron. Then, we obtain the vector v_{fc} which is the fusion of fact descriptions and charges, as shown in Equation (9).

$$v_{fc} = MLP([v_f, v_{ce}]) \tag{9}$$

Finally, as shown in Equation (10), we do a hadamard product of v_{fc} and a_{ct} and obtain the final decoding vector v_t of terms of penalty. Then, as shown in Equation (11), we input the vector v_t into

softmax function to get prediction results $\hat{y}_4$.

$$v_t = v_{fc} \odot a_{ct} \tag{10}$$

$$\hat{y}_4 = softmax(v_t) \tag{11}$$

In the training process, we use cross-entropy loss function as the loss function of our model. After calculating each cross-entropy loss for each subtask, we sum each loss of different subtasks as the total loss. As shown in Equation (12), i represents the i-th subtask, Y_i represents the total number of classes of the i-th subtask, and j represents the j-th class.

$$loss_{total} = -\sum_{i=1}^{4}\sum_{j=1}^{Y_i} y_{i,j} log(\hat{y}_{i,j}) \tag{12}$$

4 Experiments

4.1 Dataset

We use the CAIL2018[1] (Xiao et al., 2018) dataset to be evaluated in this paper. Similar to the work of Zhong et al. (Zhong et al., 2018), we do some relevant preprocess on the datasets. Firstly, we filter out the crime data that contained multiple charges and multiple relevant law articles. Secondly, we remove the crime data with charges appeared less than 100 times in the datasets. Finally, similar to the work of TOPJUDGE (Zhong et al., 2018), we divide the terms of penalty into 11 non-overlapping intervals. The detailed information of the CAIL2018 are shown in Table 1.

Dataset	Amount	Subtasks	Amount
Training Set	101513	Charges	119
Testing Set	26731	Law Articles	103
Validation Set	10818	Terms of penalty	11

Table 1: Statistical information of the CAIL2018 dataset.

4.2 Compared Models

In order to compare on three subtasks, we built a multi-task implementation on those not designed for multi-task baseline models. We use Bi-LSTM, TextCNN (Kim, 2014) and Hierarchical Attention Networks (HAN) (Yang et al., 2016) as three different structures to encode the fact descriptions. For HAN structure, we employ a word level of Bi-GRU network with attention and a sentence level of Bi-GRU network with attention to encode the fact descriptions. We employ three different multilayer perceptrons for multitask prediction for these three baselines. We use TOPJUDGE (Zhong et al., 2018) and Few-Shot (Hu et al., 2018) as our another compared models based on their multi-task joint learning and additional auxiliary information design. For the Few-Shot model, we also employ three different multilayer perceptrons for multitask prediction.

4.3 Experimental Setting

In our experiment, we use THULAC (Li and Sun, 2009) for word segmentation. We use skip-gram (Mikolov et al., 2013) for pre-training of all fact descriptions and get a pre-trained 200-dimensional matrix of word vectors. For the position of defendant, we embed each position into a 100-dimensional vector and perform joint training in the model. For the CNN-based and Bi-LSTM-based models in the baselines, we set the maximum document length to 512 words. For the HAN-based models, the maximum sentence length is set to 100 words, the maximum document length is set to 15 sentences. The unit dimension of hidden layer is set to 256, and the output dimension of each level vector is set to 256.

[1]https://github.com/china-ai-law-challenge/CAIL2018

In our model, we embed the maximum sentence length of the relative position of defendant with a blank vector. For training, we use the Adam optimizer to control stochastic gradient descent. The learning rate of the optimizer set to 0.001, the batch size set to 128, and the epoch set to 16. We select the model that performed best on the validation set, and report the results on the testing set.

4.4 Experimental Results

Similar to previous work, we use accuracy (Acc.), macro-precision (MP), macro-recall (MR) and macro-F1 (F1) as metrics in this paper, the final experimental results are shown in Table 2. As LJP task is a multi-label classification task, and there is an extremely unbalanced phenomenon among various categories in the CAIL2018 dataset, we mainly focus on the comparison of the results of macro-F1.

Tasks	Law Articles				Charges				Terms of penalty			
Metrics	Acc.	MP	MR	F1	Acc.	MP	MR	F1	Acc.	MP	MR	F1
Bi-LSTM	79.33	76.45	77.11	75.20	81.69	81.26	**81.77**	80.38	39.66	31.99	29.34	28.26
Text CNN	76.77	74.21	73.13	71.43	82.38	81.20	78.16	78.32	37.85	32.49	27.78	27.79
HAN	81.08	76.85	77.48	76.05	81.97	80.89	81.90	80.37	41.07	31.25	30.71	28.40
TOPJUDGE	**82.11**	76.14	75.82	75.01	82.40	79.48	79.21	78.29	40.04	32.74	30.45	29.59
Few-Shot	79.59	75.62	74.97	73.97	83.33	82.22	80.42	80.56	40.33	30.88	**33.38**	**30.65**
Our model	81.04	**78.43**	**77.27**	**76.49**	**84.47**	**82.42**	81.46	**81.14**	**41.96**	**34.89**	31.11	30.45

Table 2: Experimental results of our model and baselines.

Firstly, we compare our model with Bi-LSTM, TextCNN and HAN models. As shown in Table 2, we can see our model achieves the best macro-F1 value in all three subtasks. And it shows that our model performs great results especially in the subtask of terms of penalty. Our model is 30.45% which is 2.19% higher than Bi-LSTM, 2.66% higher than TextCNN, 2.05% higher than HAN. The results prove the effectiveness of the subtask of seriousness of charge introduced in our model.

Secondly, we compare our model with TOPJUDGE model which is also a multi-task LJP model. As shown in Table 2, our model also achieves better performance in all three subtasks. Our model increases by 1.48% on law articles prediction subtask, 2.85% on charges prediction subtask and 0.86% on terms of penalty prediction subtask. This result shows that our model is ascending to a certain extent on three subtasks compared with the TOPJUDGE model.

Finally, we compare our model with the Few-Shot model which also uses an auxiliary information to help improve the performance of a subtask. We can see that our model increases by 2.52% on law articles prediction, 0.58% on charges prediction, and decreases by 0.2% on terms of penalty prediction which is comparable with the Few-Shot model. The results indicate that the overall performance of our model can be improved on the basis of improving term prediction results.

4.5 Ablation Studies

In order to analyze the influence of each part of our model, several ablation experiments are conducted in this paper. We remove four parts from our model to see the influences: 1)We remove the word level attention calculated by the position of defendant which is named as w/o drp_att. 2)We remove the whole part of using position of defendant, named as w/o drp_pos+drp_att, which means that the model only judges with the fact descriptions. 3)We remove the subtask of the seriousness of charges which is named as w/o seriousness to see the influence of the subtask to the whole model. 4)We remove the whole part of relative position of defendant and the subtask of the seriousness of charges, which means that the fact descriptions is only encoded by hierarchical attention networks and predicted by multitask learning, and the model is named as w/o drp_both+seriousness. The results of different parts of ablation studies are shown in Table 3.

As shown in Table 3, when we remove the subtask of seriousness of charge (w/o seriousness), the macro-F1 of the subtask of terms of penalty prediction is reduced by 2.04%, which shows that the introducing of the subtask of seriousness of charge can significantly improve the result of the terms of penalty

Tasks	Law Articles		Charges		Terms of Penalty		Seriousness of Charge	
Metrics	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
Our Model	81.04	76.49	84.47	81.14	41.96	30.45	87.18	80.09
w/o drp_att	81.71	76.3	83.49	81.31	41.63	29.95	86.88	80.05
w/o drp_pos+drp_att	81.68	75.53	83.28	81.02	41.66	29.84	86.87	79.57
w/o seriousness	81.28	76.59	83.87	81.51	41.41	28.41	/	/
w/o drp_both+seriousness	81.08	76.05	81.97	80.37	41.07	28.40	/	/

Table 3: Results of ablation experiments.

prediction. In addition, according to Table 3, we can see that the decoder with the introducing of the subtask of seriousness of charge is the most effective part in all additional components.

We also can see that after embedding the position information of defendant, the prediction results of charges and law articles can be improved. Moreover, compared with embedding the position information of defendant, using position information of defendant to improve word level attention can further improve the performance of the model in three subtasks. When we remove all the position information of defendant, the macro-F1 of the subtask of law articles prediction will decrease by 0.96%. This result shows that the position information of defendant can mainly improve the result of law articles prediction. In the end, when we combine the position information of defendant and the subtask of the seriousness of charges into a model, the performances of all three subtasks are improved.

4.6 Interpretability Analysis

In order to analyze the interpretability of our model, we choose a representative case to illustrate how the design of the subtask of the seriousness of charges can be improved in interpretability of the prediction of terms of penalty.

As shown in Fig.5., given the fact descriptions, previous method will predict and give the terms of penalty directly without any auxiliary information. While in our model, firstly, we will preliminarily predict the prediction results of charges, law articles and the seriousness of charge. With the predicted charge and the seriousness of the charge, our model can determine the range of legal charge terms of penalty, this is important and useful for the judge and the public to get the auxiliary information of the terms of the penalty. Finally, the model outputs the prediction result of the terms of penalty. Compared with previous direct prediction process of terms of penalty, the prediction process of our model has a better interpretability of the prediction.

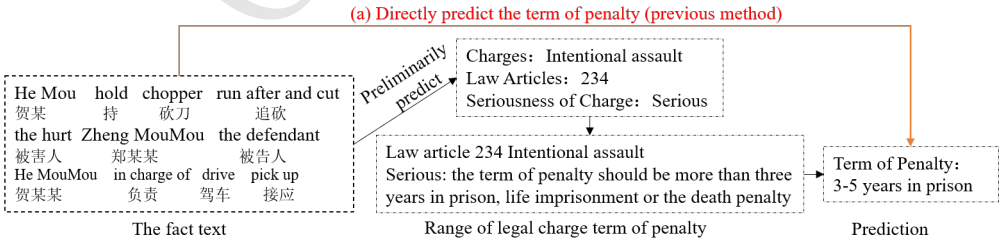


Figure 5: Terms of penalty prediction process of our model compared with previous method.

5 Conclusion

In this paper, we propose to design and combine a subtask of the seriousness of charges for multi-task legal judgement prediction. Evaluations demonstrate the effectiveness of our model on charge prediction, law article recommendation and the terms of penalty prediction, indicating that the introduced subtask of the seriousness of charges and the sufficient encoding of the fact descriptions for the defendant are useful. Our model also shows the good interpretability on the task of terms of penalty prediction. In

the future, we will explore a better method to incorporate the contextual information of defendant and investigate the usefulness of different subtasks for multi-task legal judgement prediction.

Acknowledgements

This work is supported by National Natural Science Foundation of China (No. 61976062), the Science and Technology Program of Guangzhou (No. 201904010303) and Special Funds for the Cultivation of Guangdong College Students' Scientific and Technological Innovation (No. pdjh2020a0197).

References

Huajie Chen, Deng Cai, Wei Dai, Zehui Dai, and Yadong Ding. 2019. Charge-based prison term prediction with deep gating network. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 6361–6366.

Zikun Hu, Xiang Li, Cunchao Tu, Zhiyuan Liu, and Maosong Sun. 2018. Few-shot charge prediction with discriminative legal attributes. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 487–498.

Daniel Martin Katz, Michael J Bommarito Ii, and Josh Blackman. 2014. Predicting the behavior of the supreme court of the united states: A general approach. *Plos One*, 12(4).

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

Zhongguo Li and Maosong Sun. 2009. Punctuation as implicit annotations for chinese word segmentation.

Wan-Chen Lin, Tsung-Ting Kuo, Tung-Jia Chang, Chueh-An Yen, Chao-Ju Chen, and Shou-de Lin. 2012. Exploiting machine learning models for Chinese legal documents labeling, case classification, and sentencing prediction in Chinese. In *Proceedings of the 24th Conference on Computational Linguistics and Speech Processing (ROCLING 2012)*, pages 140–141.

Chao-Lin Liu and Chwen-Dar Hsieh. 2006. Exploring phrase-based classification of judicial documents for criminal charges in chinese. In *Foundations of Intelligent Systems*, pages 681–690.

Yi-Hung Liu, Yen-Liang Chen, and Wu-Liang Ho. 2015. Predicting associated statutes for legal problems. *Inf. Process. Manag.*, 51(1):194–211.

Zonglin Liu, Meishan Zhang, Ranran Zhen, Zuoquan Gong, Nan Yu, and Guohong Fu. 2019. Multi-task learning model for legal judgment predictions with charge keywords. *Journal of Tsinghua University(Science and Technology)*, 59(7):497.

Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyan Zhao. 2017. Learning to predict charges for criminal cases with legal basis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2727–2736.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *Computer Science*.

Wenguan Wang, Yunwen Chen, Hua Cai, Yanneng Zeng, and Huiyu Yang. 2019. Judicial document intellectual processing using hybrid deep neural networks. *Journal of Tsinghua University(Science and Technology)*, 59(7):505.

Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, and Jianfeng Xu. 2018. CAIL2018: A large-scale legal dataset for judgment prediction. *CoRR*, abs/1807.02478.

Nuo Xu, Pinghui Wang, Long Chen, Li Pan, Xiaoyan Wang, and Junzhou Zhao. 2020. Distinguish confusing law articles for legal judgment prediction. *CoRR*, abs/2004.02557.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489.

Wenmian Yang, Weijia Jia, Xiaojie Zhou, and Yutao Luo. 2019. Legal judgment prediction via multi-perspective bi-feedback network. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 4085–4091.

Hai Ye, Xin Jiang, Zhunchen Luo, and Wenhan Chao. 2018. Interpretable charge predictions for criminal cases: Learning to generate court views from fact descriptions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1854–1864.

Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. 2018. Legal judgment prediction via topological learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3540–3549.

Haoxi Zhong, Yuzhong Wang, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. Iteratively questioning and answering for interpretable legal judgment prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:1250–1257.

Clickbait Detection with Style-aware Title Modeling and Co-attention

Chuhan Wu[†], **Fangzhao Wu**[‡], **Tao Qi**[†], **Yongfeng Huang**[†]
[†]Department of Electronic Engineering & BNRist, Tsinghua University, Beijing 100084, China
[‡]Microsoft Research Asia, Beijing 100080, China
{wuchuhan15, wufangzhao, taoqi.qt}@gmail.com
yfhuang@tsinghua.edu.cn

Abstract

Clickbait is a form of web content designed to attract attention and entice users to click on specific hyperlinks. The detection of clickbaits is an important task for online platforms to improve the quality of web content and the satisfaction of users. Clickbait detection is typically formed as a binary classification task based on the title and body of a webpage, and existing methods are mainly based on the content of title and the relevance between title and body. However, these methods ignore the stylistic patterns of titles, which can provide important clues on identifying clickbaits. In addition, they do not consider the interactions between the contexts within title and body, which are very important for measuring their relevance for clickbait detection. In this paper, we propose a clickbait detection approach with style-aware title modeling and co-attention. Specifically, we use Transformers to learn content representations of title and body, and respectively compute two content-based clickbait scores for title and body based on their representations. In addition, we propose to use a character-level Transformer to learn a style-aware title representation by capturing the stylistic patterns of title, and we compute a title stylistic score based on this representation. Besides, we propose to use a co-attention network to model the relatedness between the contexts within title and body, and further enhance their representations by encoding the interaction information. We compute a title-body matching score based on the representations of title and body enhanced by their interactions. The final clickbait score is predicted by a weighted summation of the aforementioned four kinds of scores. Extensive experiments on two benchmark datasets show that our approach can effectively improve the performance of clickbait detection and consistently outperform many baseline methods.

1 Introduction

Clickbait is a type of web content that is designed to attract users' attention and further entice them to click hyperlinks to enter specific webpages, such as news articles, advertisements and videos (Chakraborty et al., 2016). Several illustrative examples of clickbaits are shown in Fig. 1. We can see that the title of the first clickbait is written in a sensationalized way by using words with strong emotions like "MUST", and the title of the second clickbait is misleading because it does not match the content of the body. Clickbaits are commonly used by online publishers, because clickbaits can draw more attention to the online websites where they are displayed and improve the revenue by attracting more clicks on advertisements (Dong et al., 2019). However, clickbaits are deceptive to users because the main content of clickbaits is often uninformative, misleading, or even irrelevant to the title, which is extremely harmful for the reading satisfaction of users (Chen et al., 2015). Thus, clickbait detection is an important task for online platforms to improve the quality of their web content and maintain their brand reputation by improving user experience (Biyani et al., 2016).

Many methods formulate clickbait detection as a binary detection task, and they mainly focus on modeling the content of online articles and the relevance between title and body (Zhou, 2017; Kumar et al., 2018; Dong et al., 2019). For example, Zhou et al. (2017) proposed to use a combination of bi-GRU network and attention network to learn representations of tweets posted by users for clickbait detection. Dong et al. (2019) proposed a similarity-aware clickbait detection model, which learns title and body

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1143-1154, Hainan, China, October 30 - November 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

Title	7 Things You MUST Know About Exercise and Weight Loss	Covid-19 news in your area	You Won't Believe How Many Beloved Mom-and-Pop Restaurants are Closing
Body	The biggest challenge for an obese person is losing a few extra pounds. Well, people sometimes let themselves eat what they like...	Download our app today and get what you want! Consider joining this community as a helpful resource...	The pandemic has caused a lot of businesses to fold, especially independent restaurants, cafes, and coffee shops.

Figure 1: Several illustrative examples of clickbaits.

representations via an attentive bi-GRU network, and measures the global and local similarities between these representations for clickbait prediction. However, in these methods the stylistic patterns of titles (e.g., capitalization) are not taken into consideration, which are useful clues for identifying clickbaits (Biyani et al., 2016). In addition, they cannot model the interactions between the contexts in the title and body, which are important for measuring the title-body relevance for clickbait detection.

Our work is motivated by the following observations. First, the content of webpage title and body is important for clickbait detection. For example, in the title of the third webpage in Fig. 1, the contexts like "You Won't Believe" are important indications of clickbaits because they express strong emotions. In addition, the body of this webpage is short and uninformative, which also implies that this webpage is a clickbait. Second, the stylistic patterns of title like the usage of numeric and capitalized characters can also provide useful clues for identifying clickbaits. For example, the title of the first webpage in Fig. 1 starts with a number "7" and it uses an all-capital word "MUST" to attract attention, both of which are commonly used by clickbaits. Therefore, modeling the stylistic patterns of title can help detect clickbaits more accurately. Third, there is inherent relatedness between the contexts within the title and body of the same webpage. For example, the words "Weight Loss" in the title of the first webpage in Fig. 1 have close relatedness with the words "losing" and "pounds" in the body. Modeling these interactions are helpful for measuring the relevance between title and body more accurately.

In this paper, we propose a clickbait detection approach with **s**tyle-**a**ware **t**itle modeling and **c**o-attention (SATC), which can consider the interactions between contexts within title and body as well as the stylistic patterns of title. We first use Transformers to learn representations of title and body based on their content, and then compute a title content score and a body content score based on the representations of title and body, respectively. In addition, we propose to use a character-level Transformer to learn a style-aware title representation by capturing the stylistic patterns in the title, and we further compute a title stylistic score based on this representation. Besides, we propose to use a co-attention network to model the interactions between the contexts within title and body, and further enhance their representations by encoding their interaction information. We compute a title-body matching score based on the relevance between the interaction-enhanced representations of title and body. The final unified clickbait score is a weighted summation of the four kinds of scores, which jointly considers the content of title and body, the stylistic information of title, and the relevance between title and body. Extensive experiments on two benchmark datasets show that our approach can effectively enhance the performance of clickbait detection by incorporating the stylistic patterns of title and the title-body interactions.

The main contributions of this paper are summarized as follows:

- We propose a style-aware title modeling method to capture the stylistic patterns of title to learn style-aware title representations for clickbait detection.
- We propose to use co-attention network to model the interactions between the contexts within title and body to better evaluate their relevance.
- Extensive experiments are conducted on two benchmark datasets, and the results validate the effectiveness of our approach in clickbait detection.

2 Related Work

Automatic detection of clickbaits is important for online platforms to purify their web content and improve user experience. Traditional clickbait detection methods usually rely on handcrafted features to build representations of webpages (Chen et al., 2015; Biyani et al., 2016; Potthast et al., 2016; Chakraborty et al., 2016; Bourgonje et al., 2017; Cao et al., 2017; Indurthi and Oota, 2017; Geçkil et al., 2018). For example, Chen et al. (2015) proposed to represent news articles with semantic features (e.g., unresolved pronouns, affective words, suspenseful language and overuse numerals), syntax features (e.g., forward reference and reverse narrative) and image features (e.g., image placement and emotional content). In addition, they incorporate users' behaviors on news, like reading time, sharing and commenting, to enhance news representation. They use various classification models like Naive Bayes and SVM to identify clickbaits based on the news and user behavior features. Biyani et al. (2016) proposed to represent webpages using content features like n-gram features extracted from title and body, sentiment polarity features, part-of-speech features and numerals features. They also incorporate the similarities between the TF-IDF features of title and the first 5 sentences in the body. Besides, they consider the informality of title, the use of forward reference, and the URL of webpage as complementary information. They used Gradient Boosted Decision Trees (GBDT) to classify webpages based on their features. Potthast et al. (2016) proposed to detect clickbaits on Twitter. They used features like bag-of-words, image tags, and dictionary matchings to represent tweets, and used bag-of-words, readability and length features to represent the linked webpage. They also incorporated several metadata features like the gender of user. They compared several machine learning models including logistic regression, naive Bayes, and random forests for clickbait classification. However, these methods need heavy feature engineering, which depends on a large amount of domain knowledge. In addition, handcrafted features are usually not optimal in representing the textual content of webpages since they cannot effectively model the contexts of words.

In recent years, several approaches explore to use deep learning techniques for clickbait detection (Agrawal, 2016; Fu et al., 2017; Zhou, 2017; Thomas, 2017; Dimpas et al., 2017; Anand et al., 2017; Kumar et al., 2018; Zheng et al., 2018; Dong et al., 2019). For example, Agrawal et al. (2016) proposed a neural clickbait detection approach, which uses convolutional neural network (CNN) with max pooling techniques to learn representations of titles. Zhou et al. (2017) proposed to use a bi-GRU network to learn contextual word representations, and use an attention network to select important words for learning informative tweet representations for clickbait detection. Kumar et al. (2018) proposed to learn title representations with an attentive bi-GRU network, and used two Siamese networks to respectively measure the relevance between the title and body and the relevance between the associated image and body. They combined the title representation and the relevance vectors for final prediction. Dong et al. (2019) proposed a similarity-aware clickbait detection model. They used a combination of bi-GRU network and attention network to learn title and body representations, and computed a similarity vector based on the global and local vector similarities between the representations of titles and bodies. They combined the title and body representations with the similarity vector for clickbait prediction. However, these methods do not consider the stylistic patterns of titles when learning their representations, which are important cues for clickbait detection. In addition, they do not consider the interactions between the contexts in the title and body, which are usually important for evaluating their relevance. Different from existing methods, our approach incorporates a character-level Transformer to capture the stylistic patterns of title, which can help recognize clickbaits more accurately. In addition, it can model the interactions between title and body via co-attention to enhance their representations.

3 Methodology

In this section, we introduce our proposed clickbait detection approach with style-aware title modeling and co-attention (SATC). The framework of our proposed *SATC* approach is illustrated in Fig. 2. It consists of four core modules, i.e., a *content modeling* module to learn representations of title and body from their content, a *style modeling* module to capture the stylistic patterns in the title, an *interaction modeling* module to capture the interactions between the contexts within title and body, and a *clickbait prediction* module to compute the clickbait score. The details of each module are introduced as follows.

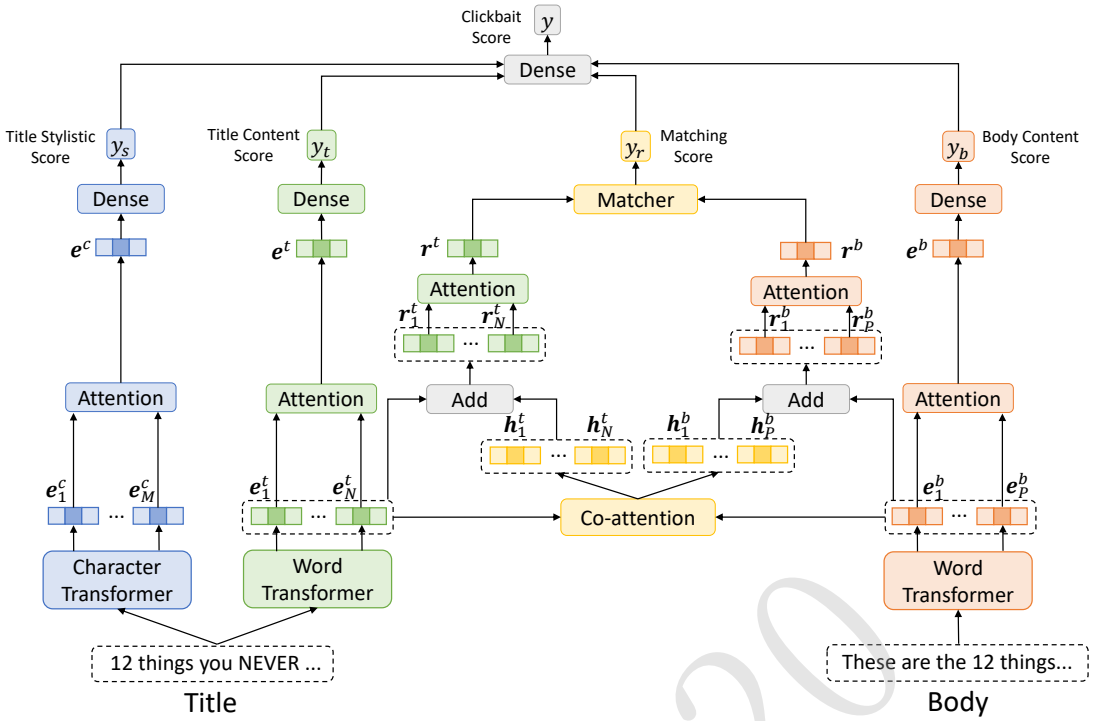


Figure 2: The architecture of our *SATC* approach for clickbait detection.

3.1 Content Modeling

The *content modeling* module is used to learn the representations of title and body from their content. We respectively denote the sequences of words in title and body as $[w_1^t, w_2^t, ..., w_N^t]$ and $[w_1^b, w_2^b, ..., w_P^b]$, where N and P respectively stand for the number of words in the title and body. In this module, we first use a word embedding layer to convert both word sequences into sequences of semantic vectors, which are denoted as $[\mathbf{w}_1^t, \mathbf{w}_2^t, ..., \mathbf{w}_N^t]$ and $[\mathbf{w}_1^b, \mathbf{w}_2^b, ..., \mathbf{w}_P^b]$. Usually the contexts of words in title and body are important for modeling their content. For example, in the title of the first webpage in Fig. 1, the contexts of the word "Loss" such as "Weight" and "Exercise" are useful clues for understanding that this word is about fitness rather than financial loss. Transformer (Vaswani et al., 2017) is an effective neural architecture for context modeling. Thus, we apply two independent Transformers to learn hidden representations of words in title and body by modeling their contexts. We denote the hidden representation sequences of words in title and body as $\mathbf{E}^t = [\mathbf{e}_1^t, \mathbf{e}_2^t, ..., \mathbf{e}_N^t]$ and $\mathbf{E}^b = [\mathbf{e}_1^b, \mathbf{e}_2^b, ..., \mathbf{e}_P^b]$, respectively. Different words in a title or body may have different importance for modeling the content. For instance, the word "MUST" in Fig. 1 is more important than the word "About" in learning title representation for clickbait detection. Thus, we apply attention mechanisms (Yang et al., 2016) to select words in the title and body to form unified representations for them (denoted as $\mathbf{e}^t$ and $\mathbf{e}^b$), which are respectively formulated as follows:

$$\mathbf{e}^t = Attention([\mathbf{e}_1^t, \mathbf{e}_2^t, ..., \mathbf{e}_N^t]), \tag{1}$$

$$\mathbf{e}^b = Attention([\mathbf{e}_1^b, \mathbf{e}_2^b, ..., \mathbf{e}_P^b]). \tag{2}$$

3.2 Style Modeling

The *style modeling* module is used to capture the stylistic patterns in the title to better identify clickbaits. Usually, there are some common patterns on the style of clickbait titles. For example, many clickbaits use all-capital words (e.g., "MUST", "NOT" and "THIS"), exclamation marks, and numeric characters to attract users' attention. Thus, it is very important to grasp these stylistic patterns in clickbait detection. To capture these patterns, we propose to use a character-level Transformer to learn style-aware title

representations from its original characters. We denote the character sequence (including whitespace) of the title as $[c_1, c_2, ..., c_M]$, where M is the number of characters. We first convert these characters into their embeddings (denoted as $[\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_M]$) via a character embedding layer, and then use a character Transformer to learn the hidden representations of these characters, which are denoted as $[\mathbf{e}_1^c, \mathbf{e}_2^c, ..., \mathbf{e}_M^c]$. Usually different characters may have different importance in style modeling. For example, in Fig. 1 the character "7" is more important than the character "a" in the word "and". Thus, we use a character-level attention network for character selection in building the style-aware title representation $\mathbf{e}^c$, which is formulated as follows:

$$\mathbf{e}^c = Attention([\mathbf{e}_1^c, \mathbf{e}_2^c, ..., \mathbf{e}_M^c]). \tag{3}$$

3.3 Interaction Modeling

The *interaction modeling* module is used to capture the interactions between title and body. For most webpages, the contexts in their titles usually have relatedness with the contexts in their bodies to a certain extent. For instance, the words "Restaurants" in the title of the third webpage in Fig. 1 have close relatedness with the words "businesses", "restaurants" and "cafes" in the body. These interactions are important cues for modeling the relevance between title and body, which is critical for clickbait detection. Thus, we propose to use a multi-head co-attention network to capture the interactions between title and body. More specifically, we first use the title word representation sequence $\mathbf{E}^t$ as the query, and use the body word representation sequence $\mathbf{E}^b$ as the key and value to compute a hidden representation sequence $\mathbf{H}^t = [\mathbf{h}_1^t, \mathbf{h}_2^t, ..., \mathbf{h}_N^t]$, which summarizes the contexts within body and their interactions with each word in the title. This process is formulated as follows:

$$\mathbf{H}^t = MultiHead(\mathbf{E}^t, \mathbf{E}^b, \mathbf{E}^b). \tag{4}$$

Next, we use the body word representation sequence $\mathbf{E}^b$ as the query, and use the title word representation sequence $\mathbf{E}^t$ as the key and value to compute an hidden representation sequence $\mathbf{H}^b = [\mathbf{h}_1^b, \mathbf{h}_2^b, ..., \mathbf{h}_P^b]$ that conveys the contexts in title and their interactions with each word in body, which is formulated as follows:

$$\mathbf{H}^b = MultiHead(\mathbf{E}^b, \mathbf{E}^t, \mathbf{E}^t). \tag{5}$$

Then, we use the interactions between title and body to enhance their representations. We add the hidden representation sequence $\mathbf{H}^t$ to the original word representation sequence $\mathbf{E}^t$ to form a unified representation sequence $\mathbf{R}^t$, i.e., $\mathbf{R}^t = \mathbf{E}^t + \mathbf{H}^t$. The unified body word representation sequence $\mathbf{R}^b$ is obtained by $\mathbf{R}^b = \mathbf{E}^b + \mathbf{H}^b$. Similar to the *content modeling* module, we also use attention networks to obtain the final interaction-enhanced representations of title and body (denoted as $\mathbf{r}^t$ and $\mathbf{r}^b$), which are formulated as follows:

$$\mathbf{r}^t = Attention([\mathbf{r}_1^t, \mathbf{r}_2^t, ..., \mathbf{r}_N^t]), \tag{6}$$

$$\mathbf{r}^b = Attention([\mathbf{r}_1^b, \mathbf{r}_2^b, ..., \mathbf{r}_P^b]), \tag{7}$$

where $\mathbf{r}_i^t$ and $\mathbf{r}_i^b$ stand for the i-th vector in $\mathbf{R}^t$ and $\mathbf{R}^b$, respectively.

3.4 Clickbait Prediction

The *clickbait prediction* module is used to compute a clicbait score based on the representations of title and body. We first use a dense layer to compute a title content score y_t based on the content representation $\mathbf{e}^t$ of the title, which is formulated as $y_t = \mathbf{w}_t^\top \mathbf{e}^t + b_t$, where $\mathbf{w}_t$ and b_t are the kernel and bias parameters. We compute a body content score y_b based on $\mathbf{e}^b$ in a similar way, which is formulated as $y_b = \mathbf{w}_b^\top \mathbf{e}^b + b_b$, where $\mathbf{w}_b$ and b_b are parameters. Next, we use a matcher to compute a title-body matching score, which indicates the relevance between title and body. It takes the interaction-enhanced representations of title and body ($\mathbf{r}^t$ and $\mathbf{r}^b$) as the input, and outputs the matching score y_r. Following (Okura et al., 2017), we use dot-product to implement the matcher, and the score y_r is computed as $y_r = \mathbf{r}^t \cdot \mathbf{r}^b$. Then, we use another dense layer to compute a title stylistic score based on the style-aware title representation $\mathbf{e}^c$, which is formulated as $y_s = \mathbf{w}_s^\top \mathbf{e}^s + b_s$, where $\mathbf{w}_s$ and b_s are parameters. The final clickbait score y is a

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1143-1154,
Hainan, China, October 30 - Novermber 1, 2020. (c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

weighted summation of the aforementioned four scores and we use the sigmoid function for normalization, which is formulated as follows:

$$y = sigmoid(\alpha_s y_s + \alpha_t y_t + \alpha_r y_r + \alpha_b y_b), \tag{8}$$

where α_s, α_t, α_r and α_b are trainable parameters.

For model training, we use binary cross-entropy as the loss function. By comparing the predicted clickbait score with the gold label, we can obtain the loss on the training samples, and further compute the gradients for model update.

4 Experiments

4.1 Dataset and Experimental Settings

Our experiments are conducted on two benchmark datasets for clickbait detection. The first one is *Clickbait Challenge*[1], which is a dataset released by the organizers of Clickbait Challenge 2017. This dataset contains the tweet texts posted by users and the content of the corresponding article. Each pair of tweet and article is annotated by 5 judgers, where each judger gives a clickbait score from 0 (non-clickabit) to 1 (clickbait) to this pair. Following (Dong et al., 2019), we regard the pairs with the mean score over 0.5 as clickbaits. The training set contains 19,538 pairs, and the validation set contains 2,495 pairs. Since the labels of the test set are not released, we evaluate the model on the current validation set, and randomly sample 10% of pairs in the training set for validation. The second one is *FNC*[2], which is released by the Fake News Challenge in 2017. In this dataset, each pair of title and body is labeled as "agree", "disagree", "discuss" or "unrelated". Following (Dong et al., 2019), we regard the pairs with "unrelated" labels as clickbaits. This dataset contains 49,972 pairs of titles and bodies for training and 25,413 pairs for test. We also use 10% of training samples for validation.

In our experiments, we use the pre-trained 300-dimensional Glove embeddings (Pennington et al., 2014) to initialize the parameters in the word embedding layer. We do not fine-tune these pre-trained word embeddings in model training to avoid overfitting. The character embeddings are 50-dimensional. The Transformers have two self-attention layers. Each layer has 8 attention heads, and the output dimension of each head is 32. We apply dropout (Srivastava et al., 2014) to the word and character embeddings at a ratio of 20%. We use Adam (Kingma and Ba, 2014) as the optimizer, and the learning rate is 0.01. The size of each mini-batch is 64. These hyperparameters are searched according to the performance on the validation sets. Each experiment is repeated 5 times, and the average results in terms of accuracy, precision, recall and Fscore are reported.

4.2 Performance Evaluation

We compare our *SATC* method with several baseline methods, including:

- DSSM (Huang et al., 2013), deep structured semantic model, where title is regarded as the query and body is regarded as document. The texts of title and body are represented by N-gram featuress.
- CLSM (Shen et al., 2014), a variant of DSSM that uses CNN to learn text representations;
- CNN (Agrawal, 2016; Zheng et al., 2018), which detects clickbaits solely based on titles. Text-CNN is used to learn title representations.
- LSTM (Glenski et al., 2017), using LSTM networks to learn title and body representations for clickbait detection.
- GRU-Att (Zhou, 2017), using a combination of bi-GRU network and attention network to learn title representations for clickbait detection.

[1]https://www.clickbait-challenge.org/.
[2]http://www.fakenewschallenge.org/

Method	Clickbait Challenge				FNC			
	Accuracy	Precision	Recall	Fscore	Accuracy	Precision	Recall	Fscore
DSSM	0.817	0.655	0.661	0.658	0.747	0.894	0.740	0.811
CLSM	0.833	0.683	0.643	0.662	0.756	0.959	0.762	0.853
CNN	0.844	0.654	0.653	0.653	0.789	0.852	0.845	0.857
LSTM	0.827	0.642	0.621	0.631	0.868	0.925	0.884	0.913
GRU-Att	0.856	0.719	0.650	0.683	0.879	0.924	0.897	0.919
Siamese Net	0.844	0.695	0.688	0.691	0.859	0.920	0.877	0.907
LSDA	0.860	0.697	0.699	0.710	0.894	0.933	0.912	0.928
SATC*	**0.889**	**0.745**	**0.722**	**0.733**	**0.907**	**0.959**	**0.917**	**0.938**

Table 1: Performance comparison of different methods on the two datasets. *Improvement is significant at the level of $p < 0.01$.

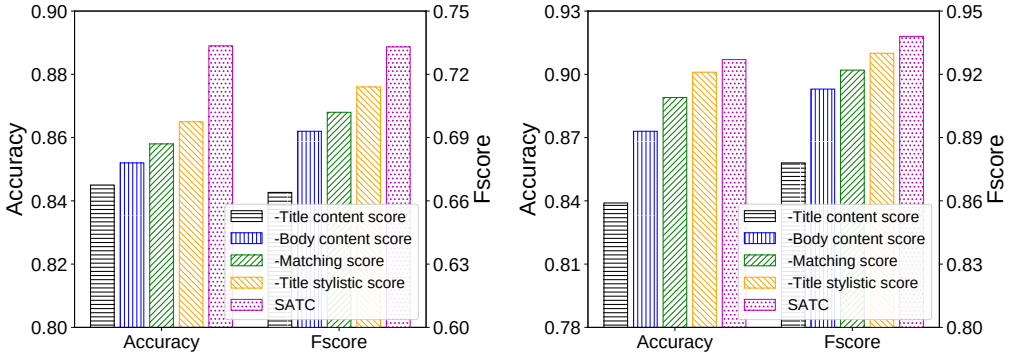
- SiameseNet (Kumar et al., 2018), which uses *GRU-Att* to learn title representations and uses Siamese networks to capture the relevance between title and body.
- LSDA (Dong et al., 2019), which uses *GRU-Att* to learn title and body representations, and measures their relevance using the global and local similarities between the representation vectors of title and body.

The results on the two datasets are summarized in Table 1.[3] According to the results, we have several main findings. First, the methods that use neural networks to learn text representations (e.g., *CNN*, *LSTM*, *GRU-Att* and *SATC*) outperform the *DSSM* method that uses handcrafted features for text representation. It shows that handcrafted features are usually not-optimal in representing the textual content of webpages for clickbait detection. Second, the methods based on attention mechanisms (e.g., *GRU-Att* and *LSDA*) usually outperform the methods without attention (e.g., *CNN* and *LSTM*). This is probably because attention mechanism can select important contexts within title and body to learn more informative representations for them, which is beneficial for clickbait detection. Third, our approach can consistently outperform the compared baseline methods. This is because our approach can capture the stylistic patterns in the title to learn style-aware title representations, and meanwhile can model the interactions between contexts in title and body to help measure their relevance more accurately. In addition, Transformers may also have a greater ability than CNN, LSTM and GRU in context modeling. Thus, our method can detect clickbaits more effectively than baseline methods.

4.3 Influence of Different Scores

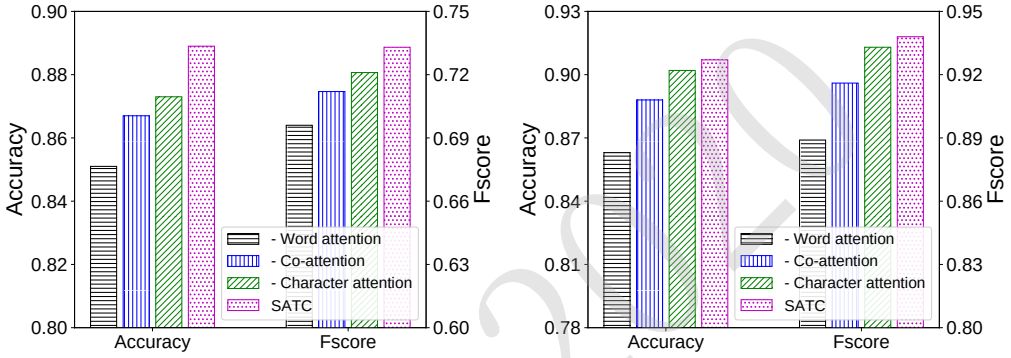
In this section, we conduct several ablation studies to explore the influence of the four clickbait scores. We compare the performance of our *SATC* approach by removing one of these scores in clickbait prediction. The results on the *Clickbait Challenge* and *FNC* datasets are respectively shown in Figs. 3(a) and 3(b). From the results, we find that the title content score plays the most important role. This is intuitive because clickbaits mainly rely on the content of their titles to attract users' attention and clicks. Thus, modeling the title content is critical for clickbait detection. In addition, we find the body content score is also important. This is because the body of many clickbaits may be misleading or uninformative. Thus, modeling the content of body is important for clickbait detection. Besides, the matching score is also useful for clickbait prediction. This is probably because the titles of some clickbaits do not perfectly match their bodies. Thus, modeling the relevance of title and body is useful for accurate clickbait detection. Moreover, we find the title stylistic score is also helpful. This is mainly because the stylistic patterns of title are important clues for identifying clickbaits, but these clues may not be captured by the content modeling module. Thus, the title stylistic score can provide complementary information to help detect clickbaits better. These results verify the effectiveness of the four different clickbait scores in our approach.

[3]Most results of baselines are taken from (Dong et al., 2019), except the result of Siamese Net on the *Clickbait Challenge* dataset since it is quite unsatisfactory. We report the results using our implementation instead.



(a) *Clickbait Challenge* dataset. (b) *FNC* dataset.

Figure 3: Influence of removing different scores in clickbait prediction.



(a) *Clickbait Challenge* dataset. (b) *FNC* dataset.

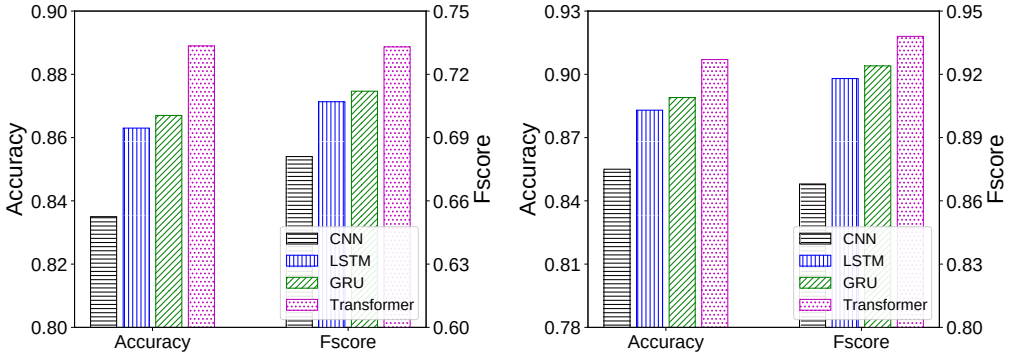
Figure 4: Effectiveness of different attention networks.

4.4 Effectiveness of Attention Mechanism

In this section, we verify the effectiveness of the word-level attention, character-level attention and co-attention networks in our approach. More specifically, we compare the performance of our *SATC* approach and its variants without one kind of attention. The results on the *Clickbait Challenge* and *FNC* datasets are respectively shown in Figs. 4(a) and 4(b). We find that the word-level attention network is very helpful. This may be because different words are usually diverse in their informativeness and the work-level attention networks can attend to the important words in title and body, which can help learn more informative representations of them. In addition, the co-attention network can also effectively improve the model performance. This may be because the co-attention network can model the interactions of words in title and body and can further enhance the title and body representations by encoding interaction information, which is beneficial for evaluating the relevance between title and body. Besides, the character-level attention network can also improve the performance to some extent. This may be because different characters also have different importance in modeling the stylistic patterns of the title and the character-level attention network is able to select useful characters, which can help learn more informative style-aware title representations.

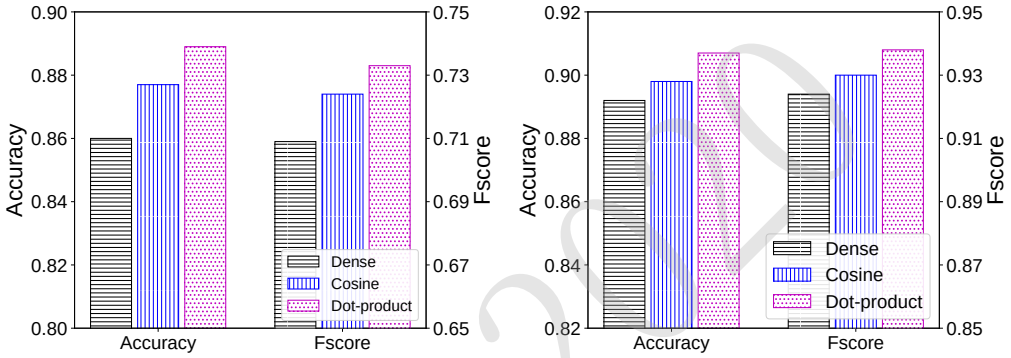
4.5 Effectiveness of Transformer

In this section, we verify the effectiveness of Transformers in text modeling in our approach. We compare the performance of *SATC* and its several variants using CNN, LSTM and GRU for text modeling, and the results are illustrated in Figs. 5(a) and 5(b). From the results, we find that using CNN is not optimal in text modeling for clickbait detection. This is because CNN can only capture local contexts, while the



(a) *Clickbait Challenge* dataset. (b) *FNC* dataset.

Figure 5: Effectiveness of Transformer in text modeling.



(a) *Clickbait Challenge* dataset. (b) *FNC* dataset.

Figure 6: Influence of using different methods for computing matching scores.

long-distance contexts are not considered. In addition, we find GRU slightly outperforms LSTM. This may be because the GRU networks contain fewer parameters and have a lower risk of overfitting. Besides, Transformer outperforms LSTM and GRU. This is because Transformer is very effective in modeling the relations between contexts, which has also been validated by existing works (Vaswani et al., 2017). Thus, we prefer Transformer for learning text representations for clickbait detection.

4.6 Influence of Matching Methods

In this section, we explore the influence of using different methods to implement the matcher in our approach to compute the matching score. We compare the performance of *SATC* using dot-product, dense network and cosine similarity as the matcher. The results are illustrated in Figs. 6(a) and 6(b). From the results, we find that using a dense network is not optimal. According to (Rendle et al., 2020), a possible reason is that dense network is difficult to measure the similarity between two vectors, and thereby the matching score may be inaccurate. In addition, we find that using dot-product is slightly better than using cosine similarity. This may be because the cosine similarity function is not sensitive to the length of the input vectors, which may not be optimal for measuring the relevance between the title and body. Thus, we choose dot-product to implement the matcher in our method.

4.7 Case Study

In this section, we conduct several case studies to better understand the characteristics of our approach. The title, body, groundtruth and the predictions results of *GRU-Att*, *LSDA* and our *SATC* on several samples are shown in Table 2, and we have several findings. In Table 2, the first sample is a clickbait because its title does not match its body. However, since the *GRU-Att* method only considers the information of

Title	Body	Label	Prediction		
			GRU-Att	LSDA	SATC
Report: NHL expansion to Las Vegas'a done deal'	Brain surgery recovery can be a gamble, but not everybody wakes up in the middle of the procedure...	1	0.07	0.88	0.95
The real-life Indiana Jane will make you sooooooooo jealous of her life	Meet the real-life Indiana Jane: American adventurer spends her life in dangerous jungles and uncharted wildernesses...	1	0.23	0.16	0.98
Apple Watch may be available outside US shortly after launch	Lately, Apple CEO has been making the rounds in Europe, stopping at various stores and chatting with employees. The last time we heard anything about his commentary on Apple Watch...	0	0.12	0.68	0.05

Table 2: The titles, bodies, labels and the predicted scores of different methods on several samples. 0 stands for non-clickbait and 1 stands for clickbait.

title, it fails to detect this clickbait. The other two methods that consider the relevance between title and body classify this sample correctly. Thus, it is important to model the title-body relevance for clickbait detection. The title of the second sample in Table 2 contains a word with repeated characters to express strong emotion, which is an important indication of clickbaits. However, this word is out-of-vocabulary, making it difficult for the *GRU-Att* and *LSDA* methods to capture this clue. Thus, these methods fail to detect this clickbait. Different from them, our approach uses a character-level Transformer to capture the stylistic patterns in the title, and thereby can detect this clickbait at a high confidence. The third sample in Table 2 is not a clickbait because the title is formal and the title is relevant to the body. However, it is not easy to measure the relevance between the title and body of this sample without considering the interactions between their words, since the body does not frequently mention the words like "US" and "Watch" that appear in the title. Thus, the *LSDA* method, which does not consider the interactions between contexts, incorrectly classifies this sample as a clickbait. Since our approach uses a co-attention network to model title-body interactions, it classifies this sample correctly.

5 Conclusion

In this paper, we propose a clickbait detection approach with style-aware title modeling and co-attention, which can capture the stylistic patterns in the title and the interactions between the contexts in the title and body. We use Transformers to learn content representations of title and body, and respectively compute two content-based clickbait scores for them based on their representations. In addition, we propose to apply a character-level Transformer to capture the stylistic patterns of title for learning style-aware title representations, which are further used to compute a title stylistic score. Besides, we propose to use a co-attention network to model the relatedness between the contexts within title and body, and further combine their original representations with the interaction information to learn interaction-enhanced title and body representations, which are further used to compute a title-body matching score. The final clickbait score is predicted by a weighted summation of the four kinds of clickbait scores. Extensive experiments on two benchmark datasets show that our approach can effectively improve the performance of clickbait detection by using style-aware title modeling to capture stylistic information and co-attention networks to model title-body interactions.

Acknowledgments

This work was supported by the National Key Research and Development Program of China under Grant number 2018YFC1604002, the National Natural Science Foundation of China under Grant numbers U1936208, U1936216, U1836204, and U1705261.

References

Amol Agrawal. 2016. Clickbait detection using deep learning. In *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*, pages 268–272. IEEE.

Ankesh Anand, Tanmoy Chakraborty, and Noseong Park. 2017. We used neural networks to detect clickbaits: You won't believe what happened next! In *ECIR*, pages 541–547. Springer.

Prakhar Biyani, Kostas Tsioutsiouliklis, and John Blackmer. 2016. " 8 amazing secrets for getting more clicks": Detecting clickbaits in news streams using article informality. In *AAAI*.

Peter Bourgonje, Julian Moreno Schneider, and Georg Rehm. 2017. From clickbait to fake news detection: an approach based on detecting the stance of headlines to articles. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 84–89.

Xinyue Cao, Thai Le, et al. 2017. Machine learning based detection of clickbait posts in social media. *arXiv preprint arXiv:1710.01977*.

Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. 2016. Stop clickbait: Detecting and preventing clickbaits in online news media. In *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 9–16. IEEE.

Yimin Chen, Niall J Conroy, and Victoria L Rubin. 2015. Misleading online content: recognizing clickbait as" false news". In *Proceedings of the 2015 ACM on workshop on multimodal deception detection*, pages 15–19.

Philogene Kyle Dimpas, Royce Vincent Po, and Mary Jane Sabellano. 2017. Filipino and english clickbait detection using a long short term memory recurrent neural network. In *IALP*, pages 276–280. IEEE.

Manqing Dong, Lina Yao, Xianzhi Wang, Boualem Benatallah, and Chaoran Huang. 2019. Similarity-aware deep attentive model for clickbait detection. In *PAKDD*, pages 56–69. Springer.

Junfeng Fu, Liang Liang, Xin Zhou, and Jinkun Zheng. 2017. A convolutional neural network for clickbait detection. In *2017 4th International Conference on Information Science and Control Engineering (ICISCE)*, pages 6–10. IEEE.

Ayse Geçkil, Ahmet Anil Müngen, Esra Gündogan, and Mehmet Kaya. 2018. A clickbait detection method on news sites. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 932–937. IEEE.

Maria Glenski, Ellyn Ayton, Dustin Arendt, and Svitlana Volkova. 2017. Fishing for clickbaits in social images and texts with linguistically-infused neural network models. *arXiv preprint arXiv:1710.06390*.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *CIKM*, pages 2333–2338.

Vijayasaradhi Indurthi and Subba Reddy Oota. 2017. Clickbait detection using word embeddings. *arXiv preprint arXiv:1710.02861*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Vaibhav Kumar, Dhruv Khattar, Siddhartha Gairola, Yash Kumar Lal, and Vasudeva Varma. 2018. Identifying clickbait: A multi-strategy approach using neural networks. In *SIGIR*, pages 1225–1228.

Shumpei Okura, Yukihiro Tagami, Shingo Ono, and Akira Tajima. 2017. Embedding-based news recommendation for millions of users. In *KDD*, pages 1933–1942.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. 2016. Clickbait detection. In *ECIR*, pages 810–817. Springer.

Steffen Rendle, Walid Krichene, Li Zhang, and John Anderson. 2020. Neural collaborative filtering vs. matrix factorization revisited. *arXiv preprint arXiv:2005.09683*.

Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. A latent semantic model with convolutional-pooling structure for information retrieval. In *CIKM*, pages 101–110.

Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958.

Philippe Thomas. 2017. Clickbait identification using neural networks. *arXiv preprint arXiv:1710.08721*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *NAACL-HLT*, pages 1480–1489.

Hai-Tao Zheng, Jin-Yuan Chen, Xin Yao, Arun Kumar Sangaiah, Yong Jiang, and Cong-Zhi Zhao. 2018. Clickbait convolutional neural network. *Symmetry*, 10(5):138.

Yiwei Zhou. 2017. Clickbait detection in tweets using self-attentive network. *arXiv preprint arXiv:1710.05364*.

Konwledge-Enabled Diagnosis Assistant Based on Obstetric EMRs and Knowledge Graph

Kunli Zhang[1,2], Xu Zhao[1,2], Lei Zhuang[1], Qi Xie[1] and Hongying Zan[1,2]
[1]School of Information Engineering, Zhengzhou University, Zhengzhou, China
[2]Peng Cheng Laboratory, Shenzhen, China
{ieklzhang,ielzhuang,ieqxie,iehyzan}@zzu.edu.cn
zhaox917@163.com

Abstract

The obstetric **E**lectronic **M**edical **R**ecord (EMR) contains a large amount of medical data and health information. It plays a vital role in improving the quality of the diagnosis assistant service. In this paper, we treat the diagnosis assistant as a multi-label classification task and propose a **K**nowledge-**E**nabled **D**iagnosis **A**ssistant (KEDA) model for the obstetric diagnosis assistant. We utilize the numerical information in EMRs and the external knowledge from Chinese Obstetric Knowledge Graph (COKG) to enhance the text representation of EMRs. Specifically, the bidirectional maximum matching method and similarity-based approach are used to obtain the entities set contained in EMRs and linked to the COKG. The final knowledge representation is obtained by a weight-based disease prediction algorithm, and it is fused with the text representation through a linear weighting method. Experiment results show that our approach can bring about +3.53 F1 score improvements upon the strong BERT baseline in the diagnosis assistant task.

1 Introduction

Health service relations on the health of millions of people, and it is a livelihood issue in our country. Specifically in China, which has a huge population, the total amount of medical resources is still insufficient. The imbalance between the supply and demand for medical services is still the focus of China's healthcare industry. Although the implementation of China's Universal Two-child Policy in 2016 achieved many benefits, it also leads to an increase in the proportion of older pregnant women and the incidence of various complications (Yang and Yang, 2016). Compared to the overall supply of the medical industry, the lack of obstetric medical resources is prominent.

Since the issue of the Basic Norms of Electronic Medical Records (Trial) (China's Ministry of Health, 2010) by the National Health and Family Planning Medical Affairs Commission in 2010, medical institutions have accumulated many obstetric Electronic Medical Records (EMRs). EMRs are detailed records of medical activities, dominated by the semi-structured or unstructured texts. There is a lot of medical knowledge and health information in EMRs, which is the core medical big data. The first course record in EMRs can be divided into the chief complaint, physical examination, auxiliary examination, admitting diagnosis, diagnostic basis, and treatment plan. In general, there is not a single diagnosis in the admitting diagnosis, it usually includes normal obstetric diagnosis, medical diagnosis, and complications. As a consequence, the diagnosis assistant task based on the Chinese obstetric EMRs can be treated as a multi-label text classification problem, in which the different diagnoses can be regarded as the variable labels. However, the doctor's diagnosis and treatment process are based on comprehensive clinical experience and knowledge in the medical field to make a diagnosis and formulate a corresponding treatment plan. At the same time, they can also explain the corresponding diagnosis basis to the patient in detail. Therefore, rich clinical experience and solid medical knowledge play a vital role in the diagnosis procedure. In order to simulate the diagnosis and treatment process of doctors, we need to introduce external knowledge that

is not available in EMRs. The introduction of medical domain knowledge requires formal expression so that it can be easily used in the diagnosis assistant model. To solve this problem, we adopt the Chinese Obstetric Knowledge Graph (COKG)[0] to introduce external medical domain knowledge.

In this paper, we use the BERT (Bidirectional Encoder Representation from Transformers) (Devlin et al., 2019) to generate the text representation of EMRs. The numerical information in EMRs is also important for the diagnosis results, it is being used to enhance the text representation with the multi-head self-attention (Vaswani et al., 2017). For entity acquisition, we compare the bidirectional maximum matching method and the Bi-LSTM-CRF method respectively, and choose the former method to obtain the entity sets from EMRs. Then the entities are linked to the COKG by a similarity-based method. Due to the fact that the negative words in EMRs will have an impact on the semantics, we employ a negative factor to deal with the negative words in EMRs and propose a weight-based disease prediction algorithm to obtain the final knowledge representation. Finally, a linear weighting method is employed to fuse the text representation and knowledge representation. The experiments on the Obstetric First Course Record Dataset support the effectiveness of our approach.

The main contributions of this paper are summarized as follows:

- In this paper, we propose the KEDA (**K**nowledge-**E**nabled **D**iagnosis **A**ssistant) model to integrate external knowledge from COKG into diagnosis assistant task.
- A weight-based disease prediction algorithm named WBDP is used to limit the influence of negative words in EMRs and generate the final knowledge representation.

2 Related Work

In this paper, we treat the obstetric diagnosis assistant task as a multi-label classification problem. The multi-label classification in traditional machine learning is usually regarded as a binary classification problem or adjust the existing algorithm to adapt to the multi-label classification task (Zhang and Zhou, 2007; Zhang and Zhou, 2006; Read et al., 2011; Tsoumakas et al., 2010).

With the development and application of deep learning, CNN and RNN are widely used in multi-label text classification tasks. For example, Kurata G et al. (2016) use CNN-based word embedding to obtain the direct relationship of the labels. Chen et al. (2017) propose a model that combined CNNs and RNNs to represent the semantic information of the text, and modeling the high-order label association. Baker S and Korhonen A (2017) use row mapping to hide the layers that map to the label co-occurrence based on a CNN architecture to improve the model performance. Ma et al. (2018b) propose a multi-label classification algorithm based on cyclic neural networks for machine translation. Yang et al. (2018) propose a Sequence Generation Model (SGM) to solve the multi-label classification problem. In recent years, the pre-training technology has grown rapidly, ELMo (Peters et al., 2018), OpenAI GPT (Radford et al., 2018), and BERT (Devlin et al., 2019) model have achieved significant improvements in multiple natural language processing tasks. They can be applied to various tasks after fine-tuning. However, due to the little knowledge connection between specific and open domain, these models do not perform well on domain-specific tasks. One way to solve this problem is to pre-train the model on a specific domain, but it is time-consuming and computationally expensive for most users. The models in this way are like ERNIE (Sun et al., 2019), BERT-WWM (Cui et al., 2019), Span-BERT (Joshi et al., 2020), RoBERTa (Liu et al., 2019), XLNET (Yang et al., 2019b), and so on. Moreover, if we can integrate knowledge at the fine-tuning process, it may bring better results. Several studies integrate external knowledge into the model. Chen J et al. (2019) use BiLSTM to model the text and introduce external knowledge through C-ST attention and C-CS attention. Li M et al. (2020) use BiGRU to extract word features, and use a similar matrix based on convolutional neural network and self-entity and parent-entity attention to introduce knowledge graph information. Yang A et al. (2019a) use knowledge base embedding to enhance the output of BERT for machine reading comprehension.

In terms of the diagnosis assistant based on Chinese obstetric EMRs, Zhang et al. (2018) utilize four multi-label classification methods, backpropagation multi-label learning (BP-MLL), random k-labelsets

[0] http:/47.106.35.172:8088/

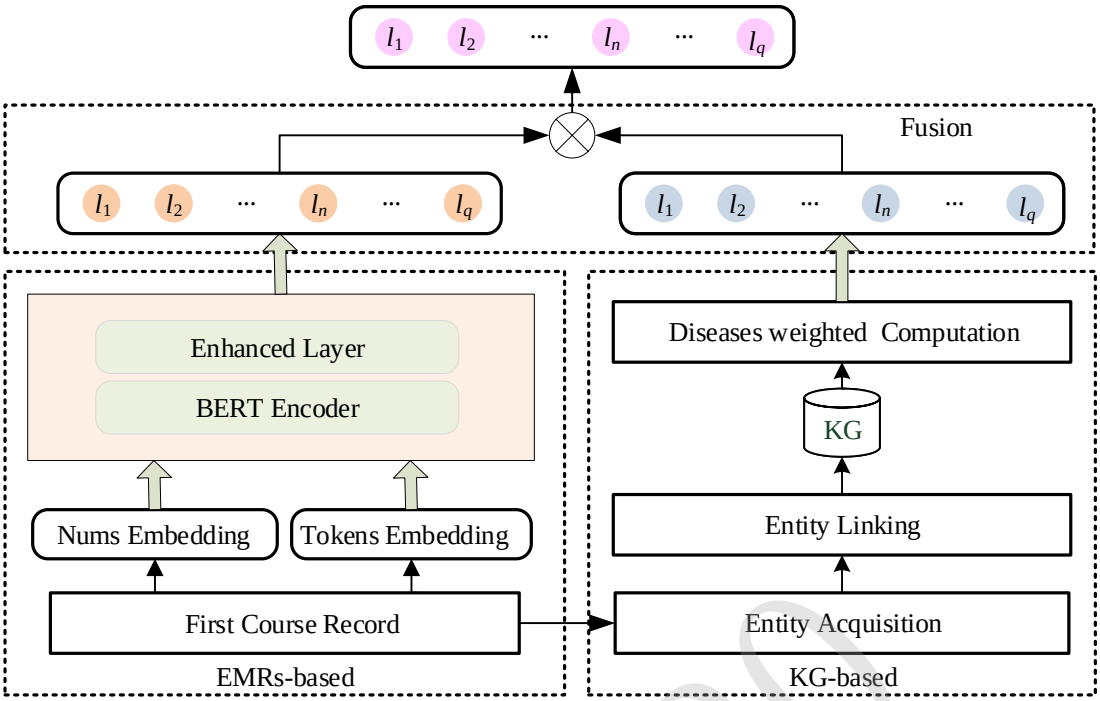


Figure 1: The architecture of the KEDA model

(RAkEL), multi-label k-nearest neighbor (MLKNN), and Classifier Chain (CC) to build the diagnosis assistant models. Ma et al. (2018a) fuse numerical features by employing the concatenated vector to improve the performance of the diagnosis assistant. Zhang et al. (2019) encode EMRs with BERT, and propose an enhanced layer to enhance the text representation for diagnosis assistant.

3 Methodology

3.1 Model Architecture

As shown in Figure 1, the KEDA model can be divided into three parts: EMRs-based module, KG-based module, and Fusion module. For any given EMR, the EMRs-based module generates the text representation by the BERT encoder firstly, then the numerical information contained in EMR is employed to enhance the text representation. Meanwhile, the KG-based module obtains the entities set and links to COKG through the entity acquisition and entity linking methods. Finally, the final knowledge representation is computed by a weight-based disease prediction algorithm and fused with the text representation through a linear weighting method. The following will introduce the implementation details of this model.

3.2 EMRs-based Module

The function of this module is to generate the text representation of EMRs. Similar to the BERT model, the input of KEDA model is composed of four parts: Token embedding, Position embedding, Segment embedding, and Nums embedding which contains the numerical information in EMRs.

BERT encoder

In this paper, we utilize the BERT as an encoder to obtain the text representation of EMRs.The input text sequence is as follows.

$$[CLS]ElectronicMedicalRecordText[SEP]$$

Where $[CLS]$ is a specific classifier token and $[SEP]$ is a sentence separator which is defined in BERT. For the diagnosis assistant task, the input of the model is a single sentence.

Enhanced Layer

The enhanced layer aims to enhance the text representation obtained by the BERT encoder through the numerical information in EMRs. Since the maximum length of the input sequence of BERT is 512, and the average length of EMRs is about 790 characters, we need to reduce the length of the input sequence. The information contained in the EMRs text can be divided into textual information and numerical information. Numerical information usually includes certain examinations or indications characterized by numerical values(For example, it contains the age, body temperature, pulse, respiration, respiration, and so on), which is also important information for diagnosis. So we separately extract the numerical information in EMRs to enhance the textual information, which not only can meet the limit of the input length, but also can better use the numerical information in the EMRs for diagnosis.

Then we adopt a multi-head self-attention proposed in Transformer (Vaswani et al., 2017) to integrate the numerical information into text representation of EMRs, as shown in Equation (1)-(4).

$$Q = K = V = W^S Concat([C]; Num_{1...M}) \tag{1}$$

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2}$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{3}$$

$$[C'] = Concat(head_i, ..., head_h)W^O \tag{4}$$

Where $[C]$ is the hidden layer state representation of [CLS], $[C']$ is the text representation after fusing numerical information. $Num_{1...M}$ is the Nums embedding containing M values, which is obtained by standardizing and normalizing the numerical information in EMRs. W^S, W^Q, W^K, W^V, and W^O are trainable parameters, where $Q \in d^{model}$.

3.3 KG-based Module

Entity Acquisition

Through the analysis of obstetric EMRs, we found that the entities such as symptoms, signs, and diseases in EMRs are high-value information for the intelligent diagnosis, so we mainly identify these entities contained in EMRs.

To achieve better performance, we compared two ways for entity acquisition. One way is a dictionary-based method, the Chinese Symptom Knowledge Base(CSKB)[1], diseases set in ICD-10, and the entity sets of diseases and symptoms in COKG are used as dictionaries. We utilize the bidirectional maximum matching algorithm used in Chinese word segmentation (Gai et al., 2014) for entity acquisition, the obtained set includes a total of 9,836 entities. Another way is to use the Bi-LSTM-CRF model for entity acquisition, the texts labeled when constructing COKG is used as the training corpus. The Detailed analysis of experimental comparison results can be found in section 4.

Entity Linking

For the entity sets obtained above, it is necessary to establish a link relationship with the nodes in the knowledge graph. In this paper, the similarity-based approach is used to link the entities in the knowledge graph.

For a given identified entity E_R, we need to find the n entities that are most similar to the knowledge graph COKG, the set of candidate entities is denote as $S = \{E_{K_1}, E_{K_2}, ..., E_{K_i}, ..., E_{K_n}\}$. Then we calculate the similarity between entities r and k, and select the entity with the highest similarity as the entity linked to COKG. The Levenshtein distance, Jaccard coefficient and the longest common substring are used to calculate the similarity respectively, as shown in Equation (5)-(7).

$$Sim_{ld} = \frac{levE_R, E_{K_i}(|E_R|, |E_{K_i}|)}{max(|E_R|, |E_{K_i}|)} \tag{5}$$

[1] http://www5.zzu.edu.cn/nlp/info/1015/1865.htm

$$Sim_{jacc} = jaccard(bigram(|E_R|), bigram(|E_{K_i}|)) \tag{6}$$

$$Sim_{lcs} = \frac{|lcs(E_R, E_{K_i})|}{max(|E_R|, |E_{K_i}|)} \tag{7}$$

These three similarity algorithms measure the similarity of two entities from different angles, and the average value is used as the final score of the similarity of two entities, as shown in Equation (8).

$$Sim(E_R, E_{Ki}) = (Sim_{ld} + Sim_{jacc} + Sim_{lcs})/3 \tag{8}$$

However, the negative words in EMRs will have an impact on the semantics of components in their jurisdiction. For example, for the descriptions of *There is no discomfort such as vaginal bleeding*(无阴道流血等不适) and *There is involuntary vaginal fluid*(不自主阴道流液) contain the negative words 无 and 不. The first word will change the actual semantics, but the latter word is only a description of *vaginal fluid*.

Therefore, we utilize the negative factor f_{neg} to limit the influence of negative words on semantics. If the negative words that do not change or partially change semantics, the entities described by those words will be linked to COKG, and the negative factor is 1 or 0.5, respectively. For those negative words that will change semantics, their negative factor is -1.

Diseases Weighted Computation

Through entity linking above, we can obtain the symptoms set $S_R = \{s_{R_1}, s_{R_2}, ..., s_{R_i}, ..., s_{R_m}\}$ and the diseases set $D_R = \{(d_{R_1} : f_{R_1}), (d_{R_2} : f_{R_2}), ..., (d_{R_i} : f_{R_i}), ..., (d_{R_q} : f_{R_q})\}$, where f_{R_i} is the frequency of disease entity and $f_{R_1} \leq f_{R_2} \leq ... \leq f_{R_i} \leq ... \leq f_{R_q}$.

Then we propose a weight-based disease prediction algorithm named WBDP. The disease and symptom sets in COKG are denoted as D_K and S_K. Through the matching of tail entities, we can get a set $D_i = \{d_{i_1}, d_{i_2}, ..., d_{i_j}, ..., d_{i_n}\}$ of n candidate disease entities in COKG for symptom s_{R_i}, the disease candidate set corresponding to all symptoms is denoted as D. For each disease d_{ij} in candidate set D, there is a symptom set $S_{d_{ij}} = \{s_{d_{ij}1}, s_{d_{ij}2}, ..., s_{d_{ij}l}, ..., s_{d_{ij}M}\}$ containing m symptoms in COKG associated with it, and $Q_{ij} = S_R \cap S_{d_{ij}}$. The purpose of WBDP is to compute the weight of disease d_{ij}, as shown in Equation (9).

$$W_{d_{ij}} = \sum_{s_{R_i} \in S_R} \frac{f_{neg} \times p(s_{R_i}, d_{ij})}{\sum_{q_r \in Q_{ij}} p(q_r, d_{ij})} \log_2 \frac{|D|}{|D_i| + 1} \tag{9}$$

Where $|D_i|$ and $|D|$ are the number of diseases in set D_i and D, f_{neg} is the negative factor of s_{R_i}, $p(s_{R_i}, d_{ij})$ is the co-occurrence probability of symptom s_{R_i} and disease d_{ij} in COKG.

We adopt two methods to deal with the disease set D_R contained in EMRs. If the disease negative factor f_{neg} is -1, it will be removed from the candidate set. Otherwise, if the candidate set associated with symptoms already contains d_{R_i}, the weight $W'_{d_{R_i}}$ will be computed according to the $W_{d_{R_i}}$ and the frequency f_{R_i}, as shown in Equation (10).

$$W'_{d_{R_i}} = W_{d_{R_i}}(1 + \frac{f_{R_i}}{\sum_{f_{R_i} \in D_R} f_{R_i}}) \tag{10}$$

If the candidate set associated with symptoms does not contain d_{R_i}, it will be add to the candidate set. Its weight is β times of the average weight, where β is a hyper-parameter and $\beta \geq 1$, the Equation is shown in (11). It is means that the diseases in EMRs have more influence on the diagnosis results than the symptoms.

$$W_{d_{R_i}} = f_{neg} \times \frac{\beta}{|D|} \sum_{d_i \in Dise} W_{d_i} \tag{11}$$

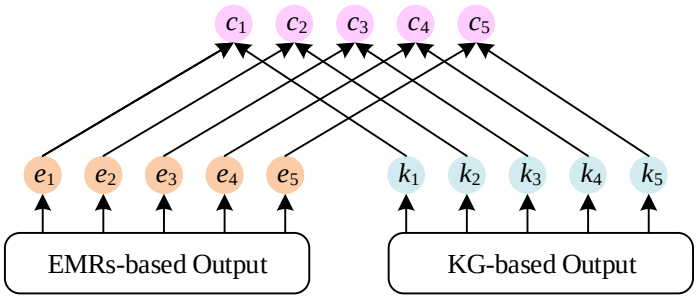


Figure 2: The fusion module of KEDA model

3.4 Fusion Module

The fusion module is aimed to integrate the output of the KG-based module into the output of the EMRs-based module. Inspired by the method proposed by (Chen et al., 2019), we employ a linear weighting method to fuse those representations, as shown in Figure 2.

The output of KG-based module and EMRs-based module is denoted as $K = [k_1, k_2, ..., k_i, ..., k_q]$ and $E = [e_1, e_2, ..., e_i, ..., e_q]$, where k_i is the normalized representation of the weights mentioned above. The fusion process is shown in Equation (12).

$$c_i = \sigma(\gamma_i e_i + (1 - \gamma_i)k_i) = \frac{1}{1 - \exp(-(\gamma_i e_i + (1 - \gamma_i)k_i))} \tag{12}$$

Where σ is the sigmoid function, γ can be seen as a soft switch to adjust the importance of two representations. There are various ways to set the γ. The simplest one is to treat γ as a hyper-parameter and manually adjust. Alternatively, it can also be learned by a neural network automatically, as shown in Equation (13).

$$\gamma = \sigma(W^T[K; E] + b) \tag{13}$$

Where W and b are trainable parameters.

3.5 Training

To train the KEDA model, the objective function is to minimize the cross-entropy in Equation (14).

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log P_i + (1 - y_i)\log(1 - P_i))] \tag{14}$$

Where $y_i \in \{0, 1\}$, N is the number of labels, and P is the model's prediction.

4 Experiments

4.1 The Procedure of Diagnosis Assistant

As shown in Figure 3, the procedure of diagnosis assistant can be divided into four parts: entity acquisition, entity linking, disease weighted computation, and weights fusion. For any given EMR, we obtain the entity sets through entity acquisition firstly, then the entities in those sets are linked to the COKG by a similarity-based method. As a result, we can get the disease nodes set and symptom nodes set from COKG. The WBDP algorithm is employed to compute the disease weights, and the negative factor f_neg is used to limit the influence of negative words in EMRs for disease or symptom entities. Ultimately, the disease weights are regarded as the final knowledge representation to fuse the text representation so that we can get the diagnosis results.

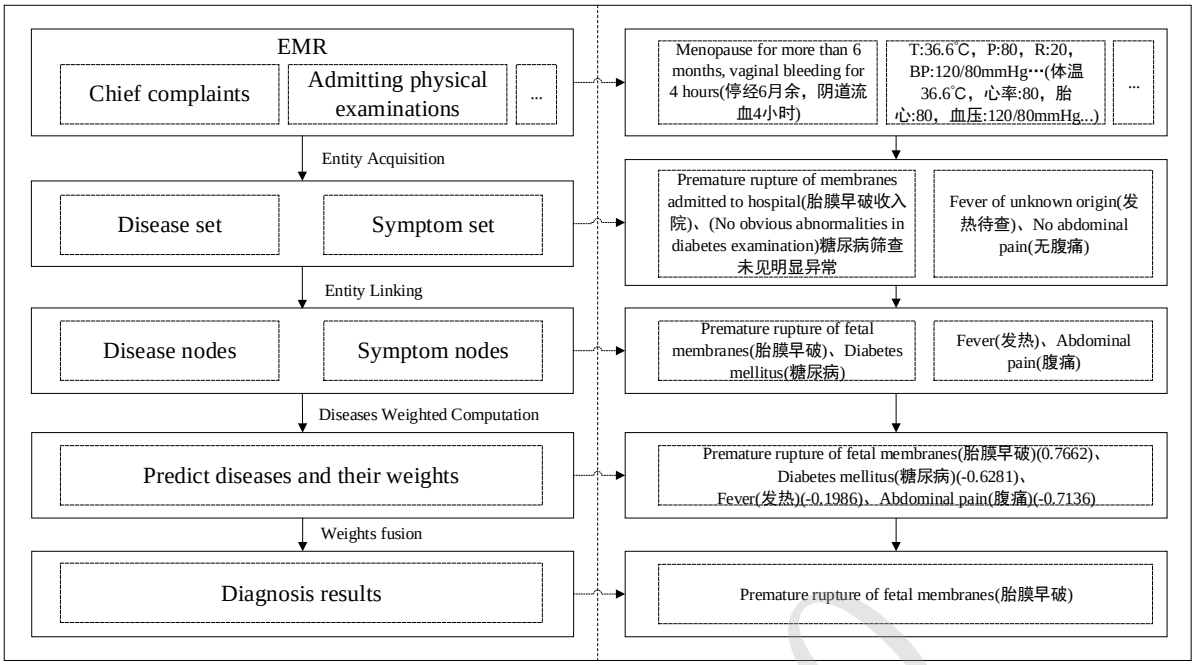


Figure 3: The procedure of diagnosis assistant

4.2 Dataset

We conducted experiments on the obstetric first course record dataset and COKG.

Obstetric First Course Record Dataset. The first course records include 24,339 EMRs from multiple hospitals in China. They were pre-processed through the steps of anonymization, data cleaning, structuring, and diagnostic label standardization. 21,905 of them were used for training and 2,434 were used for testing.

COKG. COKG uses the MeSH-like framework as the knowledge ontology to define the entity and relationship description system with obstetric diseases as the core. It contains knowledge from various sources such as the professional thesaurus, obstetrics textbooks, clinical guidelines, network resources, and other multi-source knowledge. COKG includes a total of 15,249 kinds of relations. Among them, 5,790 kinds of relations are semi-automatically extracted, and 9,459 kinds of relations are automatically extracted. The number and source of relations are shown in Table 1.

4.3 Experimental Setup

In this paper, the EMRs are preprocessed by de-identifying, data cleaning, structuring, data filtering, and standardization of diagnostic labels. During the data filtering process, the information that is duplicated and has little effect on the diagnosis is removed. On the one hand, it can meet the limitation of the input length of the BERT model, and on the other hand, it can also retain the useful information. The version of BERT model we used is BERT-base-Chinese, the main parameters are hidden size 768, max position embedding 512, num attention heads 12, num hidden layers 12, maximum input length 512, learning rate 5e-5, batch size 6, training epoch 20. All our experiments are run on an RTX2080ti GPU(12G).

4.4 Results

Experimental results on the obstetric first course record dataset are shown in Table 2. F1 (F1-micro), Hamming Loss, One Error, and AP (Average Precision) were used as evaluation metrics. BERT indicates the results of the baseline Google BERT, SGM is the results of SGM(Sequence Generation Model)(Yang et al., 2018), BERT+A, and BERT+A-AP are from (Zhang et al., 2019), which experiments are carried out on the same dataset as this paper. The KG-based means only use knowledge graph information, and KEDA is our proposed model.

Table 1: The relations statistics in COKG.

Relation	Semi-automatic extraction	Automatic extraction	Total
disease-disease	1,053	942	1,995
disease-symptom	1,680	3,199	4,879
disease-anatomic site	78	63	141
disease-check	529	815	1,344
disease-medicine	447	612	1,059
disease-operation	225	2	227
disease-other treatments	323	0	323
disease-prognosis	17	0	17
disease-epidemiology	160	84	244
disease-sociology	878	367	1,245
disease-others	170	2,889	3,059
disease-synonym	262	486	748

Table 2: The results on obstetric first course record dataset.

Model	F1(%)	Hamming Loss	One Error	AP(%)
SGM	60.00	0.0200	0.0630	39.00
BERT	79.58	0.0132	0.0961	84.97
BERT+A	80.26	**0.0129**	0.0863	85.42
BERT+A-AP	80.28	**0.0129**	0.0891	85.74
KG-based	53.57	0.0220	0.2417	52.13
KEDA	**83.11**	0.0143	**0.00152**	**88.90**

From Table 2, it can be seen that the improvements in our model over the BERT baseline and other results from (Zhang et al., 2019) are significant and consistent overall evaluation metrics. The AP of KG-based is only 52.13%, which is far lower than the result of KEDA. There may be two reasons for this situation, one of them may be some diagnoses are not obstetric diseases. Another possibility is that COKG is constructed from multi-source texts, which have different levels of detail for different diseases, it may make the number of triples of some diseases insufficient for accurate prediction.

Although the KG-based method does not have an advantage in various indicators, the results of the KEDA are better than BERT and others, indicating that the fusion of knowledge graph can improve the performance of diagnosis assistant. By further analyzing the diagnostic labels in the results, we find that the integration of knowledge graph is more obvious for the improvement of low-frequency labels. For example, the label *Placental abruption*(胎盘早剥) only appeared 5 times in the dataset, due to the scarcity of samples, it is difficult to make accurate predictions using only the BERT-based method. But there are 47 triplets in COKG that describe its symptoms, signs, and related diseases. After introducing the corresponding knowledge graph information, the accuracy of this type of disease has been significantly improved.

4.5 The Results of Entity Acquisition

As mentioned above, in order to choose a better entity acquisition method, we compared the bidirectional maximum matching and Bi-LSTM-CRF on the manually labeled 100 EMRs, the results are shown in Table 3. It is can be seen that the effect of the bidirectional maximum matching method is better than Bi-LSTM-CRF in testing. Bi-LSTM-CRF is trained on texts such as obstetric teaching materials, national norms, clinical practice, etc.

The differences in training data and test data may have an impact on the effectiveness of the model. The dictionaries of the bidirectional maximum matching method come from CSKB and ICD-10, which are more suitable for the description and content in obstetric EMRs. This may be one of the reasons for

Table 3: The results of entity acquisition.

Method	**F1(%)**	**P(%)**	**R(%)**
Bidirectional Maximum Matching	89.42	85.20	94.10
Bi-LSTM-CRF	86.53	88.10	85.03

Table 4: The setting of hyper-parameter γ on KEDA.

γ	**F1(%)**	**P(%)**	**R(%)**	**AP(%)**
0.1	62.46	63.25	60.23	64.70
0.3	64.24	65.32	63.68	66.57
0.5	75.30	77.38	74.19	78.95
0.7	77.23	79.86	74.52	80.90
0.9	71.25	73.19	68.26	74.28
Trained	**83.11**	**87.21**	**79.36**	**88.90**

its better effect on entity acquisition.

4.6 The Setting of Hyper-Parameter γ

The goal of this part is to verify the effectiveness of the fusion module. Firstly, We manually tune the hyper-parameter γ to explore the relative importance of EMRs-based and KG-based. We adjust γ from 0 to 1 with an interval of 0.2, and the results are shown in Table 4. When γ is equal to 0 or 1, the model will become the KG-based or EMRs-based, its results can be found in Table 2. From these results, the model with $\gamma = 0.7$ performs best. When γ gradually increases, the model performs better, but after 0.7, the performance of the KEDA will decline. This shows that too much introduction of knowledge will also affect the overall performance of the model.

Moreover, the hyper-parameter γ is treated as a trainable parameter to train with the model, the results are shown in the last row of Table 4. Compared with manual adjustment, the way to use γ as a trainable parameter is a better choice.

4.7 Error Analysis

In this section, we analyze the bad cases induced by our KEDA model. Most of bad cases can be divided into two categories.

First, some entities in EMRs are not obstetric disease or symptom, which can not find their corresponding nodes in COKG. For example, those entities like *otitis media*(中耳炎), *glaucoma*(青光眼) and so on, there are not enough descriptions in COKG. Thus, the model can not make the correct diagnosis.

Second, COKG is constructed on multi-source obstetric disease texts, which have different levels of detailed description of different diseases. Among them, the proportion of diseases with less than 10 triplets accounts for more than 60%. If some diseases have fewer triplets in COKG, the model also cannot achieve good performance.

5 Conclusion

In this paper, the obstetric diagnosis assistant task is treated as a multi-label classification problem. We propose a KEDA model for this task, which integrates the numerical information from EMRs and external knowledge from COKG to improve the performance of diagnosis. We utilize the bidirectional maximum matching method to get the entities in EMRs, and the similarity-based approach is used to link the entities in knowledge graph COKG. Then we propose a WBDP algorithm to compute the weights of the entities in the candidate set. Finally, a linear weighting method is employed to fuse the text representation and knowledge representation. The results on the obstetric EMRs support the effectiveness of our approach compared to the BERT model. It turns out that even though the pre-training of BERT

involves a large number of corpora, the knowledge graph of the specific domain can still provide useful information.

In the future, we will incorporate more valuable information into deep neural networks to further improve the performance of the diagnosis assistant. We find that some disease entities in EMRs are not included in COKG(For example, the disease entity 'patella fracture' is a diagnosis label in EMRs, but it is not an obstetric disease), to introduce other knowledge graphs that contain more disease entities is an effective feature for diagnosis.

Acknowledgements

This work has been supported by the National Key Research and Development Project (Grant No. 2017YFB1002101), Major Program of National Social Science Foundation of China (Grant No. 17ZDA138), China Postdoctoral Science Foundation (Grant No. 2019TQ0286), Science and Technique Program of Henan Province (Grant No. 192102210260), Medical Science and Technique Program Co-sponsored by Henan Province and Ministry (Grant No. SB201901021), Key Scientific Research Program of Higher Education of Henan Province (Grant No. 19A520003, 20A520038), the MOE Layout Foundation of Humanities and Social Sciences (Grant No. 20YJA740033), and the Henan Social Science Planning Project (Grant No. 2019BYY016).

References

Simon Baker and Anna Korhonen. 2017. Initializing neural networks for hierarchical multi-label text classification. In *BioNLP 2017*, pages 307–315, Vancouver, Canada,, August. Association for Computational Linguistics.

Guibin Chen, Deheng Ye, Zhenchang Xing, Jieshan Chen, and Erik Cambria. 2017. Ensemble application of convolutional and recurrent neural networks for multi-label text categorization. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2377–2383. IEEE.

Jindong Chen, Yizhou Hu, Jingping Liu, Yanghua Xiao, and Haiyun Jiang. 2019. Deep short text classification with knowledge powered attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6252–6259.

China's Ministry of Health. 2010. Basic specification of electronic medical records (trial). Technical Report 3.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-training with whole word masking for chinese bert. *arXiv preprint arXiv:1906.08101*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Rong Li Gai, Fei Gao, Li Ming Duan, Xiao Hui Sun, and Hong Zheng Li. 2014. Bidirectional maximal matching word segmentation algorithm with rules. In *Advanced Materials Research*, volume 926, pages 3368–3372. Trans Tech Publ.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Gakuto Kurata, Bing Xiang, and Bowen Zhou. 2016. Improved neural network-based multi-label classification with better initialization leveraging label co-occurrence. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 521–526.

Mingchen Li, Gabtone Clinton, Yijia Miao, and Feng Gao. 2020. Short text classification via knowledge powered attention with similarity matrix based cnn. *arXiv preprint arXiv:2002.03350*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Hongchao Ma, Kunli Zhang, and Yueshu Zhao. 2018a. Study on obstetric multi-label assisted diagnosis based on feature fusion. *Journal of Chinese Information Processing*, 32(5):128–136.

Shuming Ma, Xu Sun, Yizhong Wang, and Junyang Lin. 2018b. Bag-of-words as target for neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 332–338, Melbourne, Australia, July. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Time Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning. *Technical report, OpenAI*.

Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. 2011. Classifier chains for multi-label classification. *Machine learning*, 85(3):333.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019. Ernie: Enhanced representation through knowledge integration. *arXiv preprint arXiv:1904.09223*.

Grigorios Tsoumakas, Ioannis Katakis, and Ioannis Vlahavas. 2010. Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering*, 23(7):1079–1089.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Hui-li Yang and Zi Yang. 2016. Effect of older pregnancy on maternal and fetal outcomes. *Chinese Journal of Obstetric Emergency(Electronic Editon)*, 5(3):129–135.

Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. SGM: sequence generation model for multi-label classification. pages 3915–3926.

An Yang, Quan Wang, Jing Liu, Kai Liu, Yajuan Lyu, Hua Wu, Qiaoqiao She, and Sujian Li. 2019a. Enhancing pre-trained language representations with rich knowledge for machine reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2346–2357.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019b. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.

Min-Ling Zhang and Zhi-Hua Zhou. 2006. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE transactions on Knowledge and Data Engineering*, 18(10):1338–1351.

Min-Ling Zhang and Zhi-Hua Zhou. 2007. Ml-knn: A lazy learning approach to multi-label learning. *Pattern recognition*, 40(7):2038–2048.

Kunli Zhang, Hongchao Ma, Yueshu Zhao, Hongying Zan, and Lei Zhuang. 2018. The comparative experimental study of multilabel classification for diagnosis assistant based on chinese obstetric emrs. *Journal of healthcare engineering*, 2018.

Kunli Zhang, Chuang Liu, Xuemin Duan, Lijuan Zhou, Yueshu Zhao, and Hongying Zan. 2019. Bert with enhanced layer for assistant diagnosis based on chinese obstetric emrs. In *2019 International Conference on Asian Language Processing (IALP)*, pages 384–389. IEEE.

Reusable Phrase Extraction Based on Syntactic Parsing

Xuemin Duan[1], Hongying Zan*[1], Xiaojing Bai[2] and Christoph Zahner[3]

[1] School of Information Engineering, Zhengzhou University, Zhengzhou, China
[2] Language Centre, Tsinghua University, Beijing, China
[3] University of Cambridge Language Centre, UK

xueminduan@163.com, iehyzan@zzu.edu.cn, bxj@tsinghua.edu.cn, cz201@cam.ac.uk

Abstract

Academic Phrasebank is an important resource for academic writers. Student writers use the phrases of Academic Phrasebank organizing their research article to improve their writing ability. Due to the limited size of Academic Phrasebank, it can not meet all the academic writing needs. There are still a large number of academic phraseology in the authentic research article. In this paper, we proposed an academic phraseology extraction model based on constituency parsing and dependency parsing, which can automatically extract the academic phraseology similar to phrases of Academic Phrasebank from an unlabelled research article. We divided the proposed model into three main components including an academic phraseology corpus module, a sentence simplification module, and a syntactic parsing module. We created a corpus of academic phraseology of 2,129 words to help judge whether a word is neutral and general, and created two datasets under two scenarios to verify the feasibility of the proposed model.

Keywords: Academic Phraseology Extraction·Academic Phrasebank·Syntactic Parsing.

1 Introduction

The Academic Phrasebank is a general resource created by the University of Manchester for academic writers. And the items of it are neutral and generic, which means that you don't have to worry about accidentally stealing someone else's idea when using these items in your academic paper. The Reusable phrases including the phrases in Academic Phrasebank do not have a unique or original construction, not express a special point of view of another writer.

Now, most of the assisted academic writing research focused on automated essay scoring (AES), but different from the ordinary essay prefer to life and social, research article is a scientific record of scientific research result or innovation thinking in theoretical, predictive, and experimental. Research article writing has more rigorous grammar, discourse structure and phraseology. (Davis and Morley, 2018) mentioned that the central of designing teaching activities developed by Academic Phrasebank is the purpose of improving the cognitive ability of student writers to potential plagiarism. Learning academic phraseology can effectively help student writers avoid plagiarism, and student writers can use the learned academic phraseology in their own research article writing, so as to improve their academic writing ability. However, Academic Phrasebank does not cover all academic phraseology in authentic research articles, so it is necessary to extract academic phraseology automatically from more unlabelled text to expand our "Academic Phrasebank".

Plenty of research relating to teaching activities about Academic Phrasebank, but little or nothing that concerns extracting academic phraseology automatically. Therefore, in this paper, we introduce an Academic phraseology extraction model based on constituency parsing and dependency parsing, which aims to extract similar samples with phrases of Academic Phrasebank from unlabelled research articles. The academic phraseology examples are shown in Table 1.

In order to analyze the semantics and structure of unlabelled sentences, we first create a corpus of academic phraseology which including all words of the phrases of Academic Phrasebank. Due to the items of Academic Phrasebank all are general and neutral, the word in a given sentence which also belongs to the corpus of academic phraseology can exist in the extraction result.

As there is no relevant study at present, this paper did not select others' baseline for comparison but created two datasets under two scenarios to verify the feasibility of the proposed model. A dataset is completed from the phrases of Academic Phrasebank, which is more standard, while a dataset is annotated from authentic research articles with more complex sentence structure, and the experimental results demonstrate the different effectiveness of the proposed model on a different dataset.

In brief, the main contributions are as follows:

– We propose a new task, named Academic Phraseology Extraction, which contributes to academic writing and provides valuable phrases for student writers to organise their research articles.

– We propose a model by syntactic parsing for Academic Phraseology Extraction, which considers phrase structure, dependency and semantic analysis of the given sentence.

– We collect sentences from authentic research articles and construct a dataset for Academic Phraseology Extraction with human-annotation. In addition, we also collect phrases from Academic Phrasebank and construct a dataset for Academic Phraseology Extraction with human-completion.

Sentences	**Academic Phraseology**
This paper have argue that the proposed TDNN could be further improved.	This paper have argue that ...
There have been efforts in developing AES approaches based on DNN.	There have been efforts in developing ...
Further study are required to identify the effectiveness of proposed AES.	Further study are required to identify the effectiveness of ...

Table 1: Academic Phraseology Extraction Results

2 Related Work

Corpus of contemporary American English (COCA) is the latest contemporary corpus of 360 million words developed by (Davis, 2008). It covers five types of the corpus of novels, oral English, popular magazines, and academic journals in different periods in the United States. Using COCA to study can make up for the lack of students' understanding of vocabulary, and at the same time, it can cultivate favorable conditions for essay writing. However, the COCA is inappropriate to be used as a corpus for judging whether a word belongs to academic phraseology in the process of academic phraseology extraction. So we create a corpus of academic phraseology for this paper.

Academic Phrasebank is a general resource developed by Dr. John Morley of the University of Manchester to help student writers writing. (Davis and Morley, 2018) has designed some relevant teaching activities developed based on Academic Phrasebank. The research holds that the most important two points of academic writing teaching purpose are to obtain timely writing feedback and improve the cognitive ability of student writer to plagiarism in academic writing. The former means automated research article scoring, and the latter means strengthening students' learning of phrases in Academic Phrasebank and authentic research article. Because the content of Academic Phrasebank is neutral and general, frequent learning of Academic Phrasebank can help students improve their cognitive ability. But the content of Academic Phrasebank is limited. If student writer want to expand their own "Academic Phrasebank", they need to extract academic phraseology from authentic research articles.

The problem of analyzing complex sentences in natural language processing is to make sentences simple to understand, by identifying clause boundaries. Before extracting academic phraseology from a sentence, we choose to simplify the sentence first. (Sharma, 2016) provides a survey of predicting clause boundaries while. (Sacaleanu, 2017) proposed a rule-based method for clause boundary detection. The latter method is a pipeline that uses phrase structure trees to determine the clauses.

3 Our Approach

In this section, we will introduce our academic phraseology extraction approach. There are three main components in our model, i.e., an academic phraseology corpus module to help identify whether a word in a sentence belongs to academic phraseology, a sentence simplification module to prevent incomplete academic phraseology from being extracted, and a syntactic parsing module to determine the final results of academic phraseology extraction. We will introduce the details of our academic phraseology extraction approach as follows.

3.1 Corpus of Academic Phraseology

The academic phraseology extraction is extracting based on the dependency and constituency structure of a sentence, but the final extraction results of two sentences composed of the same dependency and constituency structure are not necessarily the same, because the content of academic phraseology is also related to the semantics of words of a sentence. For example, there are two sentences that only have different subjects, "Further study" and "Bert and transformer". Although they both act as the components of nominal phrases in the sentences, the former can appear in the result, but the latter can not. This is because the content of "Further" and "study" are all neutral and general, but "Bert", "and" and "transformer" have a special word. How to judge whether a word is neutral and general? we need a corpus containing a large number of neutral and general words to help us judge.

The corpus of academic phraseology we created contains all words in Academic Phrasebank, which helps us judge whether a word or phrase should appear in the final result. It has a pivotal role in extracting academic phraseology from the unlabelled text. As the phrases in Academic Phrasebank are all academic phraseology, In the process of academic phraseology extraction, the words of a sentence that appear in Academic Phrasebank can all appear in the result of academic phraseology extraction.

We segmented the phrases in Academic Phrasebank and deleted the repeated words to obtain the corpus of academic phraseology. Academic Phrasebank contains 12,451 phrases, and the resulting corpus of academic phraseology we constructed contains 2,129 words.

3.2 Sentence Simplification

English sentences are mainly composed of subject, predicate, object, attribute, adverbial, complement, and other components, in which the predicate component can only be composed of verbs, and the rest of the sentence components can be composed of words or replaced by clauses. English sentences containing clauses are often long and complex, and it is difficult to extract academic phraseology from them. Therefore, for complex sentences, it is necessary to divide them into simple clauses first.

The sentence simplification is to identify more than two English sentences with more than two clauses, mark the boundary of the clauses, and decompose the complex sentences into many simple sentences. In order to improve the accuracy of academic phraseology extraction, we first simplify the complex sentences before extraction and then extracts the academic phraseology from the simple sentences. This kind of syntactic text simplification is non-destructive. It mainly extracts embedded clauses from sentences with complex structures, so as to rewrite them without affecting their original meanings. This process reduces the average sentence length and complexity, making the text simpler. The key point of sentence simplification is to extract the implied clause from the sentence with a complex structure

In this paper, we identify the relationship between the main sentence and the paratactic or subordinate sentence by constituency parsing, classify the subordinate sentence, determine the optimal clause boundary in the sentence, and extract the clause from the constituency parse tree by using the defined rules.

First, get a constituency parse tree of given complex English sentence, then identify the non-root clausal node of the constituency parse tree (e.g. SBAR, S.) and remove it from the main tree but retain these subtrees, then remove all hanging in the main tree prepositions, subordinate conjunctions and adverbs, the result was simplified sentences. The sentence simplification examples are shown in Table 2.

Sentences	Simplified Sentences
The prompt-dependent models can hardly learn generalized rules from rated essays for nontarget prompts, and are not suitable for the prompt independent AES.	[”The prompt-dependent models can hardly learn generalized rules from rated essays for nontarget prompts.”, ”The prompt-dependent models are not suitable for the prompt independent AES.”]
A supervised model is employed to identify the essays in a given set of essays, and it aims to recognize the essays with the extreme quality in the test dataset.	[”A supervised model is employed to identify the essays in a given set of essays.”, ”A supervised model aims to recognize the essays with the extreme quality in the test dataset.”]
Such relative precision is at least 80% on different prompts so that the overlap of the selected positive and negative essays is fairly small.	[”Such relative precision is at least 80% on different prompts.”, ”The overlap of the selected positive and negative essays are fairly small.”]

Table 2: Sentence Simplification Results

3.3 Syntactic Parsing

Our academic phraseology extraction approach is a rule-based approach using constituency parse tree and dependency tree. By identifying the main verb and determining which nominal phrases of the sentence belongs to academic phraseology by the corpus of academic phraseology, we can easily extract the academic phraseology from the sumolified sentence.

The steps for extracting academic phraseology are explained with the help of the following examples: ”Further study are required to identify the effectiveness of poposed AES.”

Step 1: Obtaining the dependency tree of the given simplified sentence to identify the main verb. The dependency tree is shown in Figure 1, we can get the main verb is ”required”.

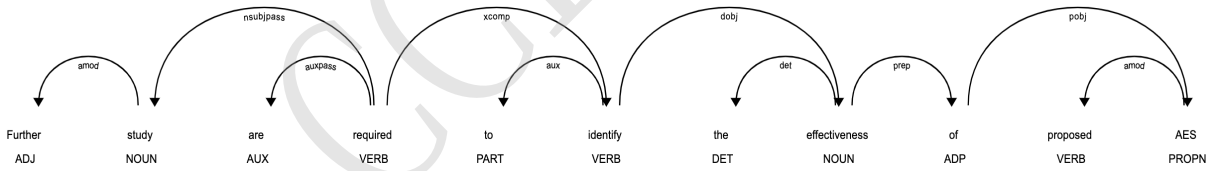


Figure 1: Dependency Parse Tree

Step 2 Obtaining the constituency parse tree to identify all nominal phrases in a sentence and their order. The constituency parse tree is shown in Figure 2.

Step 3 Taking the main verb as the center and classifying the nominal phrases with left part of verb or right part of verb. Then, using the corpus of academic phraseology to determine whether a nominal phrase is deleted or retained. If the left part of the main verb occupied in the corpus of academic phraseology means that it can be retained. The right part of the main verb is divided into several noun phrases and analyzed from the first one. If the first one belongs to the corpus of academic phraseology, then continue to analyze the next one. If not, delete it and the part on its right,and then finish the analysis. All nominal phrases of this sentence and their determines are shown in Table 3.

The first nominal phrase, ”Further study”, can be retained because ”further” and ”study” all exist in the corpus of academic phraseology. So is the second nominal phrase. The third nominal phrase, ”proposed AES”, should be deleted since ”AES” not exist in the corpus of academic phraseology. According to Table 3, we can get the result of academic phraseology extraction is ”Further study are required to identify the effectiveness of ...”

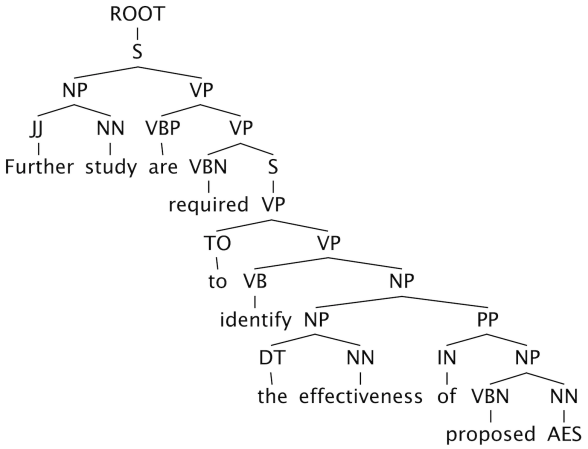


Figure 2: Constituency Parse Tree

	Nominal Phrases	**Judgements**
left part	Further study	retain
right part	the effectiveness	retain
	proposed AES	delete

Table 3: Nominal Phrases Judgements.

4 Experiments

In this section, we present our experiment datasets and results, which devote to answering the following questions that how effective is the proposed academic phraseology extraction model in extracting academic phraseology from sentences written according to the phrases in Academic Phrasebank and whether this model can obtain a same performance in extracting academic phraseology from real academic papers compared to the former.

4.1 Datasets

Since there are not existing academic phraseology dataset now,we created two datasets under two scenarios, ”standard” and ”authentic”, to verify the feasibility of the proposed model. The ”standard” dataset is completed from the phrases of Academic Phrasebank by human. There is no special sentence pattern in sentences completed from Academic Phrasebank,which means that this dataset is more standard. The ”authentic” dataset is annotated from authentic research articles by human. There are some special sentence patterns in sentences of authentic research article, which means that this dataset contains many complex sentence patterns that may appear in authentic research paper, such as inverted sentences and accent sentences. It is more ”authentic”.

We took 1,000 phrases from academic phrasebank and manually completes them into sentences. In addition, we also selected 1,000 complete sentences from authentic research articles and manually annotate their academic phraseology. They are combined together to form the academic phraseology datasets in this paper. The contents are shown in the Table 4.

4.2 Evaluation Metrics

In the process of extracting academic phraseology from sentence, we hope to get more words of our predicted academic phraseology that are the same as those in true academic phraseology. Based on this sense, we calculate Precision, Recall and F score for academic phraseology extraction model.

Datasets	Sentences
Academic Phrasebank Phraseology Dataset	1,000
Authentic Research Articles Phraseology Dataset	1,000

Table 4: Academic Phraseology Extraction Model Datasets.

4.3 Results and Analysis

We use the proposed academic phraseology model to experiment with two datasets, the overall experimental results are shown in Table 5.

From the overall results, we can observe that the performance of the proposed model on Academic Phrasebank Phraseology Dataset is better than on Authentic Research Articles Phraseology Dataset. This is because the academic phraseology extraction model proposed in this paper is designed for the common sentence pattern with the highest frequency in research articles. The Authentic Research Articles Phraseology Dataset has more special sentence patterns, such as inverted sentences and accent sentences.

There is still a lot of room for improvement. If we analyze and modify the proposed academic phraseology extraction model separately for the special sentence patterns that appear less frequently in research articles, the performance of the proposed model on all datasets will be improved.

Datasets	Precision	Recall	F1 score
Academic Phrasebank Reusable Phrases Dataset	0.96	0.62	0.72
Authentic Research Articles Reusable Phrases Dataset	0.84	0.99	0.88

Table 5: The Performance of Reusable Phrase Extraction Model on Different Datasets.

5 Conclusion

In this paper, we define a new task in assisted writing, Academic Phraseology Extraction, which devotes to providing valuable phrases for student writers to write their research articles. For extracting the similar samples with the phrases of Academic Phrasebank, we proposed an academic phraseology extraction model. The proposed model are divided into three components: corpus of academic phraseology, sentence simplification and syntactic parsing. Experiments on a academic phrasebank phraseology dataset and a authentic research article phraseology dataset validate the effectiveness of our approach.

References

Davis, M. and Morley, J. 2018. *Journal of Learning Development in Higher Education ISSN*, p.667X.

Davies, M. 2008. *The corpus of contemporary American English: 450 million words*, 1990-present.

Sharma, S.K. 2016. *International Journal of Computer Applications & Information Technology*,8(2), p.152.

Sacaleanu, B., Marascu, A. and Jochim, C. 2017. *International Business Machines Corp*,U.S. Patent 9,652,450.

Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J.R., Bethard, S. and McClosky, D. 2014. *The Stanford CoreNLP natural language processing toolkit*,(pp.55-60).

Oakey, D. 2020. *Journal of English for Academic Purposes*,44, p.100829.

WAE_RN: Integrating Wasserstein Autoencoder and Relational Network for Text Sequence

Xinxin Zhang
Zhongyuan University of Technology
Zhengzhou, China
2018007088@zut.edu.cn

Xiaoming Liu
Zhongyuan University of Technology
Henan Key Laboratory on Public Opinion Intelligent Analysis ,Zhengzhou, China
ming616@zut.edu.cn

Guan Yang*
Zhongyuan University of Technology
Zhengzhou, China
yangguan@zut.edu.cn

Fangfang Li
oOh! Media
North Sydney, Australia
Fangfang.Li@oohmedia.com.au

Abstract

One challenge in Natural Language Processing (NLP) area is to learn semantic representation in different contexts. Recent works on pre-trained language model have received great attentions and have been proven as an effective technique. In spite of the success of pre-trained language model in many NLP tasks, the learned text representation only contains the correlation among the words in the sentence itself and ignores the implicit relationship between arbitrary tokens in the sequence. To address this problem, we focus on how to make our model effectively learn word representations that contain the relational information between any tokens of text sequences. In this paper, we propose to integrate the relational network(RN) into a Wasserstein autoencoder(WAE). Specifically, WAE and RN are used to better keep the semantic structurse and capture the relational information, respectively. Extensive experiments demonstrate that our proposed model achieves significant improvements over the traditional Seq2Seq baselines.

1 Introduction

Sequence problems are common in daily life that involves DNA sequencing in bioinformatics, time series prediction in Information science, and so on. NLP tasks, such as word segmentation, named entity recognition(NER), machine translation(MT), etc, are actually text sequence problems. For text sequence tasks, it is required to predict or generate target sequences based on the understanding of input source sequence, so it plays a pivotal role in NLP to deeply understand the generic knowledge representation in different context.

To learn the features of input sequences, probabilistic graphical models, such as Hidden Markov Models(HMM) and Conditional Random Field (CRF), can use manually defined feature functions to transform raw data into features, but the quality of the feature functions directly determines the quality of the data presentation.

Because deep learning can automatically learn the useful and highly abstract features of the data via artificial neural network(ANN), many researchers devoted themselves to using Neural Networks(NNs) to obtain low dimensional distributed representations of input data, especially in language modeling, using AutoEncoder(AE) (Rumelhart et al., 1988) to retain the text sequence semantic information in different context has shown promising results. These language models are pretrained on large-scale corpus and complex models to obtain the data representation which contains global information and has strong generalization ability, then the latent representation can be adapted to several contexts by fine-tuning them on various tasks. However, these models simply make use of word order information or position information and ignore the implicit relationship between arbitrary tokens in the sequence, resulting in learning inadequately hidden feature representations and obtaining only superficial semantic representation. More recently, studies on attention (Bahdanau et al., 2015; Luong et al., 2015) and self-attention(Klein and Nabi, 2019; Tan et al., 2018) mechanism demonstrate that it can effectively improve the performance of several NLP tasks by exchanging information between sentences. However, it only

*corresponding author

calculates the contribution between vectors by means of weighted sum without exploring and taking advantage of the implicit structural relationships among tokens.

In this work, we propose add relational networks(RN) (Santoro et al., 2017) to the Wasserstein AutoEncoder(WAE)(Kingma and Welling, 2014) on the basis of the Seq2Seq architecture to collect the complex relationship between objects and retain the semantic structure in sentences. Specifically, to keep the relational information and structural knowledge we add RN layer to encoder since RN integrates the relational reasoning structure that can constrain the functional form of neural network and capture the core common attributes of relational reasoning. To better capture the complex relationships and preserve the semantic structure we use WAE as our encoder because WAE maps input sequences into the wasserstein space that allows various other metric spaces to be embedded in it while preserving their original distance measurements.

The main contributions of our work can be summarized as follows:

1. We put forward an innovative idea to learn more meaningful and structural word representations in text sequneces. We consider relations between objects entail good flexibility and robustness, which are informative and helpful.

2. We propose a WAE_RN model, which integrates WAE and RN to obtain useful and generalized internal latent representations and the implicit relationships in the text sequencse.

3. We conducts experimental verification on two text sequence tasks named entity recognition and EN-GE machine translation. The experimental results demonstrate our proposed model can achive better semantic representation.

2 Related Work

2.1 AutoEncoder

Traditional AutoEncoder(AE) maps the high level characteristics of input data distribution in high dimension to the low(latent vector), and the decoder absorbs this low level representation and outputs the high level representation of the same data. Many researchers have been working on how to get better semantic representations of input sequences, methods using AE such as ELMo(Peters et al., 2018), BERT(Devlin et al., 2019), ALBERT(Lan et al., 2020), ERNIE(Zhang et al., 2019; Sun et al., 2020), XLNet(Yang et al., 2019), etc have been proven as effective techniques. Each model achieves the optimal effect at that time due to its own advantages, and their corresponding pre-trained word vector can still facilitate many downstream tasks even now. However, the latent representation learned by AE is encoded and decoded just in a deterministic way and with no constraint in the hidden space, resulting in a lack of diversity in encoding results, it was later followed by approaches based on VAE(Kingma and Welling, 2014; Bowman et al., 2016) and WAE(Tolstikhin et al., 2018).

VAE converts the potential representation obtained by the encoder into a probabilistic random variable and learn a smooth potential space representation, then the decoder reconstructs the input data and outputs the reconstructed original data. The results have shown that VAE performs competitively compared to traditional AutoEncoder, for example, (Zhang et al., 2016) attempts to use VAE for machine translation, which incorporate a continuous latent variable to model the underlying semantics of sentence pairs. (Shah and Barber, 2018) specifies the prior as a Gaussian mixture model and further develop a topic-guided variational autoencoder (TGVAE) model that is able to generate semantically-meaningful latent representation while generating sentences. However, training on VAE often leads to the disappearance of the KL term. In addition, VAE assumes that the latent variables follow a gaussian distribution, so only a gaussian encoder can be used. To solve these problems, VAE is replaced with WAE by researchers.

Wasserstein Autoencoder (WAE) use the Wasserstein distance that measures the distance between two distributions to replace the KL divergence in VAE to prevent the KL term from disappearing and help the encoder capture useful information during training. Besides, the goal of WAE is to minimize the direct distance between the marginal and the prior distribution and does not force the posterior of each sample to match the prior. In this way, different samples can keep a distance from other samples, which makes

the results generated are more diverse. For instance, (Bahuleyan et al., 2019) propose a WAE variant that use an auxiliary loss to encourage the encoder more stochastic, their studies verified the WAE model achieves much better reconstruction performance. Moreover, (Wang and Wang, 2019) pointed out that the latent space is so complex that we only use standard Gaussian to assume the prior is not enough, and then they proposed to supplement some geometric properties of input space with Riemannian metric tensor to the latent space to learn more flexible latent distribution.

Furthermore, Warstam space is more flexible than Euclidean space, which is helpful for capturing the complex relationships and retaining the semantic structure. Since we focus on capturing the universal semantic representation, we choose WAE as our encoder to generate more meaningful and more flexible latent representation while maintaining the original semantic structure.

2.2 Relational Network

Because Recurrent Neural Network(RNN) gives an output for the input at each moment combined with the current model state, RNN-based model can only learn the sequence relation. While Convolutional Neural Network(CNN) continuously extracts local and overall features through a series of filters, so CNN-based model has poor ability to learn some transformation or relationship. To address this issue, there is a simple solution, that is adding some specific learning modules such as RN to help the model express and learn. RN is a neural network integrated with Relational reasoning structure, which aims to constrain the functional form of the neural network to capture the core common attributes of Relational reasoning. Almost all recent methods focus on using RN to capture the relationships between objects. For example, (Zhang et al., 2018) introduce RN to learn better representations of the input data and experiments on machine translation demonstrate RN can help retain relationships between words. (Chen et al., 2019) also use RN to capture the dependencies within a sentence between any two words and verify the effectiveness of their proposed method on two benchmark NER datasets, which all support that the RN can model relations between the input sequences.

Inspired by the success of the RN in learning the relationships between elements, in this paper, we directly incorporate RN into the WAE models, thus to fully learn the semantic representation and keep the relational information and structural knowledge between sequences to the greatest extent.

3 Preliminary

Since the purpose of the proposed method is to better obtain the semantic representation of text sequences, we will focus on the following two issues.

3.1 The Problem of Sequence Prediction

Sequence prediction is the most basic and widely used task, such as word segmentation, part-of-speech(POS) tagging, named entity recognition(NER), dependency analysis, etc. Essentially, it can be viewed as a matter of classifying each element in a linear sequence according to its context representation. That is, after understanding the input sequence and extracting its useful information, the optimal mark is made for each sequence, and then a set of globally optimal marks is selected for a given sequence at one time.

Suppose we have an input sequence $\vec{x}$ of L elements, and a tag sequence $\vec{y}$ of the same length, i.e. $\vec{x} = (x_1, x_2, \ldots, x_L)^T$, $\vec{y} = (y_1, y_2, \ldots, y_L)^T$, where x_i represents the i-th sequence and y_j represents the j-th tag, it's also requires that the value of y_j is taken from a predefined set of finite tags and i equals j, the final goal is to assign a globally optimal label y_j for each input sequence x_i. End-to-end learning is directly modeling conditional probabilities $p(y|x)$ and then map the input sequence $x_1, x_2, \ldots, x_L$ to the output sequence $y_1, y_2, \ldots, y_L$, i.e.(1).

$$Y = (y_1, y_2, \ldots y_L) = \underset{y}{\operatorname{argmax}}\, p(y|x, \theta) \tag{1}$$

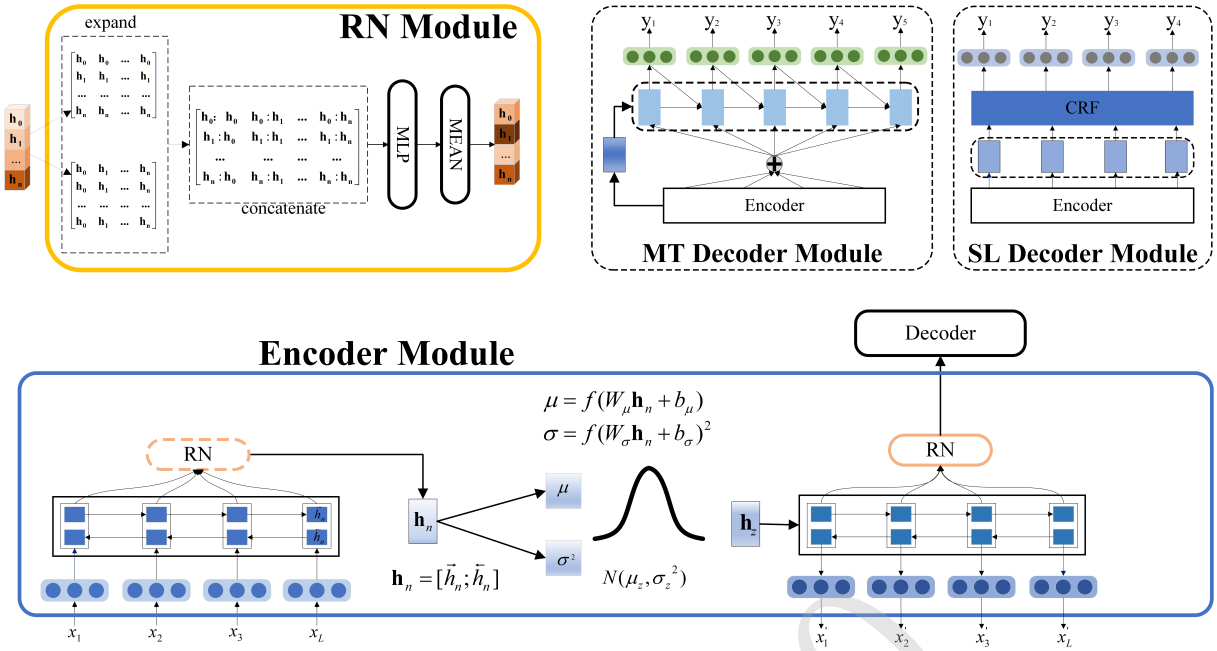


Figure 1: Model architecture

3.2 The Problem of Sequence Generation

Sequence generation is translating the dataset into a clear narrative of human understanding based on the real understanding of text content, such as machine translation, dialogue generation, abstract generation and so on. We usually decompose the generation probability into the product of the generation probability of context-related subsequence, and then use the method of auto-regression to get the text in the form of natural language that human can understand.

Suppose the input sequence is $\vec{x}$, the goal is to understand the input sequence and generate the corresponding output sequence $\vec{y}$, i.e. $\vec{x} = (x_1, x_2, \ldots x_{|X|})^T$, $\vec{y} = (y_1, y_2, \ldots y_{|Y|})^T$, where $|X|$ and $|Y|$ correspond to the length of input sequence and output sequence respectively. Different from sequence prediction, the purpose of sequence-to-sequence learning is to model the conditional probability $p(y|x)$ with all the sequences before the current sequence as the condition, and then map the input sequence to an output sequence, i.e. (2).

$$Y = (y_1, y_2, \ldots y_{|Y|}) = \underset{y}{\operatorname{argmax}}\, p(y|x;\theta) = y \arg\max \left(\prod_{i=1}^{|Y|} p\left(y_i|x, y_{<i};\theta\right) \right) \tag{2}$$

4 Relational Network based WAE Model

4.1 Architecture of Proposed

In order to obtain universal semantic representations that contain structured knowledge, we propose a Relational Network based Wasserstein AutoEncoder (WAE_RN) model, which have the ability to embed the potential structural information contained in sequence into semantic representation. Specifically, a relation network layer is employed to quantify the potential relationships between any two elements in the input sequence, and then these relationships are embedded into the input sequence by WAE to get semantic representation that contains relational information. Finally, the generic representation is sent to different decoders to perform different downstream tasks. Next, we will elaborate our proposed model in detail.

4.2 The Wasserstein AutoEncoder Layer

As shown in the bottom of Fig. 1, the first encoder of WAE collects the semantic information of the data, and the RN module learns the relational information between the outputs of RNNs, then the context representation is mapped to the Wasserstein space. Compared with embedding data into Euclidean space, which is the most common method, WAE embeds the input data into the Wasserstein space as a probability distribution to can help us capture the complex relationship and retain the semantic structure, so we can obtain the distribution $\mathbf{h}_n = \left[\vec{h}_n; \vec{h}_n\right]$ that covers both semantic and relational information of input data $x_1, x_2, \ldots, x_L$. Note that the relational network module can be placed either in front of or behind the first encoder, our experiments showed that it is better for the named entity recognition task to put it in the front while for the machine translation task to put it in the back.

After reparameterizing, the reconstructed hidden state $\mathbf{h}_z = N\left(\mu_z, \sigma_z^2\right)$(where $\mu_z = f(W_\mu \mathbf{h}_n + b_\mu), \sigma_z^2 = f(W_\sigma \mathbf{h}_n + b_\sigma^2)$) is sent to the second encoder in WAE as its initial state, after that this encoder relearns the latent representation of input data under the guidance of the hidden state obtained in the previous step, so as to obtain the semantic representation that both follows the source semantic information and retains the structured information. To fully exploit the relational information, we send the representation learned by the second encoder into the relationship network again.

Different from VAE, WAE can use both Gaussian encoder and deterministic encoder. Besides, the goal of Wasserstein distance is to minimize the direct distance between the marginal distribution and the prior, without forcing the posterior of each sample to match the prior, so that different samples can keep a distance from other samples to produce more diverse results.

4.3 The Relational Layer

The architecture of our RN module is shown in the upper left corner of Fig. 1, different from (Zhang et al., 2018), our RN doesn't use the CNN layer. Besides, to keep the original information of the input sequence to the great extent, we don't use any nonlinear transformations, keeping the dimensions the same. To learn the implicit internal relation between any two elements, we use some transformation between tensors to make objects fully connected and associated with each other, which means, for any vector $C = (\vec{c}_1, \vec{c}_2, \ldots \vec{c}_n)$, after concatenating, its each element $\mathbf{c}_{i,j} = [\vec{c}_i; \vec{c}_j]$. Then we directly calculate the relationships between any objects: $RN(\mathbf{o}_{i,j}) = f_\phi\left(W_{MLP}\mathbf{c}_{i,j} + b_{MLP}\right)$. Here, a multi-layer perceptron is used for f_ϕ to find the relationship between all pairwise objects and judge whether and how they are related.

4.4 The Prediction Layer

There is no difference between the decoder used in our model and the traditional decoder. As shown in the upper right corner of Fig. 1, for machine translation tasks, the decoder is the ordinary RNNs with beam search layer, which generates target sequences one by one in an auto-regressive way, while for the named entity recognition task, the decoder is the RNN network with the CRF layer.

4.5 The Objective

For AE, the training objective is the cross-entropy loss or the reconstruction loss, given by $J_{rec}(\theta, \phi, x) = E_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right]$. In order to compute the loss of our model, we use MMD (given as $MMD = \left\|\int k(z,;)dp(z) - \int k(z,\cdot)dq(z)\right\|_{\mathrm{H}_k}$) to approximate Wasserstein distance, where H_k refers to the Hilbert space defined by the kernel k, for high dimensional Gaussian function, k was usually chosen as the inverse quadratic kernel : $k(x, y) = \frac{C}{C+\|x-y\|_2^2}$.

$$L(\theta; \phi; x) = E_{q(x)}\left[J_{rec}(\theta, x) + \alpha J_{task}(\Phi, x)\right] + \beta MMD \tag{3}$$

Thus the loss function(3) of our model consists of three terms: the first is the reconstruction loss, which encourages the encoder to learn to reconstruct data; the second is the Wasserstein distance between the distribution of the encoder $q_\theta(z|x)$ and prior $p(z)$ (usually p is $N(0, 1)$), which measures how much information is lost when q is represented by p; the third is the task loss between the source input $x_1, x_2, \ldots x_{|X|}$ and the generated target sequences $y_1, y_2, \ldots y_{|Y|}$. However, in the experiment we

observe that the reconstruct loss has a great influence on the results of our model, resulting in poor performance. To address this problem, we impose a weight α(here α is 2) on the translation loss to balance the influence between the task loss and the reconstruct loss. To achieve better performance, we also give another weight β(here β is 0.0001) onMMD. To the end, our model can be trained in an end-to-end manner by minimizing (3).

5 Experiments

In this section, we aim to investigate our model's performance over NER and MT, where NER belongs to the problem of sequence prediction and MT belongs to the problem of sequence generation. We first present our experimental set up, then compare our method to other baseline systems, finally we give some analyses about our method.

5.1 Datasets

We use two benchmark datasets: OntoNotes5.0 Chinese NER dataset(OntoNotes5.0 Ch-NER) and IWSLT2014 German-English dataset(IWSLT14en-de) for evaluation, the details about these corpora are shown in Table1.

Table 1: Statistics of OntoNotes5.0 Ch-NER and IWSLT2014en-de

Dataset	Type	Train	Valid	Test
OntoNotes5.0 Ch-NER	Sentences	53.5k	12.8k	4.5k
	Chars	750k	110k	90k
	Entities	62.5k	9.1k	7.5k
IWSLT2014en-de	Sentences	150k	6.9k	6.7k

5.1.1 OntoNotes5.0 Ch-NER

OntoNotes5.0 Ch-NER contains eleven different entity name types(such as PERSON, NORP, GPE, etc.) and seven different value types(DATE, TIME, MONEY, etc.). We use the same OntoNotes data split used for co-reference resolution in the CoNLL-2012 shared task (Pradhan et al., 2012) and convert the IOB boundary encoding to BIO tagging scheme(B, I, O). We preprocess by filtering out char-level sentences longer than 150 words and replacing all words that appear less than three times with an $< unk >$ token, but for testing data, we use the original dataset.

5.1.2 IWSLT14en-de

IWSLT14en-de contains transcripts of TED talks and translate between German and English in both directions. Following previous works, we use the same data cleanup as (Ranzato et al., 2016). We apply the same tokenization and truecasing using standard Moses scriptsto both our model and baseline. For training data, sentences longer than 50 tokens were chopped and rared words were replaced by a special $< unk >$ token, for testing data, we also use the original version of testing files.

5.2 Experimental Setting

For NER task, we use strong bidirectional Long Short Term Memory with CRF(Bi-LSTM-CRF) baseline, but for MT the baseline is a standard implementation of Bi-LSTM seq2seq model with dot-product attention (Bahdanau et al., 2015; Luong et al., 2015) and for decoding we use a beam width of 10 and limit the max sequence length to 100. Detail hyper-parameters can be found in Table2.

For NER task, we use the entity level accuracy rate, recall rate and F1 value to calculate the score and report standard F1-score for CoNLL NER tasks (Pradhan et al., 2012). For MT task, we adopt BLEU for translation quality evaluation and calculate the BLEU scores on test set using Moses *multi-bleu.perl* script.

Table 2: Hyper-Parameter Settings

Learning rate	$1e^{-3}$
Learning rate decay	0.5
Batch size	64
Clip norm	5.0
Embedding dim	256
Hidden dim	256
Latent dim	32
Dropout	0.3
Uniform init	0.1
Patience	20

Table 3: Corpus BLEU scores (%) on IWSLT14en-de translation tasks

	IWSLT14Ge − En(BLEU)	IWSLT14En − Ge(BLEU)
2017RaphaelShu	29.56	-
2018PoSenHuang	30.08	25.36
2019BryanEikema	28.0	23.4
Ours		
RNN_attn(baseline)	27.84	23.74
RNN_attn_RN	28.18(0.3 ↑)	23.95(0.2 ↑)
WAE(d)_attn	28.55(0.7 ↑)	24.24(0.5 ↑)
WAE(d)_attn_RN	**28.87**(**0.9** ↑)	**24.46**(**0.7** ↑)

5.3 Results and Analysis

In order to enhance the fairness of the comparisons and verify the solidity of our improvement, we train 5 times with random uniform distribution initialization and report average results of our proposed model as well as our re-implemented baselines. Note that we just use simple Seq2Seq architecture as our baseline and don't add any other methods(such as label smoothing, tied embedding, BPE, pre-trained word vector, etc) to the baseline, because our goal is to demonstrate that our proposed method can yield a more general semantic representation, rather than further boost performance.

5.3.1 Results on Machine Translation

For IWSLT14en-de translation tasks, we use deterministic encoder rather than Gaussian encoder for largely alleviating the training difficulties. We show the test results of different models in Table3.

The former lines in the table list the performance of previous methods. (Shu and Nakayama, 2018) propose compress word embedding to directly learn the discrete codes via deep compositional code learning, improving the BLEU scores from 29.45% to 29.56%. Using SleepWAke Networks (SWAN) that is a segmentation-based sequence modeling method to explicitly model the phrase structure in output sequences, (Huang et al., 2018) achieves the state-of-the-art results at that time. (Eikema and Aziz, 2019) use Auto-Encoding Variational NMT model to generate source and target sentences jointly from a shared latent representation, achieving de→en and en→de BLEU scores of 28.0% and 23.4% respectively.

The latter lines show the performance of ours, we can see that our proposed WAE_RN model achieves significant improvement over the baseline system. It demonstrates that our model can capture more useful information and improve the performance of NMT system. In particular, our proposed model outperforms the baseline by 0.9% BLEU points, while only use RN and DAE improves the baseline 0.3% and 0.7% respectively, which effectively illustrate that the combine of RN and WAE can both collect the complex relationship and retain the semantic structure between objects.

Table 4: The evaluation results on OntoNotes5.0 Chinese NER task

method	P(%)	R(%)	F
CoNLL2012	78.20	66.45	71.85
Ours			
BiLSTM_CRF (baseline)	73.08	69.20	71.08
BiLSTM_CRF_SelfAttn	70.93	67.10	68.96
BiLSTM_CRF_LM	72.69	69.59	71.11
BiLSTM_CRF_RN	73.43	69.71	71.52
WAE_CRF	72.79	69.61	71.17
WAE_CRF_RN (best)	73.09	**70.76**	**71.90(0.8 ↑)**

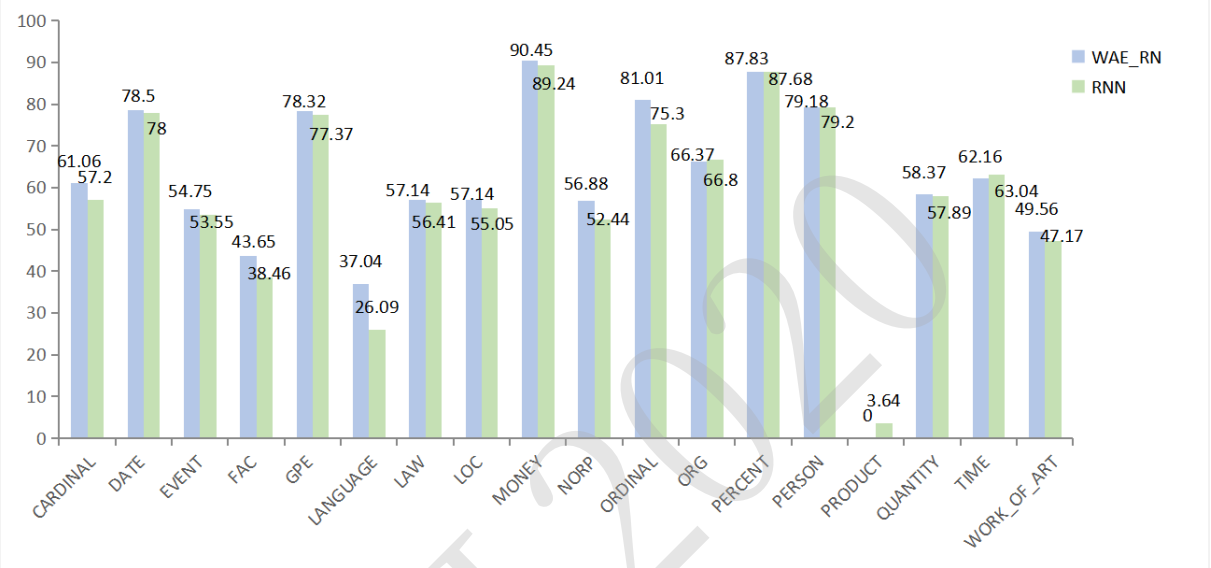


Figure 2: Performance of our model and baseline on each category.

5.3.2 Results on Sequence Labeling

For OntoNotes5.0 Chinese NER task, we use Gaussian encoder. As shown in Table4, the first results is from the CoNLL-2012 Shared Task (Pradhan et al., 2013) and the others are ours, we can observe that WAE_RN can significantly outperforms our re-implemented baseline by 0.8, which demonstrates the robustness of our models. As depicted in Fig. 2, we can see that our method performs well on most categories, such as 'ORDINAL', 'NORP', 'LANGUAGE', etc, and slightly below baseline on the categories of 'PERSON', 'ORG' and 'TIME'. It also should be noted that our model can't find the entity named 'PRODUCT', which is the smallest number of entities in the training dataset. From the results, we can observe that our proposed model does have a positive impact on learning word representation.

Besides, we also conduct experiments using different models to explain the the performance promotion of each module, experimental results on NER task confirm the effectiveness of our proposed model, similar as shown in MT tasks.

6 Conclusion

This paper presents a WAE_RN model for text sequence tasks, which aims at learning word representatiosn containing structured knowledge. To be specific, to preserve the semantic structure between objects, we propose use WAE as the model's encoder. To capture the core common attributes of relational reasoning, we introduce RN. Both of which combine well to learn the generic representation that contains relational information. Experimental results on MT and NER tasks demonstrate that the proposed model leads to significant improvements. In the future, we plan to extend the general representation to transfer

learning.

Acknowledgements

This work was supported by the Science and Technology Planning Project of Henan Province of China(Grant No. 182102210513 and 182102310945) and the National Natural Science Foundation of China(Grant No.61672361 and 61772020).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Hareesh Bahuleyan, Lili Mou, Hao Zhou, and Olga Vechtomova. 2019. Stochastic wasserstein autoencoder for probabilistic sentence generation. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4068–4076. Association for Computational Linguistics.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In Yoav Goldberg and Stefan Riezler, editors, *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 10–21. ACL.

Hui Chen, Zijia Lin, Guiguang Ding, Jianguang Lou, Yusen Zhang, and Börje Karlsson. 2019. GRN: gated relation network to enhance convolutional neural network for named entity recognition. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6236–6243. AAAI Press.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Bryan Eikema and Wilker Aziz. 2019. Auto-encoding variational neural machine translation. In Isabelle Augenstein, Spandana Gella, Sebastian Ruder, Katharina Kann, Burcu Can, Johannes Welbl, Alexis Conneau, Xiang Ren, and Marek Rei, editors, *Proceedings of the 4th Workshop on Representation Learning for NLP, RepL4NLP@ACL 2019, Florence, Italy, August 2, 2019*, pages 124–141. Association for Computational Linguistics.

Po-Sen Huang, Chong Wang, Sitao Huang, Dengyong Zhou, and Li Deng. 2018. Towards neural phrase-based machine translation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Tassilo Klein and Moin Nabi. 2019. Attention is (not) all you need for commonsense reasoning. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4831–4836. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1412–1421. The Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237. Association for Computational Linguistics.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In Sameer Pradhan, Alessandro Moschitti, and Nianwen Xue, editors, *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning - Proceedings of the Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes, EMNLP-CoNLL 2012, July 13, 2012, Jeju Island, Korea*, pages 1–40. ACL.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using ontonotes. In Julia Hockenmaier and Sebastian Riedel, editors, *Proceedings of the Seventeenth Conference on Computational Natural Language Learning, CoNLL 2013, Sofia, Bulgaria, August 8-9, 2013*, pages 143–152. ACL.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence level training with recurrent neural networks. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1988. Learning representations by back-propagating errors. *Nature*, 323(6088):696–699.

Adam Santoro, David Raposo, David G. T. Barrett, Mateusz Malinowski, Razvan Pascanu, Peter W. Battaglia, and Tim Lillicrap. 2017. A simple neural network module for relational reasoning. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 4967–4976.

Harshil Shah and David Barber. 2018. Generative neural machine translation. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pages 1353–1362.

Raphael Shu and Hideki Nakayama. 2018. Compressing word embeddings via deep compositional code learning. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Yu Sun, Shuohuan Wang, Yu-Kun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. ERNIE 2.0: A continual pre-training framework for language understanding. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8968–8975. AAAI Press.

Zhixing Tan, Mingxuan Wang, Jun Xie, Yidong Chen, and Xiaodong Shi. 2018. Deep semantic role labeling with self-attention. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 4929–4936. AAAI Press.

Ilya O. Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schölkopf. 2018. Wasserstein auto-encoders. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Prince Zizhuang Wang and William Yang Wang. 2019. Riemannian normalizing flow on variational wasserstein autoencoder for text modeling. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 284–294. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 5754–5764.

Biao Zhang, Deyi Xiong, Jinsong Su, Hong Duan, and Min Zhang. 2016. Variational neural machine translation. In Jian Su, Xavier Carreras, and Kevin Duh, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 521–530. The Association for Computational Linguistics.

Wen Zhang, Jiawei Hu, Yang Feng, and Qun Liu. 2018. Refining source representations with relation networks for neural machine translation. In Emily M. Bender, Leon Derczynski, and Pierre Isabelle, editors, *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 1292–1303. Association for Computational Linguistics.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: enhanced language representation with informative entities. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1441–1451. Association for Computational Linguistics.