文章编号：

基于转移神经网络的中文 AMR 解析*

吴泰中[1]，顾敏[1]，周俊生[+1]，曲维光[1]，李斌[2]，顾彦慧[1]

（1.南京师范大学 计算机科学与技术学院，江苏 南京 210023；

2. 南京师范大学 文学院，江苏 南京 210097）

摘要：抽象语义表示（Abstract Meaning Representation, AMR）是一种领域无关的句子语义表示方法，它将一个句子的语义抽象为一个单根有向无环图，AMR 解析旨在将句子解析为对应的 AMR 图。目前，中文 AMR 研究仍然处于起步阶段。本文结合中文 AMR 特性，采用基于转移神经网络的方法对中文 AMR 解析问题展开了实验性研究。首先，实现了一个基于转移解码方法的增量式中文 AMR 解析神经网络 baseline 系统；然后，通过引入依存路径语义关系表示学习和上下文相关词语语义表示学习，丰富了特征的学习与表示；最后，模型中应用序列化标注实现 AMR 概念识别,优化了 AMR 概念识别效果。实验结果表明，该模型在中文 AMR 解析任务中达到了 0.61 的 Smatch F1 值，明显优于 baseline 系统。

关键词：抽象语义表示；转移神经网络；概念识别

中图分类号：TP391　　**文献标识码：**A

Chinese AMR Parsing using Transition-based Neural Network

Taizhong Wu[1], Min Gu[1] , Junsheng Zhou[+1], Weiguang Qu[1], Bin Li[2], Yanhui Gu[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)

Abstract: Abstract Meaning Representation (AMR) is a domain-independent sentence semantic representation method. It abstracts the semantics of a sentence into a single directed acyclic graph. AMR parsing aims at parsing sentences into corresponding AMR graphs. At present, research on Chinese AMR is still in its infancy. In this paper, an experimental study of Chinese AMR parsing is carried out based on Chinese AMR features and the transition-based neural network. An incremental Chinese AMR parsing baseline strategy utilizing transition-based decoding method is proposed. Then, semantic representation of dependency paths and context information are utilized into the proposed model, which enriches feature representation and learning. Finally, the concept recognition in AMR parsing is conducted by applying sequence labeling. Experiments demonstrate that the proposed model outperforms the baseline and achieves Smatch F1 of 0.61 on Chinese AMR Parsing.

Key words: Abstract Meaning Representation; Transition-based Neural Network; Concept Identification

1 引言

语义是语言形式所要表达的内在含义，如何实现对自然语言句子的完整语义理解，是人工智能和自然语言处理研究领域的一个重要研究目标[1]。从某种意义上来讲，自然语言处理研究的最终目标就是在语义理解的基础上实现各类自然语言处理的应用任务。然而，由于语

* **收稿日期：**　　**定稿日期：**

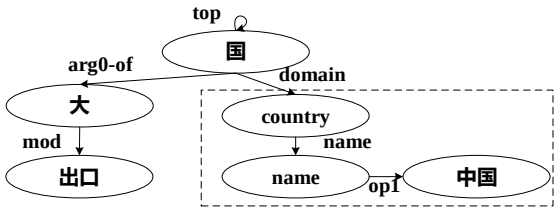
基金项目：国家自然科学基金(61472191，61772278，41571382)，福建省信息处理与智能控制重点实验室开放基金(MJUKF201705)，江苏省高校哲学社会科学研究项目(2016SJB740004)和江苏省高校自然科学研究重大项目(15KJA420001) 。

作者简介： 吴泰中（1993——），男，硕士，研究方向：自然语言处理；顾敏（1993——），女，硕士，研究方向：自然语言处理；周俊生（1972——），男，教授，研究方向：自然语言处理；李斌（1981——），男，副教授，研究方向：计算语言学；曲维光（1964——），男，教授，研究方向：自然语言处理；顾彦慧（1978——），男，副教授，研究方向：自然语言处理。

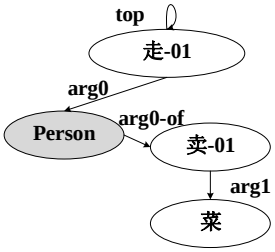
+ 通讯作者，邮箱：zhoujs@njnu.edu.cn

义的表示形式多样，传统的句子语义解析通常针对一个特定领域设计一套形式化意义表示语言[2-3]，高度的领域相关性使得语义分析的难度增大，同时对语义分析模型的泛化性提出了高要求。

针对整句的逻辑语义表示规范和语料库缺失的问题，Banarescu 等人于 2013 年提出了抽象语义表示（Abstract Meaning Representation, AMR）[4]。AMR 是一种全新的、领域无关的句子语义表示方法，它将一个句子的语义抽象为一个单根有向无环图。相比于英文 AMR 标注，中文 AMR 标注的研究起步较晚[5]。Li 等人基于英文 AMR 的框架结构，将 AMR 语义表示体系引入到汉语中，同时也充分考虑了汉语与英语的表达差异性，重点解决了 AMR 概念和词语对齐的问题，初步建立了一套汉语抽象语义的表示方法和标注规范[6]，并发布了对应的 AMR 标注语料库。由于汉语具有特有的动补结构、量词、重叠式、离合词、省略等现象，因此中文 AMR 的标注相对复杂。图 1 给出了中文句子的 AMR 图表示示例。句子中的词语与 AMR 图中的节点对齐，AMR 图中的边及边的标签表示了其语义层面的关联。AMR 图中的节点表示一个语义概念，边表示两个概念之间的语义关系。利用 AMR 图中的所有概念和关系构成的集合，能够以一种合理且一致的方式抽象表示所有句子的语义。



(a) 句子“中国是出口大国”对应 AMR 图



(b) 句子“卖菜的走了”对应 AMR 图

图 1 AMR 示例

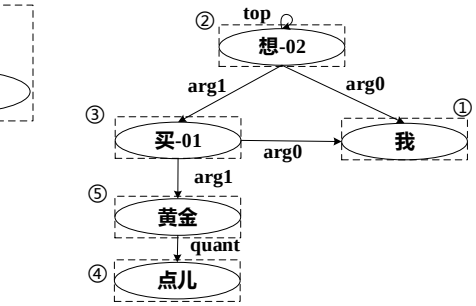


图 2 句子“我想买点儿黄金”对应的 AMR 图示例

在 AMR 标注规范和 AMR 语料库的基础上，研究人员尝试通过不同的 AMR 解析算法实现句子到 AMR 图的自动转化。AMR 标注规范的提出和 AMR 解析算法研究推动着自然语言处理上层应用的发展，例如：文本摘要、信息抽取、标题生成等[7-8]。

随着神经网络模型的应用，与基于统计学习模型的英文 AMR 解析相比，基于神经网络模型的英文 AMR 解析在模型架构、特征表示和学习、模型训练等方面都有了一定的提升。神经网络模型解决了特征稀疏表示导致的参数爆炸问题，实现了组合特征的自动学习，避免了人工特征工程。低维稠密的特征表示在向量空间上呈现出共性或相似性，为模型的训练提供了更适合的特征，同时为多任务学习提供了统一底层表示。

针对中文 AMR 解析任务，本文提出了一个基于转移神经网络的增量式中文 AMR 解析模型。转移解码算法在依存分析和英文 AMR 解析中应用广泛且取得了一定的效果[9-11]。中文 AMR 表示是一种复杂的图结构形式。因此，本文引入基于双栈的扩展 Shift/Reduce 解码

算法，以实现完整 AMR 图的生成。

此外，汉语表达具有省略、隐喻等复杂的语义特点，如何有效地进行语义表示学习是一个亟待解决的问题。因此，本文在基于转移解码方法的中文 AMR 解析 baseline 系统的基础之上，通过依存路径语义关系表示学习和上下文相关词语语义表示学习，丰富特征的学习与表示。中文 AMR 中允许补全句子中省略或者隐含的概念，且同一个词语在不同的语境下语义各不相同，其对齐的概念也不相同。如图 1（b）中，AMR 图中补全了概念节点“person”。概念之间语义关系边的构建依赖于概念识别的结果，概念识别的性能直接影响 AMR 解析模型的性能。针对概念识别和消歧问题，本文采用序列化标注思想，基于深度双向 LSTM-CRF 模型实现概念识别和消歧，以进一步提高了中文 AMR 解析模型的性能，为语义分析及上层的应用奠定基础。

2 AMR 解析

AMR 解析旨在将句子解析成对应的 AMR 图。AMR 图中的每个节点表示一个语义概念，语义概念可以是词语、PropBank framesets 或特殊的关键词。概念片段可以是单个概念节点，也可以是由多个概念节点连接组成的子图结构，如图 1 中所示，虚线框内的节点表示一个概念片段。有向边的标注表示的是两个概念之间的关系。如图 2 所示，节点“我”与节点“买-01”之间的边标记为“:arg0”关系，表示主谓关系。节点“黄金”与节点“点儿”之间的边标记为“:quant”关系，表示数量关系。

中文 AMR 具有如下特点：

（1）AMR 的基本组成单元是概念，而非词语。AMR 图中的每个节点表示一个语义概念，语义概念可对齐到句子中的词语上。

（2）在 AMR 图中，会出现一个子节点有多个父节点的情况，该子节点被称为可重入节点（Reentrant Nodes），对应的边称为重入边（Reentrant Arcs）。如图 2 中节点“我”有两个父节点“买-01”和“想-02”。

（3）AMR 图中存在非投影现象。非投影（Non-projective）现象指将图结构表示的句子中的每个词语向下垂直投影形成的线性词序列同句子的词语排列顺序不一致，即从 AMR 图上看，边之间存在交叉情况。如图 2 所示，（a）中虚线框内的节点对齐到（b）中句子中虚线框内中词语，将 AMR 图投影成线性词序列顺序时，节点“想-02”和节点“我”之间的边与节点“Root”和节点“黄金”之间的边存在交叉。

2.1 英文 AMR 解析

在英文 AMR 解析中，传统的基于统计学习模型的 AMR 解析按照解析过程与解析策略的不同，可以分为基于图的解析方法、基于转移的解析方法、基于组合范畴语法的解析方法、基于机器翻译方法。其中，基于图的解析方法和基于转移的解析方法最为常见。Flanigan 等人在 2014 年提出了第一个 AMR 解析器 JAMR[12]，这是一种基于图的解析方法，将 AMR 解析任务划分为两个子任务：概念识别和关系识别。概念识别将输入的句子中的词语或词串映射到 AMR 图中的概念片段。通过半马尔科夫模型实现概念片段的序列化标注，在概念识别输出的概念片段序列基础上，通过最大生成连通子图（Maximum Spanning Connected Subgraph, MSCG）算法从概念片段之间所有的关系中搜索具有最大得分的子图。以 CAMR[13] 为代表的基于转移的解析方法则通过预测转移动作，生成 AMR 图。

然而，传统的基于统计的模型过度依赖人工特征工程获取复杂的组合特征，且组合特征造成模型参数空间过大，造成了 AMR 解析的时间、空间效率较低。借助于神经网络模型强大的表示学习能力，Damonte 等人将神经网络模型引入 AMR 解析[11]。

根据特征提取方法的不同，基于神经网络的 AMR 解析模型可分为组合特征提取模型、

基于循环神经网络（Recurrent Neural Network，RNN）的特征提取模型和基于卷积神经网络（Convolutional Neural Network，CNN）的特征提取模型。

为了捕捉特征与模型评分之间的非线性关系，通常需要加入组合特征，即把多个基本特征组合起来作为新的特征。Damonte 等人使用数值型特征和 embedding 特征的组合[11]，通过隐藏层进行特征的连接，形成组合向量表示。

为进一步简化特征设计，Foland 等人提出的 AMR 解析模型中将 RNN 引入模型[14]。Ballesteros 等人则使用 Stack-LSTM（Stack Long Short-Term Memory）进行 AMR 状态表示学习，不借助外部资源获取特征表示[15]。Barzdins 和 Gosko 在 AMR 解析中首次使用了序列到序列（Sequence to Sequence, Seq2seq）模型，他们使用深度优先算法序列化 AMR 图，通过 LSTM 编码中间表示，再通过解码获得序列化的 AMR 表示[16]。但是由于数据稀疏问题，其精度远低于基于统计的模型。在此基础上，Konstas 等人优化了表示学习，同时利用大规模未标注数据集作为外部资源进行自训练，提高了 AMR 解析效果[17]。

对于 AMR 图而言，图中的概念节点可以对齐到句子中的词语。英文单词的词缀、词根含有丰富的语义信息。如单词“unprecedented”中，前缀“un”在 AMR 中需要标注为“:polarity”关系。虽然词向量可以在语义空间上描述一个单词的信息，但是对于词缀、词根所含有的信息可能无法清晰地表达。所以，Wang 等人使用 CNN 编码字符特征向量[18]，获取字符级别的特征，以提高 AMR 解析性能。

2.2 中文 AMR 解析

与传统的基于统计学习模型的英文 AMR 解析类似，Wang 等人提出了一种通过对依存树进行转换的方法来实现 AMR 解析[19]。该模型主要包括两个步骤：首先，使用现有的依存解析器将句子生成相应的依存句法树；然后，采用转移算法将依存树转换为 AMR 图。该模型设计了 9 种转移动作，通过对转移动作进行打分，利用贪心解码算法从转移动作集合中选择得分最高的动作对依存树进行相应的动作，从而实现从依存树到 AMR 图的转换。然而，该模型依赖于依存树作为中介，依存解析中的错误会直接传播到 AMR 解析中。

结合中、英文 AMR 解析的研究现状，针对中文 AMR 解析任务，本文提出了基于转移神经网络的中文 AMR 解析模型。

3 基于转移神经网络的中文 AMR 解析模型

3.1 转移解码算法

由于依存树与 AMR 图在结构上较为相似，受到依存分析中基于 Shift/Reduce 的依存分析算法的启发，本文采用转移解码算法，实现 AMR 解析。由于传统的基于转移解码方法（如 Arc-standard 算法和 Arc-eager 算法）主要用于生成满足投影性质的依存树结构，而对于交叉边（Crossing Arcs）和可重入边情形一般无法进行有效处理。在现有的依存图分析研究基础上[20]，本文采用一种基于双栈的扩展 Shift/Reduce 解码算法实现 AMR 解析。

基于转移的解析方法基于预先定义的转移动作集合，一步一步地从当前状态分析、预测转移动作，实现增量式的 AMR 解析。本文使用一个四元组表示当前分析状态$c = (\sigma, \sigma', \beta, A)$，其中，$\sigma$表示主栈（Primary Stack）存放已解析概念节点；次栈σ'（Secondary Stack）用于暂时存放概念节点；β表示缓存（Buffer），存放未解析的词语序列；A存放已产生的部分子图。初始化时，待解析的句子中的所有词语存放于缓存中，主栈中包含一个根节点。基于当前状态，通过预测转移动作，更新当前状态。循环操作，直到栈中只包含根节点，缓存为空时，解析过程终止，形成完整的 AMR 图表示。

扩展 Shift/Reduce 模型中共需要定义五种转移动作，具体的动作集如表 1 所示。其中前四种动作与 arc-eager 方法中的动作相似，仅针对主栈执行操作。引入的第五个动作 Mem 是

用于将主栈中的栈顶元素压入到次栈中，从而满足交叉边或多个父节点等特殊情形的处理。

表 1 动作集的形式化定义和描述

动作	描述	前提条件
LEFT_ARC(l)	$(\sigma\|\sigma_0,\sigma',\beta_0\|\beta,A)\rightarrow(\sigma\|\sigma_0,\sigma',\beta_0\|\beta,A\cup\{<\sigma_0,l,\beta_0>\})$	$\|\sigma\|\geq 1$，$\|\beta\|\geq 1$
RIGHT_ARC(l)	$(\sigma\|\sigma_0,\sigma',\beta_0\|\beta,A)\rightarrow(\sigma\|\sigma_0,\sigma',\beta_0\|\beta,A\cup\{<\beta_0,l,\sigma_0>\})$	$\|\sigma\|\geq 1$，$\|\beta\|\geq 1$
SHIFT	$(\sigma,\sigma'\|\sigma'_0,\beta,A)\rightarrow(\sigma\|\sigma_0,\sigma',\beta,A),\ then\ (\sigma,\sigma',\beta_0\|\beta,A)\rightarrow(\sigma\|root(a(\beta_0)),\sigma',\beta,A\cup E_a)\ \ where\ a(\beta_0)=(V_a,E_a)$	$\|\beta\|\geq 1$
REDUCE	$(\sigma\|\sigma_0,\sigma',\beta,A)\rightarrow(\sigma,\sigma',\beta,A)$	$\|\sigma\|\geq 2$
MEM	$(\sigma\|\sigma_0,\sigma',\beta,A)\rightarrow(\sigma,\sigma'\|\sigma'_0,\beta,A)$	$\|\sigma\|\geq 2$

其中，$a(\beta_0)=(V_a,E_a)$表示β_0对应的 AMR 子图，包含节点集合V_a和边集合E_a。

算法 1 给出了扩展 Shift/Reduce 算法中的 Oracle 算法定义。在模型训练中，按从左到右的顺序依次处理输入句子中的每一个词语，根据当前的分析状态执行相应的转移动作，直到处理完输入句子的所有词语为止，最终 Oracle 返回整个转移动作序列 T，作为训练的标准转移动作。

算法 1　Oracle 算法

输入：句子$x=\{x_0,x_1,\dots,x_n\}$和标准 AMR 图
输出：转移动作序列T
1:　$T=[]$
2:　while $index<\text{size}(x)$ do
3:　　$concept\leftarrow\text{getConcept(x[index])}$
4:　　$index\leftarrow index+1$
5:　end while
6:　$C\leftarrow C_s(concept)$
7:　if $\exists l[(\beta_0,l,\alpha_0)\epsilon A_g]$ then
8:　　$T.\text{append(LEFT\_ARC}(l))$
9:　else if $\exists l[(\alpha_0,l,\beta_0)\epsilon A_g]$ then
10:　　$T.\text{append(RIGHT\_ARC}(l))$
11:　else if $\exists i\in[1,n],l[(\alpha_0,l,\beta_i)\epsilon A_g\vee(\beta_i,l,\alpha_0)\epsilon A_g]$ then
12:　　$T.\text{append(MEM)}$
13:　else if $t\neq$ Mem and $\neg\exists i\in[1,n],\ l[(\alpha_0,l,\beta_i)\epsilon A_g\ \vee\ (\beta_i,l,\alpha_0)\epsilon A_g]$ then
14:　　$T.\text{append(REDUCE)}$
15:　else
16:　　$T.\text{append(SHIFT)}$
17:　return T

Oracle 算法首先对转移动作序列变量 T 进行初始化（第 1 行），然后循环处理输入句子中的每一个单词，获取当前词对应的概念片段（2-5 行），然后进行初始化（第 6 行）。Oracle 根据当前状态依次进行判断并将相应的转移动作添加到 T 中（7-17 行）。具体地，Oracle 首先检查主栈中的栈顶元素与队列中的队头元素之间是否可以创建 AMR 图中的某条边（左弧或右弧），如果可以创建左弧则将动作 LEFT_ARC 添加到 T 中，如果可以创建右弧则将动作 RIGHT_ARC 添加到 T 中。在当前状态不满足前面两个条件的情况下，如果主栈中的栈顶元素α_0与队列中除队头元素外的其他元素之间存在标准 AMR 图中的某条边，则将转移动作 Mem 添加到转移动作序列 T 中（第 11 行到第 12 行）。如果当前状态满足上一次执行的转移

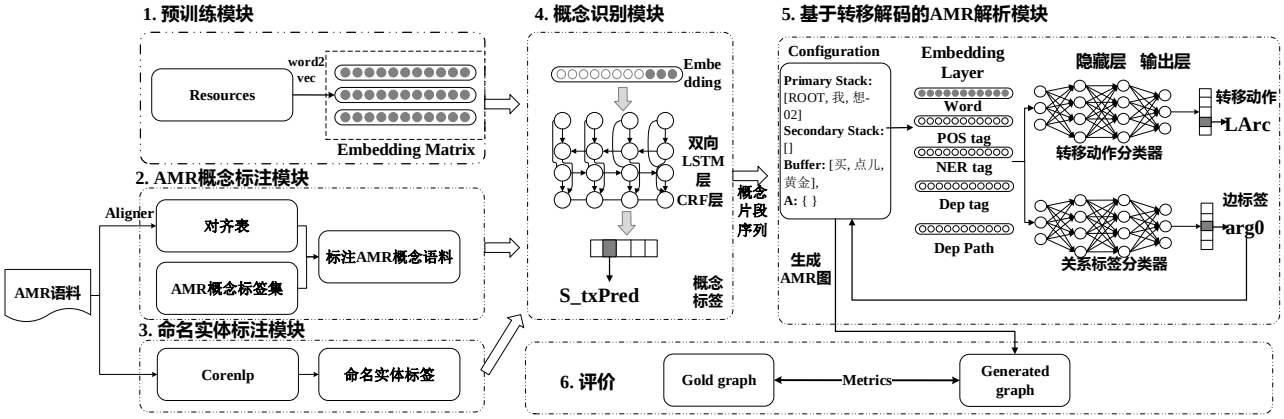


图 3 基于转移神经网络的中文 AMR 解析模型结构图

动作不为 Mem 且主栈中的栈顶元素α_0与队列中除队头元素外的其他元素之间不存在标准 AMR 图中的边，则将转移动作 Reduce 添加到转移动作序列 T 中（第 13 行到第 14 行）。如果当前状态不满足前面四个条件，则将动作 Shift 添加到转移动作序列 T 中（第 15 行到第 16 行），最后返回转移动作序列 T（第 17 行）。

3.2 整体模型结构

基于双栈的扩展 Shift/Reduce 解码算法是本文提出的中文 AMR 解析模型的基础。baseline 系统基于该解码算法，同时采用启发式搜索的方法进行概念识别，实现增量式 AMR 解析。在此基础之上，本文引入依存路径语义关系和上下文相关词语语义表示学习。同时，应用序列化标注思想实现 AMR 概念识别，由此提出了基于转移神经网络的中文 AMR 解析模型。该模型可分为 6 个部分，其模型结构如图 3 所示。其中，预训练、AMR 概念标注和命名实体标注属于预处理操作。本文基于外部语料资源，使用 Word2vec1模型预训练词向量，作用于模型的输入层。AMR 概念标注基于规则完成 AMR 语料概念标注，作为概念识别模块的训练语料。基于预先定义的概念类别，结合 AMR 对齐结果，通过规则判断，给出 AMR 语料的句子中每一个词对应的概念标签。此外，本文使用 Corenlp[2]标注 AMR 语料中的命名实体，作为概念识别模块和基于转移解码的 AMR 解析模块的特征输入。

在进行了相应的预处理后，基于序列化标注的概念识别模块利用预先定义 AMR 概念标签集和根据标签集标注的 AMR 概念训练语料，训练基于深度双向 LSTM-CRF 的深度概念识别模型。该概念识别模型中使用外部语料资源预训练词向量，输入为词向量、词性标记向量和命名实体标记向量的组合，使用深度双向 LSTM-CRF 学习整句的特征，经过多层全连接层降维和 CRF 层，计算最佳标签序列，生成对应的 AMR 概念片段。

在 AMR 概念识别结果的基础上，本文基于扩展 Shift/Reduce 算法，通过训练前馈神经网络分类器，预测转移动作和关系标签，生成 AMR 图。针对中文 AMR 解析任务，本文设计了两个分类器：（a）转移动作分类器，（b）标签分类器。根据概念识别标签序列和分类器的预测结果，选择最优的概念子图和转移动作，构造一个单根有向无环的 AMR 图。最后，对生成的 AMR 图进行相关的评价、分析。

3.2.1 概念识别

[1] https://code.google.com/p/word2vec
[2] https://stanfordnlp.github.io/CoreNLP/

AMR 对齐将句子中的单词或由多个单词构成的单词序列与 AMR 图中的概念（即节点）或概念片段（即多个节点构成的子图）对应起来。本文使用人工标注的对齐结果，获取单词到 AMR 图概念的映射关系。

根据 AMR 对齐结果，分析 AMR 概念片段的组成规则，可以发现概念片段可通过归纳操作，从而缩小概念识别时的搜索空间。基于对齐结果，本文分析了 AMR 语料中包含的概念的分布情况。对于基于 CTB 语料标注的中文 AMR 语料的训练语料部分，其所包含的概念分布情况如图 4 所示，其中，Non-predicate 表示非多个概念组成的概念片段的非谓词概念，如“菜”，“我”等；Predicate 表示非多个概念组成的概念片段的谓词概念，如“卖-01”，“想-02”；Multi-concept 表示由多个概念组成的概念片段，如“country$\xrightarrow{\text{name}}$name$\xrightarrow{\text{op1}}$ 中国”，No align 表示未对齐的概念；No align 表示未有对齐信息的概念。

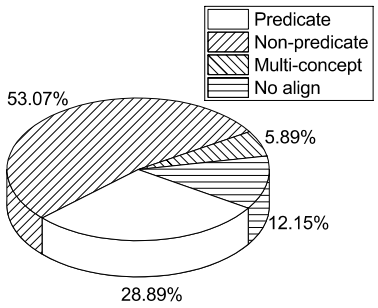


图 4 概念分布情况示意图

由图 4 可以看出，在中文 AMR 语料中，AMR 概念片段中单个谓词和非谓词所占比重较大，由多个节点构成的概念片段约占 5.89%。由于对齐信息的缺失，导致在 AMR 解析过程中，仅依赖对齐表进行启发式搜索会造成部分 AMR 概念节点的缺失。同时，由于同一个词语在不同的语境下对齐的概念各异，因此，本文采用序列化标注的思想，基于深度双向 LSTM-CRF 模型进行概念识别和消歧，其模型结构如图 5 所示。本文对待解析的词语序列进行的 AMR 概念标签标注，通过概念标签序列，生成对应的 AMR 概念子图。

AMR 概念标签集的制定依赖于 AMR 概念片段的内部结构，标记使用“BIOES+概念类别”的形式表示。本文使用“BIOES”体系标识句子中的各个词语是否为概念片段的组成并区分概念片段的边界，其中“B”标记概念片段起始的第一个词，“I”标记概念片段内部非起始并且非结尾的词，“E”标记概念片段的最后一个词，“S”标记单个词组成的概念片段，“O”标记未有对齐概念片段的词语。本文定义了 32 类类别，设计了 119 个概念标签。例如，图 1 中虚线框内的概念片段，其对应句子中的词语为“中国”，所属概念标签为“S_txNamed”，表示单个词语构成的命名实体类概念。

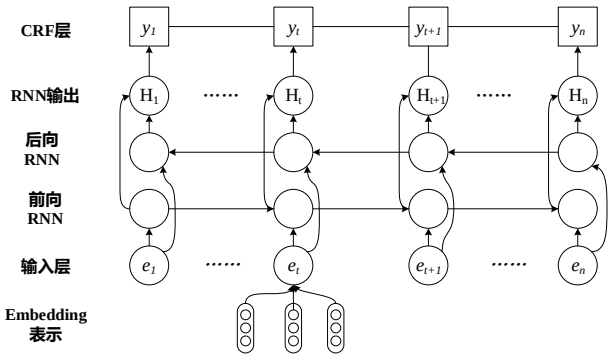


图 5 基于 LSTM-CRF 的概念识别模型

在预训练模块，本文基于 Word2vec 在大规模未标注语料库上预训练词向量，获得词向量矩阵M_w。对于输入句子序列$\mathrm{x} = (x_1, x_2, \dots, x_i, \dots, x_n)$中的第$i$个单词，通过查表操作获得相应的词向量$e_i^x$。与词性标记向量$e_i^p$和对应的命名实体标记向量$e_i^{NER}$共同组成输入特征向量$e_i$：

$$e_i = [e_i^x; e_i^p; e_i^{NER}] \tag{1}$$

其中，[;]表示向量的连接。

将组合特征向量作为双向 LSTM 层的输入。双向 LSTM 层中，前向 LSTM 的输入是输入序列的顺序序列，后向 LSTM 的输入是输入序列的逆序序列。通过双向 LSTM 层计算后，在t时刻，前向 LSTM 的输出是$\overrightarrow{H_t}$，后向 LSTM 的输出是$\overleftarrow{H_t}$。前后向的输出组成双向 LSTM 的输出H_t：

$$H_t = [\overrightarrow{H_t}; \overleftarrow{H_t}] \tag{2}$$

本文使用的是 4 个双向 LSTM 堆叠形成的深度双向 LSTM 模型，每层之间增加一层全连接层进行降维，底层双向 LSTM 的输出作为上层双向 LSTM 的输入，最后一个双向 LSTM 的输出记为$H \in \mathbb{R}^{n*k}$，n为句子长度，k表示标签数目，$H_{i,j}$表示第i个词语的第j个标签得分。

对于一个输出标签序列$\mathrm{y} = (y_1, y_2, \dots, y_i, \dots, y_n)$，定义其得分为：

$$\mathrm{s(x,y)} = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=1}^{n} H_{i, y_i} \tag{3}$$

其中，A表示转移得分矩阵。

模型训练选用交叉熵损失函数，使用 SGD 算法进行优化，同时加入 dropout 和 L2 正则化方法，防止模型过拟合。在该模块通过计算得分，选择得分最高的概念标签序列。基于获取的 AMR 概念标签序列，利用规则生成对应的 AMR 子图，用于基于转移解码的 AMR 解析模块。

3.2.2 语义关系表示学习与上下文相关词语语义表示学习

对于 AMR 图而言，仅仅获取当前栈顶元素和下一个词语之间的依存句法标签无法满足需求。在依存树中是父子关系的两个词语对齐到 AMR 图中可能距离较远。因此，本文尝试对依存句法路径进行建模，获取依存句法路径语义表示，实现长距离依存表示建模。本文将依存树中两个节点v_1和v_2的最近公共祖先（Nearest Common Ancestor）记为$\mathrm{nca}(v_1, v_2)$。v_1、v_2到其$\mathrm{nca}(v_1, v_2)$的两条路径分别为$v_1 \to \cdots \to \mathrm{nca}(v_1, v_2)$和$\mathrm{nca}(v_1, v_2) \leftarrow \cdots \leftarrow v_2$。

模型中将依存的路径分为两部分路径——词路径和关系路径。这两种路径分别用两个 LSTM(LSTM_{tok}和LSTM_{rel})进行建模。$\mathrm{nca}(v_1, v_2)$所对应的 LSTM 单元中的隐藏层输出向量就作为v_1、v_2之间语义关系的表示，如图 6 所示。

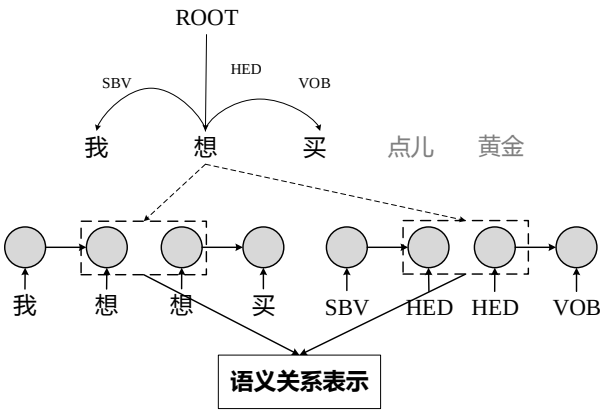


图 6 句子“我想买点儿黄金”中“我”和“买”的语义关系表示模型

词路径部分的表示定义为$R^{tok}_{(v_1,v_2)}$：

$$R^{tok}_{(v_1,v_2)} = [\vec{h}^{tok}_{\mathrm{nca}(v_1,v_2)};\ \overleftarrow{h}^{tok}_{\mathrm{nca}(v_1,v_2)}] \tag{4}$$

其中，$\vec{h}^{x}_{\mathrm{nca}(v_1,v_2)}$表示 x 路径下某一方向的 LSTM 中$\mathrm{nca}(v_1,v_2)$对应的隐藏层输出。

关系路径部分的表示定义为$R^{rel}_{(v_1,v_2)}$：

$$R^{rel}_{(v_1,v_2)} = [\vec{h}^{rel}_{\mathrm{nca}(v_1,v_2)};\ \overleftarrow{h}^{rel}_{\mathrm{nca}(v_1,v_2)}] \tag{5}$$

v_1、v_2之间语义关系的表示定义为$R_{(v_1,v_2)}$：

$$R_{(v_1,v_2)} = [R^{tok}_{(v_1,v_2)}; R^{rel}_{(v_1,v_2)}] \tag{6}$$

本文将所得到的语义关系的表示$R_{(\sigma_0,\beta_0)}$，送入到转移神经网络的输入层。其中，σ_0和β_0分别表示栈顶元素和缓存中的第一个元素。

由于中文的语言环境相对复杂，省略、隐喻等现象增大了中文 AMR 的研究难度。并且，当前的上下文语境直接影响 AMR 解析。因此，本文采用语言模型学习上下文相关的词语语义表示，得到基于语言模型的词向量表示（Embeddings from Language Model，ELM），模型结构如图 7 所示。Peters 等使用神经网络建立语言模型，从语言模型中获取词的向量表示[21]。该种基于语言模型训练的向量表示在语义角色标注、命名实体识别和情感分析等任务中都取得了较好的效果。本文采用类似的语言模型，基于 CTB5.0[3]语料和中文 AMR 标注语料，学习针对中文 AMR 解析任务的上下文相关语义表示。

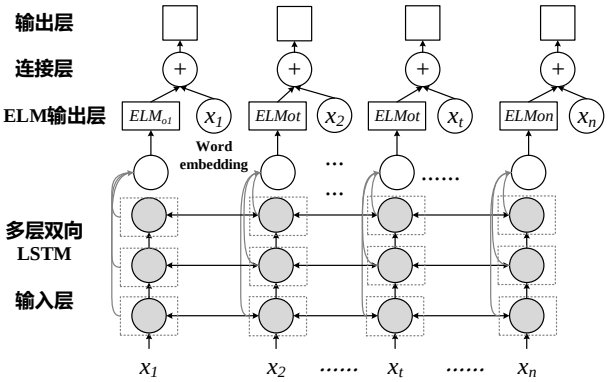


图 7 基于语言模型的上下文相关语义表示模型

在每一个位置 t，每一层 LSTM 的隐藏层输出$\vec{h}^{LM}_{t,j}$(其中$j = 1,2,\dots,L$, L 为 LSTM 的层数)都可以作为上下文相关的词表达。

$$\vec{h}^{LM}_{t,j} = \overrightarrow{LSTM}(x^{LM}_t) \tag{7}$$

其中，$\overrightarrow{LSTM}$表示抽象化的前向 LSTM 函数表示。

对于每一个词x_t，一个 L 层双向语言模型可以计算出 2L+1 个表示：

$$\begin{aligned} R^{LM}_t &= \{x^{LM}_t, \vec{h}^{LM}_{t,j}, \overleftarrow{h}^{LM}_{t,j} | j = 1,2,\dots,L\} \\ &= \{h^{LM}_{t,L} | j = 0,1,\dots,L\} \end{aligned} \tag{8}$$

[3] https://catalog.ldc.upenn.edu/ldc2005t01

其中，$h_{t,0}^{LM} = x_t^{LM}$，$h_{t,L}^{LM} = \left[\overrightarrow{h}_{t,j}^{LM}; \overleftarrow{h}_{t,j}^{LM}\right](j = 1,2,\dots,L)$。

本文将每一层 LSTM 输出的表达加权求和，得到基于语言模型的词表达：

$$ELMo_t = \gamma \sum_{j=1}^{L} s_j \, h_{t,L}^{LM} \tag{9}$$

其中，因子γ用于避免计算出的词表达过大或者过小。

基于语言模型训练得到的词表达具有很好的上下文依赖性。我们将其与原有词表达相连结，作为新的词语表示使用。

3.2.3 转移动作分类器和标签分类器

在转移解码过程中，本文基于前馈神经网络模型设计了两个分类器，分别是：转移动作分类器和标签分类器。

转移动作分类器根据当前状态，预测出下一个需要执行的动作(包括 SHIFT、LEFT_ARC、RIGHT_ARC、REDUCE 和 MEM)。分类器中定义的特征模板如表 2 所示。

当系统执行 LEFT_ARC 和 RIGHT_ARC 动作时，需要判断添加的新边的类型，并给出边的标签(如:arg0、:arg1 等)。标签分类器可以根据执行完 LEFT_ARC 和 RIGHT_ARC 动作后状态，预测边的类型。其定义的特征模板如表 3 所示。

表 2 转移动作分类器的特征模板

特征类型	特征模板
数值	σ_0、a(β_0)分别到其所在图的根节点的深度
	σ_0、a(β_0)分别到其最左叶子节点的深度
	σ_0、a(β_0)各自的孩子节点个数
	σ_0、a(β_0)各自的父节点的深度
词语	σ_0、α'_0、β_0各自对应的词的表示
	σ_0、a(β_0)各自最左父节点对应的词的表示
	σ_0、a(β_0)各自最左孩子节点对应的词的表示
	σ_0、a(β_0)各自最左孩子节点的最左孩子节点对应的词的表示
词性	σ_1、σ_0、β_0、β_1各自对应的词的词性
命名实体	σ_1、σ_0、β_0、β_1各自的命名实体类型
依存关系	$\forall i \in \{0,1\}$: σ_i和β_0之间的依存关系、β_0和σ_i之间的依存关系
	$\forall i \in \{1,2,3\}$: β_i和β_0之间的依存关系、β_0和β_i之间的依存关系
	$\forall i \in \{1,2,3\}$: σ_0和β_i之间的依存关系、β_i和σ_0之间的依存关系

表 3 标签分类器的特征模板

特征类型	特征模板
数值	σ_0、a(β_0)分别到其所在图的根节点的深度
	σ_0、a(β_0)分别到其最左叶子节点的深度
	σ_0、a(β_0)各自的孩子节点个数
	σ_0、a(β_0)各自的父节点的深度
词语	σ_0、β_0各自对应的词的表示
	σ_0、a(β_0)各自最左父节点对应的词的表示
	σ_0、a(β_0)各自最左孩子节点对应的词的表示
	σ_0、a(β_0)各自最左孩子节点的最左孩子节点对应的词的表示
词性	σ_0、β_0各自对应的词的词性
命名实体	σ_0、β_0各自的命名实体类型
依存关系	σ_0和β_0之间的依存关系、σ_0和β_0之间的依存关系

分类器使用 Adagrad 优化算法对如下目标函数进行最小化：

$$L(\theta) = -\sum_{i=1}^{n} y_i log(\hat{y_i}) + (1 - y_i) log(1 - \hat{y_i}) + \lambda ||\theta||_2^2 \quad (10)$$

其中，θ为模型参数，y_i表示i时刻，模型的期望输出值，$\hat{y_i}$表示i时刻模型的实际输出值，$\lambda||\theta||_2^2$为 L2 正则项。

4 实验分析

4.1 实验数据及评价指标

本文使用的实验语料为中文 AMR 标注语料[22]，将该数据集随机划分为训练集、验证集和测试集三个部分，其中的句子数量分布信息如表 4 所示。

表 4 中文 AMR 语料句子数量分布

语料	训练集	验证集	测试集
中文 AMR 语料	7608 句	1264 句	1278 句

本文中使用 CTB5.0 语料训练词向量，包含 507,222 个词语，词向量维数为 100。对于概念识别部分，模型 dropout 率为 0.3，表示学习和分类器部分 dropout 为 0,.3。模型 batch size 为 64，各个 LSTM 层维数均为 200，ELM 中隐藏层单元数为 100，其他均为 200。

对于实验的评测，本文采用的是目前公认的 AMR 评测方法 Smatch[23]，通过计算其正确率（P）、召回率（R）和 F1 值进行评价。

4.2 实验结果分析

为方便表示，本文将使用启发式搜索概念识别的转移解码 AMR 解析模型 baseline 记为 TC-AMR（**T**ransition based **C**hinese **AMR** Parser）；在 baseline 的基础上，将仅优化语义关系表示学习的模型记为 TC-AMR2；将优化语义关系表示学习和上下文相关词语语义表示学习的模型记为 STC-AMR（**S**emantic-enhanced **T**ransition based **C**hinese **AMR** Parser）；将基于深度双向 LSTM-CRF 模型优化概念识别的中文 AMR 解析模型记为 LC-AMR（**L**STM-**C**RF-**AMR**）。

4.2.1 学习整体解析对比

本文对比分析了中文 AMR 解析整体的 Smatch 结果，如图 8 所示。

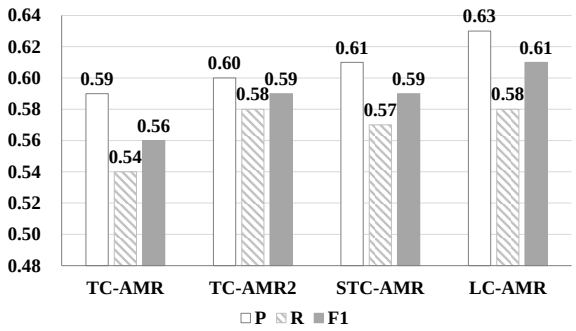


图 8 中文 AMR 整体解析结果对比图

图 8 中，基于深度双向 LSTM-CRF 模型优化概念识别的中文 AMR 解析模型 LC-AMR

在中文 AMR 自动解析任务上达到了 0.61 的 Smatch F1 值，优于 TC-AMR，TC-AMR2 和 STC-AMR 模型。与 baseline 相比，增加语义关系表示学习和上下文相关词语语义表示学习的模型 STC-AMR 整体 F1 值提高了 3%。由此可见，基于 LSTM 学习的
语义关系表示和上下文相关词语语义表示可以有效提高中文 AMR 解析性能。相比于使用启发式搜索的方法实现概念识别的 STC-AMR 模型，LC-AMR 基于序列化标注实现概念识别，提高了概念识别性能，同时使得后续的关系识别效果提高，达到了最优的中文 AMR 解析性能。本文使用的数据集是随机划分的，而 Wang 等人使用的数据集未进行此项操作[19]。在未随机划分的数据集上，Wang 等人提出的 CAMR 系统获得的 Smatch F1 值为 0.587。而本文提出的模型达到了 0.591，略高于 CAMR。

4.2.2 单项指标对比

此外，本文对比分析了 AMR 图的各个单项评价指标，共计 7 种，分别为：Unlabeled：表示不带关系标签的评价；No WSD：表示去除 Propbank suffix 的概念节点评价；Named Ent.：表示命名实体识别评价，Negations：表示否定关系（:polarity 关系边）识别评价，Reentrancies：表示重入边识别评价，Concepts：表示概念识别评价，SRL：表示":arg"关系边评价。具体结果如表 5 所示。因为 Wang 等人论文中未给出单项评价的具体数值结果，所以本文此处未与之做比较。

表 5 中文 AMR 解析单项评价结果

单项评价	TC-AMR			TC-AMR2			STC-AMR			LC-AMR		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
Unlabeled	0.66	0.60	0.63	0.68	0.65	0.66	0.68	0.64	0.66	0.70	0.65	**0.67**
No WSD	0.59	0.54	0.56	0.60	0.58	0.59	0.61	0.57	0.59	0.63	0.58	**0.61**
Named Ent.	0.73	0.63	0.67	0.72	0.67	0.70	0.73	0.65	0.69	0.81	0.72	**0.76**
Negations	0.61	0.61	0.61	0.63	0.66	**0.64**	0.62	0.65	0.63	0.58	0.59	0.58
Concepts	0.81	0.70	0.75	0.81	0.73	0.77	0.81	0.73	0.77	0.82	0.74	**0.78**
Reentrancies	0.30	0.32	0.31	0.32	0.36	0.34	0.31	0.37	0.34	0.31	0.37	**0.34**
SRL	0.56	0.46	0.50	0.60	0.50	0.54	0.58	0.50	**0.54**	0.58	0.50	0.53

注：表中同一行加粗表示的数值为模型对比结果的最高值。

从表 5 中可以看出，基于深度双向 LSTM-CRF 优化概念识别的中文 AMR 解析模型 LC-AMR 相比于其他模型在关系边的识别和概念的识别都有了提升。但是，在对于否定关系边的识别上，略低于其他模型。LC-AMR 在进行概念识别时，将否定概念节点标记为 txPolarity 概念标签，对于此类概念标签的识别准确率仅为 0.90。对于否定节点的预测错误传播到关系识别中，造成了否定关系边预测的召回率较低，仅为 0.58。

5 总结

本文针对中文 AMR 解析任务，探讨如何使用神经网络模型进行中文 AMR 解析研究。提出了一个基于扩展 Shift/Reduce 转移神经网络的中文 AMR 解析模型，并通过 LSTM 模型学习语义关系表示和上下文相关词语语义表示，增强模型的特征表示学习。在此基础上，模型引入深度双向 LSTM-CRF 模型进行概念识别和消歧。实验结果表明，本文提出的模型在中文 AMR 解析任务中达到了 0.61 的 Smatch F1 值，明显优于 baseline 系统。

在下一步的工作中，我们将扩展现有概念标签集合，以覆盖更大规模的 AMR 语料。另外，我们将研究如何进一步优化模型结构，引入带有复杂门机制的记忆神经网络模型等来提高现有模型的表达能力，进一步提高中文 AMR 解析性能。

参考文献

[1] 孙茂松，刘挺，姬东鸿. 语言计算的重要国际前沿[J]. 中文信息学报, 2014, 28(1): 1-8.

[2] Zettlemoyer L S, Collins M. Learning to Map Sentences to Logical Form: Structured Classification with Probabilistic Categorial Grammars[C]// Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence (UAI), Amsterdam, The Netherlands, AUAI Press, 2005: 658-666.

[3] Wong Y, Mooney R J. Learning for Semantic Parsing with Statistical Machine Translation[C]// Proceedings of Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL), New York, USA, Association for Computational Linguistics, 2006: 439-446.

[4] Banarescu L, Bonial C, Cai S, et al. Abstract Meaning Representation for Sembanking[C]// Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse (LAW), Sofia, Bulgaria, Association for Computer Linguistics, 2013: 178-186.

[5] 曲维光, 周俊生, 吴晓东,等. 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, 2017, 32(1):26-36.

[6] Li B, Wen Y, Qu W G, et al. Annotating the Little Prince with Chinese AMRs[C]// Proceedings of the 10th Linguistic Annotation Workshop and Interoperability with Discourse (LAW), Berlin, Germany, Association for Computer Linguistics, 2016:7-15.

[7] Liu F, Flanigan J, Thomson S, et al. Toward Abstractive Summarization Using Semantic Representations[C]// Proceedings of Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL), Denver, USA, Association of Computational Linguistics, 2015: 1077-1086.

[8] Pan X, Cassidy T, Hermjakob U, et al. Unsupervised Entity Linking with Abstract Meaning Representation[C]// Proceedings of Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL), Denver, USA, Association of Computational Linguistics, 2015: 1130-1139.

[9] Zhang X, Du Y, Sun W, et al. Transition-based parsing for deep dependency structures[J]. Computational Linguistics, 2016(3): 1-38.

[10] Dyer C, Ballesteros M, Wang L, et al. Transition-based dependency parsing with stack long short-term memory[C]// Proceedings of 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 (ACL), Beijing, China, Association of Computational Linguistics, 2015: 334-343.

[11] Damonte M, Cohen S B, Satta G. An Incremental Parser for Abstract Meaning Representation[C]// Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL), Valencia, Spain, Association for Computational Linguistics, 2017: 536-546.

[12] Flanigan J, Thomson S, Carbonell J, et al. A Discriminative Graph-based Parser for the Abstract Meaning Representation[C]// Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL), Baltimore, USA, Association of Computational Linguistics,2014: 1426-1436.

[13] Wang C, Xue N W, Pradhan S. A Transition-based Algorithm for AMR Parsing[C]// Proceedings of Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL), Denver, USA, Association of Computational Linguistics, 2015: 366-375.

[14] Foland W, Martin J H. Abstract Meaning Representation Parsing using LSTM Recurrent Neural Networks[C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL), Vancouver, Canada, Association for Computational Linguistics, 2017: 463-472.

[15] Ballesteros M and Onaizan Y. AMR Parsing using Stack-LSTMs[C]// Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP), Copenhagen, Denmark, Association for Computational Linguistics, 2017: 1269-1275.

[16] Barzdins G, Gosko D. RIGA at SemEval-2016 Task 8: Impact of Smatch Extensions and Character-Level Neural Translation on AMR Parsing Accuracy[C]// Proceedings of International Workshop on Semantic Evaluations (SemEval), San Diego, USA, Association for Computer Linguistics, 2016: 1143-1147.

[17] Konstas I, Iyer S, Yatskar M, et al. Neural AMR: Sequence-to-Sequence Models for Parsing and Generation[C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL), Vancouver, Canada, Association for Computational Linguistics, 2017: 146-157.

[18] Wang C, Xue N W. Getting the Most out of AMR Parsing[C]// Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP), Copenhagen, Denmark, Association for Computational Linguistics, 2017: 1257-1268.

[19] Wang C, Li B, Xue N W. Transition-based Chinese AMR Parsing[C]// Proceedings of Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (NAACL), New Orleans, USA, Association for Computer Linguistics, 2018: 247-252.

[20] Zhang X, Du Y, Sun W, et al. Transition-Based Parsing for Deep Dependency Structures[J]. Computational Linguistics, 2016, 42(3):353-389.

[21] Peters M E, Neumann M, Lyyer M, et al. Deep Contextualized Word Representations[C]// Proceedings of Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (NAACL), New Orleans, USA, Association for Computer Linguistics, 2018: 2227-2237.
[22] 李斌, 闻媛, 宋丽, 等. 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(6): 93-102.
[23] Cai S and Knight K. Smatch: an Evaluation Metric for Semantic Feature Structures[C]// Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL), Sofia, Bulgaria, Association for Computer Linguistics, 2013: 748-752.

作者联系方式：周俊生，江苏省南京市栖霞区文苑路 1 号南京师范大学计算机科学与技术学院，210023，18951917729，zhoujs@njnu.edu.cn