

CCL23-Eval 任务2系统报告: 基于图融合的非自回归和自回归中文AMR语义分析

辜仰淦*, 周仕林*, 李正华

苏州大学 计算机科学与技术学院, 江苏 苏州

{yanggangu, slzhou.cs}@outlook.com, zhli13@suda.edu.cn

摘要

本文介绍了我们在第二十二届中国计算语言学大会中文抽象语义表示解析评测中提交的参赛系统。抽象语义表示 (Abstract Meaning Representation, AMR) 以有向无环图的形式表示一个句子的语义。本次评测任务针对中文抽象语义表示 (Chinese AMR, CAMR), 参赛系统不仅需要对常规的AMR图解析预测, 还需要预测CAMR数据特有的概念节点对齐、虚词关系对齐、概念同指。我们同时使用多个自回归模型和多个非自回归模型, 然后基于图融合的方法将多个模型输出结果融合起来。最终, 我们在两个赛道共六个测试集上取得了五项第一名, 一项第二名。

关键词: 中文抽象语义表示; 自回归; 非自回归; 图融合

System Report for CCL23-Eval Task 2: Autoregressive and Non-autoregressive Chinese AMR Semantic Parsing based on Graph Ensembling

Yanggan Gu*, Shilin Zhou*, Zhenghua Li

School of Computer Science and Technology, Soochow University, Suzhou, China

{yanggangu, slzhou.cs}@outlook.com, zhli13@suda.edu.cn

Abstract

This paper introduces the system we submitted in the shared task of Chinese Abstract Meaning Representation (CAMR) at the Twenty-two Chinese National Conference on Computational Linguistics. The participating systems need to parse not only conventional AMR graphs, but also alignments between concept nodes and words, alignments between relations and functional words, and coreference between concept nodes, which are unique to CAMR. We use multiple autoregressive and non-autoregressive models, and then fuse the multiple model outputs based on the graph ensemble approach. In the end, we won five first places and one second place in a total of six test sets on two tracks.

Keywords: Chinese Abstract Meaning Representation, Autoregressive, Non-autoregressive, Graph ensemble

*两位对本文的贡献相等。

©2023 中国计算语言学大会

根据《Creative Commons Attribution 4.0 International License》许可出版

项目资助: 国家自然科学基金 (62176173)、江苏高校优势学科建设工程资助项目

1 引言

抽象语义表示 (Abstract Meaning Representation, AMR) 是一种与领域无关的、通用的整句语义表示方法, 该方法使用单根有向无环图来表示一个句子的语义结构(Banarescu et al., 2013)。AMR作为自然语言处理领域的重要任务, 被广泛应用于下游任务如文本摘要 (Liao et al., 2018; Chen et al., 2021), 机器翻译(Song et al., 2019), 对话系统(Bai et al., 2021)等。

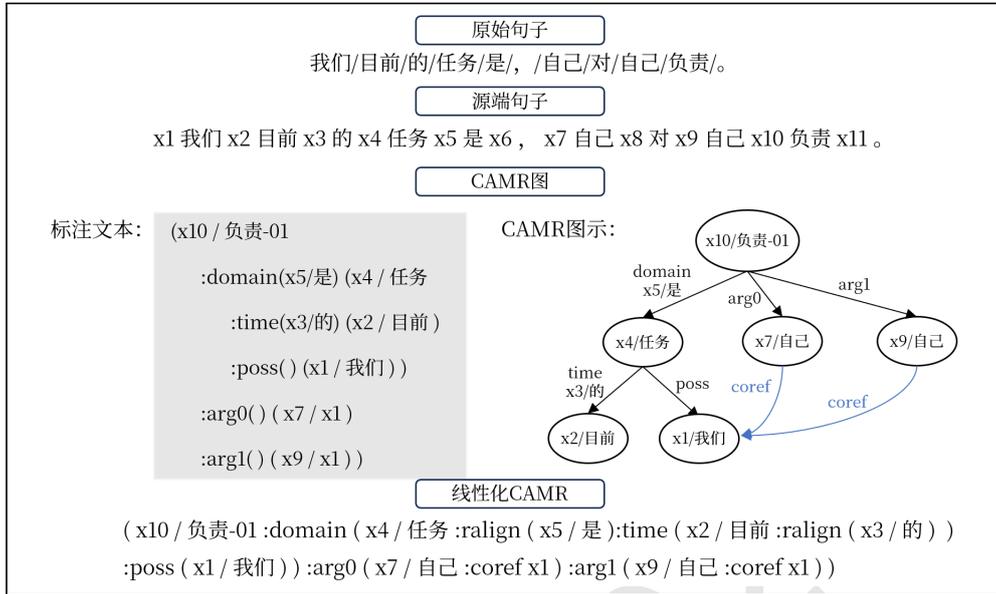


Figure 1: CAMR图及其线性化

本次评测任务针对中文抽象语义表示 (Chinese AMR, CAMR)。与英文AMR不同, CAMR数据集具有三个重要特性。

- (1) 显式刻画并评价输入词语与概念的显式对齐关系(Li et al., 2019)。输入句子分词后, 每个词语得到一个编号, 采用“x+数字”的形式。如果一个概念节点对应一个词语, 则将该词的编号作为对应的概念节点的编号。以图1为例, 概念节点“负责-01”对应编号为“x10”的词语, 即“负责”。概念节点中“-01”表示是谓词的词义编号。值得注意的是, 有些概念节点 (如表示并列概念的概念节点) 并不由句子中的词语触发, 因此没有对齐关系; 对于这样的概念节点, CAMR会赋予一个虚拟的编号 (如“x26/and”)。
- (2) 显式刻画并评价虚词与关系的对齐。汉语句子中的虚词是一类特殊的概念节点, 它们有时会对应CAMR中的某个关系。这种情况下, CAMR会显式标注这种对应关系。以图1为例, 虚词“的”触发了表示时间的关系: time, 因此该词 (连同编号) 会一起标注在边上, 即“: time(x3/的)”。类似的, 可以解释“: domain(x5/是)”。
- (3) 显式刻画并评价概念同指, 即同一个概念节点和句子中其他词对应, 在图中担当其他角色。以图1为例, “自己”这个抽象概念同时担当了概念节点“负责-01”的arg0和arg1角色, 并且分别是由“x7”和“x9”这两个词语触发。

在CAMRP2022¹评测任务中, 排名第一的SUDA-HUAWEI²采用了非自回归的方法, 通过动作预测、对齐预测、关系预测等一步步解析CAMR图。在英文AMR解析任务中, 由于其泛用性和有效性, 以自回归的方式解析AMR图已成为主流 (Bevilacqua et al., 2021a; Lam et al., 2021; Bai et al., 2022)。相比自回归方式, 非自回归不需要按照某个顺序生成节点, 能够独立地生成所有节点, 它的搜索空间较小, 解码速度也更快。相对地, 自回归方式在解码时能够充分利用上文信息, 其序列生成方式也更符合直觉。但它的搜索空间更大, 解码速度更慢, 同时面临着错误传播等问题(Zhang et al., 2019)。

¹<https://github.com/GoThereGit/Chinese-AMR/tree/main/CAMRP2022>

²<https://github.com/zsLin177/camr>

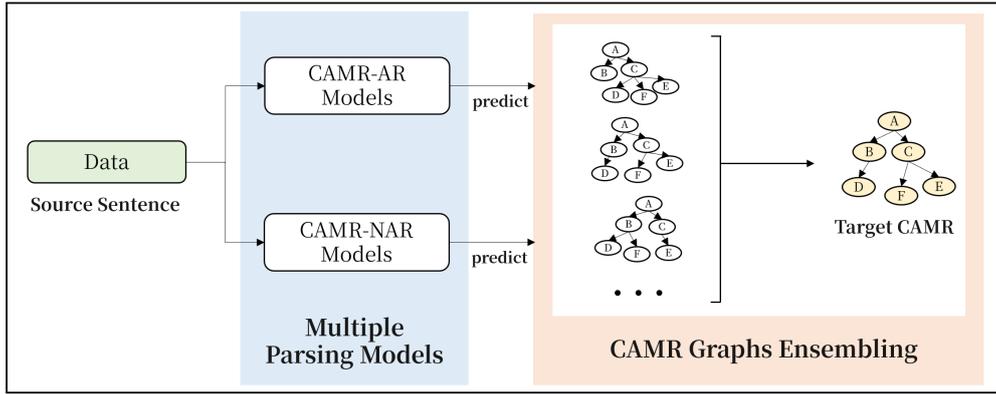


Figure 2: 参赛系统图示

如图2所示，为了综合两类模型的优点，在本次评测中，我们同时使用多个自回归模型和多个非自回归模型预测CAMR图，然后基于图融合算法(Hoang et al., 2021)将多个模型的输出结果融合起来。其中，对于自回归模型，我们借鉴Bevilacqua et al. (2021a)，将CAMR语义图线性化为序列，采用Transformer架构(Vaswani et al., 2017a)以序列生成的方式解析CAMR图，并通过后处理的方法保证图解析的有效性。对于非自回归模型，我们采用CAMR2022 SUDA-HUAWEI所设计的CAMR解析模型。此外，我们还探索了句法信息对CAMR图解析的影响。实验表明，我们的方法相较前人工作有显著的性能提升。我们的代码开源在 <https://github.com/EganGu/camr-seq2seq>。

在下文中，我们先后在第2节和第3节介绍参赛系统所采用的自回归与非自回归CAMR解析模型，随后在第4节介绍图融合算法。介绍完我们的参赛系统后，我们在第5节给出实验结果与分析。最后，我们在第6节进行总结并展望未来的研究工作。

2 基于自回归方法的CAMR解析

本节介绍了我们所设计的自回归CAMR解析模型。我们基于Transformer架构通过Seq2Seq的方式生成CAMR语义图。借鉴Bevilacqua et al. (2021b)，我们使用BART(Lewis et al., 2020)作为模型底座。

首先，为了适配生成模型，我们首先将CAMR语义图进行线性化。进一步地，我们对BART的词表进行调整，加入了CAMR图中常见的概念和关系标签。最后，对于模型可能生成的无效图（例如非连通图），我们设计了一套轻量级的启发式后处理方法，在不添加额外信息的前提下将其处理为合法的CAMR图。此外，我们还尝试使用BiLSTM融合词性和句法特征，辅助CAMR图的生成。

自回归模型的源端输入是分好词的句子³，输入Token包括词编号、汉字、英文Subword等。输出为线性化的CAMR图序列，输出Token包括AMR符号（如“:arg0”）、汉字以及英文Subword等。图1给出了源端输入句子、其对应的CAMR图以及线性化的CAMR图示例。

2.1 图线性化

数据划分	虚词出现频率	同指出现频率	数据划分	虚词出现频率	同指出现频率
Train	13.00%	1.74%	TestB	15.42%	2.48%
Dev	12.43%	1.18%	TestC	10.41%	0.30%
TestA	11.63%	1.68%			

Table 1: 实词关系中虚词对齐和概念同指的出现频率

自回归模型仅适用于序列结构，无法直接应用到CAMR的图结构上。因此，我们在预处理阶段需要将CAMR图线性化成序列。根据CAMR的特点，我们首先将虚词关系对齐、概念同指

³为方便模型进行概念对齐信息生成，我们将词的编号显式地加入源端句子。

处理成特殊关系，将其简化。然后删除CAMR图中重复的变量、空格等，在不改变句子语义的情况下获得线性化的CAMR图。

相较于英文AMR，CAMR增加了概念对齐信息以及虚词（与）关系对齐、概念同指(Li et al., 2019)。概念对齐信息使用词的编号对节点中的实例标签进行替换。如图1中的“任务”词节点，按照英文AMR规范的概念节点为“(r / 任务)”，而在CAMR中，由于该节点与原句中的第4个词对应，所以对齐后的概念节点为“(x4 / 任务)”。由于概念对齐并不违背原有的AMR规范表示，我们完整保留了概念对齐信息。但虚词关系对齐、概念同指的加入，使得CAMR的格式更加复杂多变。如图1所示，CAMR中，关系标签相较英文AMR多了一个圆括号，这是为了兼容可能存在的虚词对齐，例如“:domain(x5/是)”中的“x5/是”表示虚词“是”，该虚词通过括号包含在“:domain”关系中，完成了虚词和关系的对齐。而概念同指是指CAMR图中如果出现多个具有相同抽象意义的概念节点，默认将后续节点将其指向第一个节点，以表示它们的抽象意义相同。在图1中，第7个词语“自己”对应的概念节点应为“(x7 / 自己)”，但由于该节点与第1个节点“(x1 / 我们)”的抽象意义相同，因此它的概念节点更新为“(x7 / x1)”。

虚词关系对齐、概念同指是对实词关系的有效补充，对句子理解起重要作用(戴玉玲 et al., 2020)。如表1所示，我们统计了虚词关系对齐、概念同指在不同数据划分中的出现频率。在训练集中，它们的频率分别为13.00%和1.74%，即平均100条关系（边）中，有13条关系对齐到句子中的某个虚词、1.74条关系涉及概念同指。这表明虚词关系对齐、概念同指是一类稀疏特征（后者尤为如此）。

综上，为了统一形式、简化CAMR图，我们去掉了关系标签中的括号，将虚词关系对齐、概念同指分别处理成特殊的“:ralign”和“:coref”关系。例如在图1中，我们将虚词关系对齐“:domain(x5/是) (x4 / 任务)”转化为“:domain (x4 / 任务 :ralign (x5 / 是))”，将概念同指“:arg0() (x7 / x1)”转化为“:arg0 (x7 / 我们 :coref x1)”。统一形式后的CAMR图满足PENAMN规范(Patten, 1993)。接着，我们借鉴Bevilacqua et al. (2021b)，将CAMR图中的空格、换行符、可能存在的重复节点进行删除，获得线性化的CAMR。

2.2 词表扩充

BART采用子词级别的词表，它的分词器主要针对于自然语言，并不能很好地适配CAMR符号体系。为了解决这个问题，我们将BART的词表进行拓展，加入 1) AMR符号，如“:op”；2) CAMR特有的对齐符号；3) 在训练语料中出现次数超过5的实例词语。在原有词表中添加CAMR符号能够减少大量不必要的子词分割，从而使得编码序列更加紧凑，降低解码的时间复杂度。

2.3 句法增强

如图3所示，我们将依存句法中的依存弧、依存标签与词性标签作为句法特征融入模型进行增强。由于给定的句法特征都是词级别的信息，并不能适配子词级别的BART模型。为此，我们将句法特征按照BART分词器由词级别转化为子词（字）级别。如图4所示，词内部的子词依存结构总是以最右端为根，原词的依存结构由最右端继承。对于词性标签，我们默认子词的标签与其组成的词相同。

给定长度为 n 的子词序列 $\mathbf{x} = w_1, \dots, w_n$ ，对于 $\forall w_i \in \mathbf{x}$ ，我们可以分别获得其依存弧向量表示⁴ \mathbf{e}_i^{arc} ，依存标签表示 \mathbf{e}_i^{label} ，词性标签表示 \mathbf{e}_i^{pos} ，并将其拼接后输入到3层的BiLSTM得到它的句法表示 \mathbf{h}_i^{syn} 。

$$\mathbf{h}_i^{syn} = \text{BiLSTMs}(\mathbf{e}_i^{arc} \oplus \mathbf{e}_i^{label} \oplus \mathbf{e}_i^{pos}) \quad (1)$$

另一方面，我们通过BART编码器获得 w_i 的上下文相关表示 \mathbf{h}_i^{bart} 。为了将句法表示 \mathbf{h}_i^{syn} 融入 \mathbf{h}_i^{bart} ，我们采用Peters et al. (2018)所提出的Scalar Mixing方法。对于两种表示，我们分别定义标量参数 γ 和权重矩阵 $\alpha_{bart}/\alpha_{syn}$ ，则有

$$\mathbf{h}_i = \gamma(s_{bart}\mathbf{h}_i^{bart} + s_{syn}\mathbf{h}_i^{syn}) \quad (2)$$

其中， $s_{bart}/s_{syn} = \text{Softmax}(\alpha_{bart}/\alpha_{syn})$ 。

得到句法增强表示 \mathbf{h}_i 后，我们将其输入到BART解码器中进行CAMR图生成。

⁴受到Gehring et al. (2017)的启发，我们将依存弧视为一种特殊的位置信息，通过可训练的嵌入（Embedding）矩阵编码其向量表示。

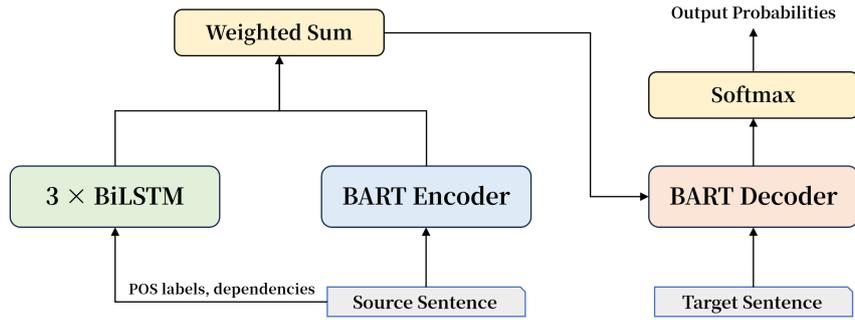


Figure 3: 句法增强的CAMR自回归模型

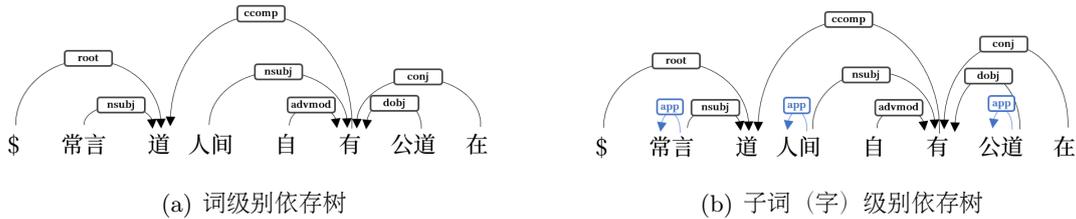


Figure 4: 依存树转换。蓝色箭头表示词内部的依存弧

2.4 后处理

为了确保模型生成的图满足CAMR结构，我们设计了启发式的后处理方法。由于生成模型在解码时没有约束，搜索空间为整个词表，可能会产生不规范的CAMR图。解码时的错误主要分为两类：1) 生成的节点无对齐，或者对齐错误；2) 生成非连通图。在后处理中，我们首先抽出生成的节点，对于实例化错误或者对齐错误的句子，我们基于原句进行重新校准。例如预测出来的概念实例“人间”错误地对齐到“x5”，如图1所示，我们在源端句子中进行搜索匹配，并将其重新对齐到“x4”。接着，对于非连通节点，将其基于“: and”关系链接到最大连通子图。

3 基于非自回归方法的CAMR解析

在本次评测中，我们也采用了去年SUDA-HUAWEI队伍设计的非自回归解析模型。该模型基于PERIN (Samuel and Straka, 2020)，先使用非自回归的方式生成AMR语义图中的节点，再基于生成的节点预测对齐信息，以及节点之间的关系信息。模型主要分为编码器和解码器两个部分，编码器部分使用Roberta (Cui et al., 2020)编码输入句子，并通过BiLSTM融入词性和依存句法信息。解码器包括动作预测、对齐预测、关系预测、属性判断和根节点预测。最后为了输出中文AMR中特有的节点对齐、虚词、共指信息，我们设计了相应的后处理操作。图5展示了我们模型的整体框架。

3.1 编码器

编码器主要由Roberta和BiLSTM两部分组成。我们利用预训练语言模型Roberta来编码输入句子 X ，来得到每个词 x_i 词级别的向量表示 \mathbf{r}_i ：

$$\mathbf{r}_i = \text{Roberta}(x_i) \quad (3)$$

因为Roberta是以子词为单位进行编码，而我们这里的词是以空格分隔得到的完整词，所以我们直接将属于同一个完整词的字词的表示相加，得到最终的词表示 \mathbf{r}_i 。

我们利用词性标签和依存句法中依存弧的标签来加入句法信息到我们的模型中，即对于每一个输入词 x_i ，我们可以分别得到它对应的词性向量表示 \mathbf{e}_i^{pos} 和句法向量表示 \mathbf{e}_i^{syn} 。最后我们将

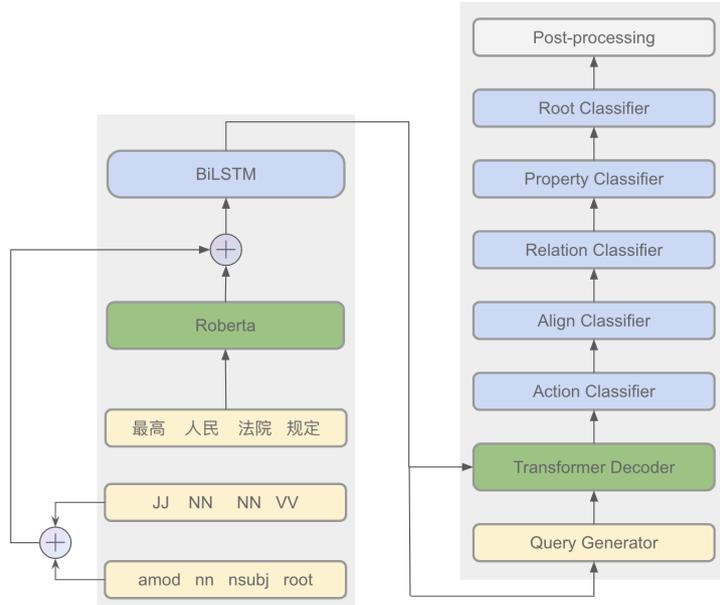


Figure 5: 基于非自回归方法的CAMR解析模型

得到的 \mathbf{r}_i 、 \mathbf{e}_i^{pos} 、 \mathbf{e}_i^{syn} 拼接后输入一层BiLSTM得到最终的编码器的输出向量 \mathbf{h}_i 。

$$\mathbf{h}_i = \text{BiLSTM}(\mathbf{r}_i \oplus \mathbf{e}_i^{pos} \oplus \mathbf{e}_i^{syn}) \quad (4)$$

3.2 解码器

在解码阶段，节点由句中的词以非自回归的方式生成，每个词最多可以生成K个节点。我们为词 x_i 生成K个query表示： $\mathbf{q}_i^1, \dots, \mathbf{q}_i^j, \dots, \mathbf{q}_i^K$ ，并为 \mathbf{q}_i^j 预测动作标签 a_i^j ，来决定由词 x_i 生成什么语义节点。具体地，我们先将编码器的输出向量 \mathbf{h}_i 输入到Query Generator中。Query Generator通过K个线性层以及tanh激活函数生成词 x_i 对应的K个query表示： $q_i^1, \dots, q_i^j, \dots, q_i^K$ ，在实验中，我们设置 $K = 3$ 。

$$\mathbf{q}_i^j = \tanh(\text{MLP}^j(\mathbf{h}_i)) \quad (5)$$

接下来，将生成的query向量和编码器的输出向量 \mathbf{h}_i 输入到三层Transformer Decoder (Vaswani et al., 2017b)中得到 \mathbf{v}_i^j 。最终解码器根据 \mathbf{v}_i^j ，使用动作分类器 (Action Classifier)，对齐分类器 (Align Classifier)，关系分类器 (Relation Classifier)，属性分类器 (Property Classifier) 和根节点分类器 (Root Classifier)，来预测 \mathbf{v}_i^j 在AMR图中作为什么语义节点，该节点对齐到哪些词，该节点与其他节点的关系，该节点是否是一个属性节点，以及该节点是否是根节点。

- 动作预测：动作分类器使用线性层打分来预测 \mathbf{v}_i^j 应该生成什么节点，目标动作包括“copy” (拷贝词 x_i 来生成节点)、“add-01” (在词 x_i 后添加“-01”，来生成节点 $x_i - 01$)、“generate [and]” (生成节点“and”)等。此外还有一个特殊动作[NULL]，表示 \mathbf{v}_i^j 不生成节点，如果 \mathbf{v}_i^j 预测出的动作是[NULL]，这就表示 \mathbf{v}_i^j 对应的query不在AMR图中，也就不参与之后的对齐等预测。在这次评测中，我们共统计有1810种动作标签。
- 对齐预测：因为一个语义节点可以对齐到多个词，所以对齐分类器使用Biaffine attention (Dozat and Manning, 2017)来判断 \mathbf{v}_i^j 与每个词是否有对齐关系，得到对齐词集合AlignSet $_i^j$ 。
- 关系预测：对于预测出的所有节点，我们分别使用两个Biaffine attention来判断一对节点之间是否存在关系，和存在什么关系。因为在中文AMR中存在着对应实词间的关系意义的虚词 (Dai et al., 2020)，我们将此类虚词和关系标签组合，形成复合标签，如“arg0+被”、“domain+是”、“location+的”等。我们为了减少标签空间，节省显存占用，只保留了那些出现次数大于1的标签，最终在我们的模型中共有1042种关系标签。

- 属性预测：在我们的模型中，我们将节点的属性（如“op1”、“op2”）也处理成了该节点的孩子节点。属性分类器需要对每个节点进行二分类来判断该节点是否是属性节点，若该节点是属性节点则将该节点从图中删去，并处理成其父亲节点的属性。
- 根节点预测：根节点分类器使用线性层打分来预测哪个节点是AMR图的根节点。

3.3 后处理

后处理部分主要分为三个部分，分别是规范化对齐、匹配虚词的对齐位置、概念同指节点的处理。

- 规范化对齐：因为在我们的模型中，节点的对齐是词语级别的，而在中文AMR中，某些节点是对齐到词语中的某几个字符的，如节点“30”，对齐到词“30余”的第一和第二个字符。我们通过后处理的方式，处理这种字符级的对齐。具体地，在由对齐预测得到对齐词集合 AlignSet_i^j 后，我们将对齐词集合中的词按照顺序拼接得到对齐词字符串，若节点字符串是对齐词字符串的子串，我们就在对齐词字符串中从左往右匹配节点字符串，找到第一个匹配位置，并返回字符级的对齐。
- 匹配虚词的对齐位置：在关系预测中，我们将虚词和关系标签组合成复合标签。在模型预测出复合标签，我们得到该关系的虚词后，我们通过匹配的方式去寻找该虚词的匹配位置。具体地，我们寻找距离该关系对应的父亲节点的对齐词最靠近的位置。特殊地，如果该句中没有匹配的位置，则返回头节点的对应词的后面一个位置。
- 概念同指节点的处理：在训练阶段，我们首先随机选取同指节点中的一个节点作为核心节点，并将其他同指节点用核心节点的编号代替，如“x1”，“x2”。在预测阶段，若解码得到的AMR图中存在标签为编号的节点，则用该编号对应的核心节点的标签代替原节点标签，并将编号相同的节点归为同指节点。

4 图融合

在得到多个自回归和多个非自回归CAMR模型后，我们采用图融合算法(Hoang et al., 2021)来融合它们所预测的CAMR图，得到最终的预测结果。模型聚合是指利用多个模型来得到相比单个模型更加稳定、精确的预测结果 (Domingos, 2000)，这项技术在各种领域的科研和竞赛中被广泛采用，并取得了亮眼的表现 (Chen and Guestrin, 2016)。我们基于Hoang et al. (2021)提出的图融合算法，并针对本次评测需要预测对齐的特点做出改进。给定 m 个AMR图 $\mathcal{S} = \{g_1, \dots, g_m\}$ ，图融合算法的核心思想是得到这 m 个图的最大公共子图，将该最大公共子图作为最终的聚合图 g_e 。

值得注意的是，由于AMR图中概念节点、边关系的情况非常复杂，很难直接对多个AMR图进行投票。Hoang et al. (2021)提出一种近似融合算法。每一次，选取一个图作为核心图，进而以核心图作为参考，其他图与该核心图进行对齐、投票，删除投票数低的节点和边后，最终产生一个聚合图。这样， m 个原始图，会产生 m 个聚合图。进而，以某种策略从 m 个聚合图中选择一个唯一的图作为最终结果。以Smatch策略为例，对于每一个聚合图，计算 m 个原始图与其之间的Smatch值，求平均后作为该聚合图的分值，最终选择分值最高的聚合图。实验中我们发现，也可以将 m 个原始图加入到候选，即从 m 个原始图加 m 个聚合图中选择最优的图作为最终结果，这样做会非常微小地提高性能，但是也会让Smatch计算量翻倍。

具体而言，Hoang et al. (2021)基于Cai and Knight (2013)提出的图近似匹配算法Smatch，提出一个启发式算法来得到多个图的聚合图 g_e 。图融合算法在每轮迭代中首先选择 S 中的一个图 g_i 作为该轮迭代的核心图，由该核心图初始化得到节点标签和边标签的投票表，接着得到图 g_i 与剩余 $m - 1$ 个图的最优匹配，根据匹配过程去更新节点标签和边标签投票表。在完成更新后删去表中得票低的标签，并选择得票最高的标签作为该节点（边）的标签，得到以 g_i 为核心图的聚合图 g_e^i 。每一轮的过程可以看作是在对该轮的核心图中的节点的标签和边标签进行投票，并保留票数最高的标签。在完成 m 轮迭代后，得到 $\{g_e^1, \dots, g_e^m\}$ 共 m 个聚合图。最后按照一定的策略从中选出最终的聚合图 g_e 。聚合图挑选策略有两种，分别是Support和Smatch分数(Cai and Knight, 2013)。Support（支持度）策略即选择在节点和边（关系）上获得最高支持度（投票数）的聚合图作为集成的目标，而Smatch策略则选择与原始图集合 S 中的图的平均Smatch分

数据划分	句子数	词语数	平均句子词数	数据划分	句子数	词语数	平均句子词数
Train	16576	386234	23.30	TestB	1999	36940	18.48
Dev	1789	41822	23.38	TestC	2000	18699	9.35
TestA	1713	39228	22.90				

Table 2: 数据集统计

	TestA			TestB			TestC		
	P	R	F ₁	P	R	F ₁	P	R	F ₁
	<i>closed</i>								
BUPT	78.40	76.44	77.41	72.09	69.68	70.87	80.96	78.62	79.77
GDUFE	76.54	78.61	77.56	75.75	61.18	67.69	82.38	73.08	77.45
WHU	78.94	74.90	76.87	72.41	67.83	70.04	80.98	76.35	78.59
Ours	81.83	78.25	80.00	75.16	70.28	72.64	83.31	79.51	81.37
	<i>open</i>								
GDUFE	75.53	75.60	75.56	69.71	67.33	68.50	78.14	77.30	77.72
SJTU	47.41	46.45	46.92	46.44	45.68	46.06	58.39	62.82	60.52
WestlakeNLP	74.40	70.24	72.26	70.42	68.63	69.52	82.49	82.73	82.61
Ours	80.82	81.79	81.30	74.39	75.03	74.71	82.13	82.11	82.12

Table 3: 评测提交结果对比

数最高的聚合图作为结果。Smatch分数通过启发式算法得到两个图的节点映射关系，从而评估两个图之间重叠部分的比例，Smatch分数越高，表明两个图越相似。

不同于英文AMR，CAMR增加了特有的概念关系对齐信息，因此我们对原本的图融合算法进行改进，对每一个节点的对齐词集合也进行投票，保留得票最高的对齐词，以提高对齐的准确率。同时，我们选择更加泛用的Smatch分数策略，选择与其他图的Smatch分数最高的聚合图作为最终结果。

5 实验

5.1 设置

我们采用官方给定的训练集和开发集上进行模型训练与挑选，然后在三个不同测试集上进行预测。数据集的具体统计数据如表2。其中，CAMRP2023新增的Test包含了来自知乎的2000句疑问短句。

对于自回归CAMR解析模型（CAMR-AR），我们采用BART-LARGE-CHINESE (Shao et al., 2021)作为模型基座，并且仅使用官方提供的数据集进行训练。模型训练30轮，学习率为 5×10^{-5} ，批次大小（Batch size）为1000 token。对于句法增强模块，我们设置句法向量表示维度为400，BiLSTM维度为512。对于没有句法标注的测试集TestB和TestC，我们基于SuPar⁵训练了一个依存句法和词性联合标注模型，来预测TestB和TestC上的句法信息。其余设置详见我们的Github仓库。对于非自回归CAMR解析模型（CAMR-NAR），其参数设置参考PERIN (Samuel and Straka, 2020)开源模型。预训练语言模型部分，我们采用了赛方推荐的Roberta (Cui et al., 2020)，以及MacBERT⁶和PERT (Cui et al., 2022)⁷三种来训练模型，获得更多样的结果。

在CAMR2023评测中，对open赛道，我们采用了CAMR-AR和CAMR-NAR两类模型，同时通过句法增强来增强模型之间的差异性，在图融合的时候得到泛化性更强的结果。对close赛道，我们仅使用微调Roberta (Cui et al., 2020)的CAMR-NAR模型。

CAMR解析的评价指标采用AlignSmatch(Xiao et al., 2022)，相较于英文AMR采用的Smatch指标，AlignSmatch将CAMR特有的概念关系对齐也作为评价对象。

	TestA			TestB			TestC		
	P	R	F ₁	P	R	F ₁	P	R	F ₁
	<i>Ablation</i>								
CAMR-AR	79.15	79.93	79.54	72.56	72.98	72.77	81.72	81.51	81.61
CAMR-AR(Syn)	79.54	80.04	79.79	72.90	72.97	72.93	81.43	81.06	81.25
	<i>Ensemble</i>								
6 × CAMR-AR	79.69	81.16	80.41	73.26	74.34	73.80	81.59	81.91	81.75
6 × CAMR-AR(Syn)	79.81	81.21	80.50	73.35	74.31	73.82	81.51	81.80	81.66
3 × CAMR-AR + 3 × CAMR-AR(Syn)	79.85	81.35	80.59	73.26	74.41	73.83	81.72	82.09	81.90
2 × CAMR-AR + 2 × CAMR-AR(Syn)	80.05	81.58	80.81	73.84	74.92	74.38	82.09	82.66	82.37
+ 2 × CAMR-NAR									

Table 4: 模型消融与图融合

Num	Strategy	TestA			TestB			TestC		
		P	R	F ₁	P	R	F ₁	P	R	F ₁
1	-	79.15	79.93	79.54	72.56	72.98	72.77	81.72	81.51	81.61
3	Support	78.56	80.82	79.68	72.42	74.04	73.22	81.17	81.79	81.48
3	Smatch	79.51	80.58	80.05	73.20	73.79	73.49	81.79	81.61	81.70
6	Support	78.63	81.49	80.04	72.07	74.72	73.37	80.87	82.16	81.51
6	Smatch	79.69	81.16	80.41	73.26	74.34	73.80	81.59	81.91	81.75
9	Support	79.33	81.40	80.35	72.76	74.56	73.65	81.18	81.91	81.55
9	Smatch	80.16	81.11	80.63	73.80	74.30	74.05	81.90	81.72	81.81

Table 5: CAMR-AR模型的图融合性能分析

5.2 结果

评测提交结果如表3所示，我们的系统在不同赛道和不同测试集上近乎都取得了最佳成绩。在closed赛道上，我们的系统相较第二名分别在TestA上提升了2.44、在TestB上提升了1.77、在TestC上提升了1.60的F₁分数；在open赛道上，我们的系统在TestA和TestB上取得了最佳成绩，但在TestC上低于WestlakeNLP队伍0.49的F₁分数。我们注意到WestlakeNLP微调了大语言模型baichuan-7B⁸来解析CAMR。这表明在简单短句组成的TestC上，大语言模型能够达到更优越的性能。但在句子长度更长的TestA和TestB上，我们相较WestlakeNLP分别提升了9.04和5.19的F₁分数，表明我们的方法能够更好地解析复杂句子的语义图。

5.3 分析

句法增强 对比表4的第1、2行，我们发现融合句法和词性信息对CAMR解析性能的影响。对比CAMR-AR，我们发现在拥有官方给定的句法和词性标注的TestA上，CAMR-AR(Syn)能够有效提升0.25的F₁分数。但在预测句法和词性的TestB、TestC上，句法增强的提升并不明显，甚至略有下降。这可能是由于预测的句法和词性带有噪声，并不能有效地帮助模型生成CAMR图。

图融合 如表4所示，我们发现通过图融合方法能够对模型性能进行稳定提升，在TestA上，使用6个CAMR-AR模型集成的性能比单个CAMR-AR模型提升了0.54的精准度(P)、1.23的召回率(R)和0.87的F₁分数。此外，实验结果表明对于图融合，不同种类模型的聚合比单一模型性能更好，在TestA上，采用三种模型的聚合性能相比采用两种模型的聚合提升了0.22的F₁分数，相比单一模型的聚合分别提升了0.40和0.31的F₁分数。说明图融合中，不同的模型带来差异化的结果能够更好的提升模型的泛化性。

如表5所示，我们进一步分析了不同的模型数量和策略对图融合性能的影响。实验结果表明，图融合所使用的模型数量越多，其性能表现也就越好。在使用Smatch策略选择聚合图的前提下，使用3个模型进行聚合在三个测试集上能够分别带来0.51、0.72和0.09的F₁分数提升，使用6个模型聚合能够进一步提升0.36、0.31和0.05的F₁分数提升。当使用更多的模型进

⁵<https://github.com/yzhangcs/parser.git>

⁶<https://huggingface.co/hfl/chinese-macbert-large>

⁷<https://huggingface.co/hfl/chinese-pert-large>

⁸<https://github.com/baichuan-inc/baichuan-7B>

行聚合时，图融合带来的提升也在渐渐减少。与使用6个模型进行聚合相比，使用9个模型只能提升0.22、0.25和0.06的 F_1 分数。这说明图融合的模型数量越多，其性能提升的边际效益将递减。除此之外，我们还探究了不同的图融合策略带来的影响，发现在所有实验样例中，采用Smatch策略的聚合性能都要高于Support策略。这一现象也符合直觉，评估两个图结构重叠部分的Smatch分数，相比汇总节点和边投票数的Support（支持度）能够全面地考虑到两个图的相似程度。

6 结语

在本次CAMR2023评测任务中，我们使用多个自回归模型和多个非自回归模型预测CAMR图，然后基于图融合算法(Hoang et al., 2021)将它们的输出结果融合起来。另外，我们还探索了句法增强和图融合策略对预测性能的影响。最终，我们在两个赛道共六个测试集上取得了五项第一名，一项第二名。

但我们的方法仍有不足，例如我们可以尝试生成大规模伪标注数据提升模型泛化能力，探索大语言模型对AMR生成模型的帮助等等。此外，对于稀疏的虚词关系对齐和概念同指，我们目前的做法是将其处理为特殊的关系标签。未来可以探究不同的CAMR图线性化方式，以更好地对这两者进行预测。

参考文献

- Xuefeng Bai, Yulong Chen, Linfeng Song, and Yue Zhang. 2021. Semantic representation for dialogue modeling. In *Proceedings of ACL-IJCNLP*.
- Xuefeng Bai, Yulong Chen, and Yue Zhang. 2022. Graph pre-training for AMR parsing and generation. In *Proceedings of ACL*.
- 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 the 7th Linguistic Annotation Workshop and Interoperability with Discourse*.
- Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021a. One spring to rule them both: Symmetric amr semantic parsing and generation without a complex pipeline. *Proceedings of the AAAI*.
- Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021b. One SPRING to rule them both: Symmetric AMR semantic parsing and generation without a complex pipeline. In *Proceedings of AAAI*.
- Shu Cai and Kevin Knight. 2013. Smatch: an evaluation metric for semantic feature structures. In *Proceedings of ACL*. Association for Computational Linguistics.
- Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of SIGKDD*.
- Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. 2021. DialogSum: A real-life scenario dialogue summarization dataset. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*.
- Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for Chinese natural language processing. In *Proceedings of EMNLP-Findings*.
- Yiming Cui, Ziqing Yang, and Ting Liu. 2022. Pert: Pre-training bert with permuted language model. *arXiv*.
- Yuling Dai, Rubing Dai, Minxuan Feng, Bin Li, and Weiguang Qu. 2020. Representation and analysis of abstract meaning of chinese function words based on relation alignment. *Journal of Chinese Information Processing*.
- Pedro Domingos. 2000. A unified bias-variance decomposition. In *Proceedings of ICML*.
- Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *Proceedings of ICLR*.

- Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional sequence to sequence learning. In *Proceedings of ICML*.
- Thanh Lam Hoang, Gabriele Picco, Yufang Hou, Young-Suk Lee, Lam Nguyen, Dzung Phan, Vanessa Lopez, and Ramon Fernandez Astudillo. 2021. Ensembling graph predictions for amr parsing. In *Advances in NeurIPS*.
- Hoang Thanh Lam, Gabriele Picco, Yufang Hou, Young-Suk Lee, Lam M. Nguyen, Dzung T. Phan, Vanessa López, and Ramon Fernandez Astudillo. 2021. Ensembling graph predictions for amr parsing. In *Advances in NeurIPS*.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of ACL*.
- Bin Li, Yuan Wen, Li Song, Weiguang Qu, and Nianwen Xue. 2019. Building a Chinese AMR bank with concept and relation alignments. *Linguistic Issues in Language Technology*.
- Kexin Liao, Logan Lebanoff, and Fei Liu. 2018. Abstract Meaning Representation for multi-document summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*.
- Terry Patten. 1993. Book reviews: Text generation and systemic-functional linguistics: Experiences from English and Japanese. *Computational Linguistics*.
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL*.
- David Samuel and Milan Straka. 2020. ÚFAL at MRP 2020: Permutation-invariant semantic parsing in PERIN. In *Proceedings of the CoNLL 2020 Shared Task: Cross-Framework Meaning Representation Parsing*.
- Yunfan Shao, Zhichao Geng, Yitao Liu, Junqi Dai, Fei Yang, Li Zhe, Hujun Bao, and Xipeng Qiu. 2021. Cpt: A pre-trained unbalanced transformer for both chinese language understanding and generation. *arXiv preprint arXiv:2109.05729*.
- Linfeng Song, Daniel Gildea, Yue Zhang, Zhiguo Wang, and Jinsong Su. 2019. Semantic Neural Machine Translation Using AMR. *TACL*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017a. Attention is all you need. In *Advances in NeurIPS*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017b. Attention is all you need. *Advances in NeurIPS*.
- Limin Xiao, Bin Li, Zhixing Xu, Kairui Huo, Minxuan Feng, Junsheng zhou, and Weiguang Qu. 2022. A novel evaluation method for chinese abstract meaning representation parsing based on alignment of concept and relation. *Journal of Chinese information processing*.
- Wen Zhang, Yang Feng, Fandong Meng, Di You, and Qun Liu. 2019. Bridging the gap between training and inference for neural machine translation. In *Proceedings of ACL*.
- 戴玉玲, 戴茹冰, 冯敏萱, 李斌, and 曲维光. 2020. 基于关系对齐的汉语虚词抽象语义表示与分析. *中文信息学报*.