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

第十九届中国计算语言学大会论文集，第719页-第730页，海南，中国，2020年10月30日至11月1日。

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

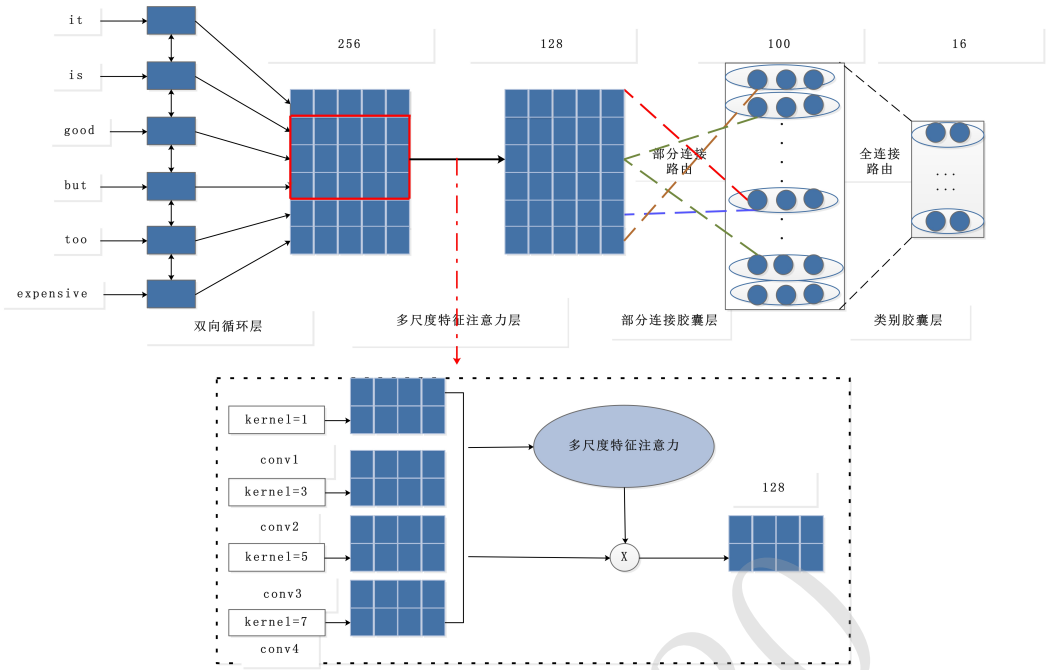

图 1. MulPart-Capsnets模型架构(不同颜色虚线表示不同子胶囊到父胶囊之间的路由)

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 ←$soft\max(b_i)$ (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。

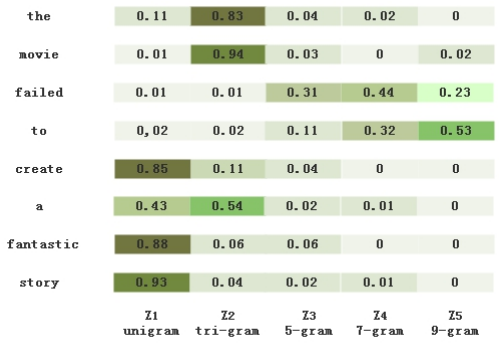

图 2. 多尺度特征注意力的可视化分析

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。

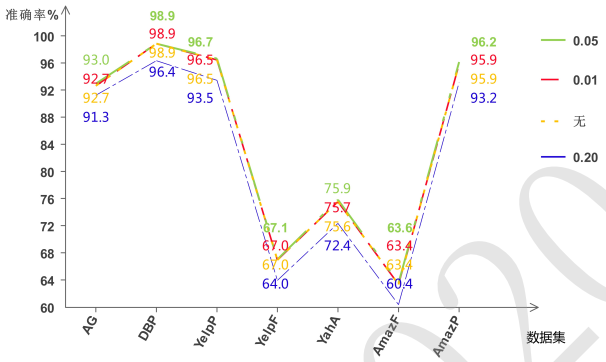

图 3. 权重阈值分析

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.