

# 知识库问答研究进展与展望

曹书林 史佳欣 侯磊 李涓子

(清华大学计算机科学与技术系 北京 100084)  
(北京信息科学与技术国家研究中心 北京 100084)

**摘要** 基于知识库的问答(Question Answering over Knowledge Base, KBQA)是问答系统的重要组成部分,要求计算机正确理解自然语言问题的语义,并从知识库中提取问题的答案。早期研究主要关注仅涉及到单个关系三元组的简单问答,近年来,随着以深度学习为代表的表示学习技术在简单问答任务的成功应用,研究重点逐渐转移到需要复杂推理能力的推理问答上。本文将对现有知识库问答的研究进展进行综述,先总结简单问答和推理问答两类任务各自的问题和挑战,然后对近年来与知识库问答相关的数据集进行多维度的分析和比较,接下来对两类任务的代表性方法进行系统性归纳介绍并分析各类方法的优缺点,最后对未来的研究方向进行展望。

**关键词** 知识库;问答系统;机器推理;可解释性;人工智能  
**中图法分类号** TP391 **DOI号** 10.11897/SP.J.1016.2023.00512

## Question Answering over Knowledge Base: An Overview

CAO Shu-Lin SHI Jia-Xin HOU Lei LI Juan-Zi

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084)  
(Beijing National Research Center for Information Science and Technology, Beijing 100084)

**Abstract** Large-scale knowledge bases, such as Wikipedia and Wikidata, have been established and updated for several years. These structured knowledge bases can benefit lots of downstream tasks about artificial intelligence, including question answering, natural language understanding, dialog system, and etc. In this article, we focus on the task of Question Answering over Knowledge Base (KBQA), which is an essential component of Question Answering systems. It requires the computer to correctly understand the semantics of natural language questions and extract the answers from the knowledge base. We divide the input questions into simple questions and complex questions according to their complexity. The research challenges and existing methods for these two kinds of questions are quite different. Simple questions require accurate linking of mentioned entity and relation, while complex questions requires more reasoning skills beyond linking. Before introducing the research methods, we first collect all KBQA datasets in recent years and give a multi-dimensional analysis. From 2011 to now, dozens of KBQA datasets have been released. Their scale becomes larger and larger and thus enables the training of large-scale neural networks. These datasets provide a necessary testbed for the development of KBQA methods. Early research mainly focused on simple questions that involve only a single triple. The common solution is to first recognize the subject entity and the relation from the input question, then

收稿日期:2020-08-04;在线发布日期:2021-12-25. 本课题得到国家重点研发计划项目(2017YFB1002101)、NSFC 通用技术基础研究联合基金重点项目(U1736204)和国家自然科学基金重点项目(61533018)资助。曹书林,博士研究生,主要研究领域为知识图谱表示和知识图谱推理。E-mail:caosl19@mails.tsinghua.edu.cn。史佳欣(共同第一作者),博士,主要研究领域为知识图谱推理、视觉推理、问答、文档摘要。E-mail:shijx12@163.com。侯磊(通信作者),博士,助理研究员,主要研究领域为知识图谱、新闻挖掘。E-mail:houlel@tsinghua.edu.cn。李涓子,博士,教授,CCF 高级会员,主要研究领域为语义 Web、社会网络挖掘。

match them with the items in knowledge base, and finally capture the answer by querying all triples. In recent years, with the rapid development of deep learning and representation learning technology, state-of-the-art methods have achieved near-human performance on SimpleQuestions benchmark, demonstrating that simple questions have been nearly tackled. As a result, the focus of researchers has gradually shifted from simple questions to complex questions. Complex questions require complex reasoning skills and are more challenging. They usually contain multiple entities, multiple relations, set operations, attribute comparison, and etc. To answer these questions, only recognizing the mentioned entities and relations is not enough. Instead, the machine must understand the logical process of the input question and then conduct an accurate execution. Existing methods can be roughly divided into semantic parsing based and information retrieval based. Semantic parsing methods translate the question from natural language to formal form explicitly, while information retrieval methods infer the answer via implicit neural matching. This article gives a more fine-grained classification for existing methods based on their detailed technical roadmap. We introduce the representative models of each category, and compare the strengths and shortcomings of different categories carefully. After the analysis of current KBQA methods, we discuss the future research direction. In the future, we should pay more attention to the interpretability of complex reasoning models, towards breaking the black-box of neural computing. To achieve this goal, it may be a good direction to combine the neural networks with symbolic machines. Besides, we can incorporate more knowledge sources, such as textual corpus, images, audios, and etc. By combining these knowledge with structured knowledge base, we can support a more practical and robust question answering system. We hope that the KBQA task can serve as a fertile ground to push the boundary of artificial intelligence.

**Keywords** knowledge graph; question answering system; machine reasoning; interpretability; artificial intelligence

## 1 引言

近年来,知识库越来越受到人们的关注,并在众多自然语言处理的下游任务中得到应用,如搜索引擎、问答系统、对话系统等.本文关注基于知识库的问答任务(Question Answering over Knowledge Base, KBQA).该任务以知识库和自然语言问题作为输入,要求计算机正确理解问题的语义,并从知识库中提取问题的答案.

图 1 给出了知识库问答的一个示例,上方是一个包含了若干实体和相关信息的知识库,下方是两个针对该知识库的问题.其中第一个问题较为简单,仅需要从知识库中找到对应的三元组即可回答.第二个问题则较为复杂,需要先找到特定的两个实体,再找出他们的身高属性,然后对属性的数值进行比较,才能给出最终的答案.解决复杂问题的一种可行的方案是将其解析为逻辑查询形式,如 SPARQL,

通过运行查询引擎来获得答案.

### 1.1 知识库中的知识类型

目前常用的大规模知识库包括 DBpedia<sup>[1]</sup>、Freebase<sup>[2]</sup>、YAGO<sup>[3]</sup>、WikiData<sup>[4]</sup>等.以 WikiData<sup>[4]</sup>为例,其中的知识可以分为三种类型,分别是:属性型知识(literal knowledge),记录实体的某个属性信息,表示为字符串、数字、日期等形式,如姚明的身高、体重、出生日期等;关系型知识(relational-knowledge),表示两个实体之间的语义关联,如(姚明,出生于,上海)阐述了姚明和上海间的语义关联;事实型知识(fact knowledge),也可以称为高阶知识(high-level knowledge),表示与一个属性事实或关系事实有关的知识,如(上海,人口数量,2339万)是一个属性型事实,其统计时间为2016年,对应的事实型知识就可以表示为((上海,人口数量,2339万),统计时间,2016年).一个事实可以对应多个事实型知识,如人口数量除了统计时间之外,还有统计方法、负责单位等信息.图 1 的知识库就包含了这

三种类型的知识.

### 1.2 简单问答与推理问答

自然语言问题可能针对一种或多种类型的知识,如“姚明的身高是多少”是针对属性型知识的提问,“姚明出生在哪里”是针对关系型知识的提问,“上海人口哪一年达到 2339 万”是针对事实型知识的提问.

在早期的研究中,研究者主要关注单个关系型知识,即问题中仅涉及到单个关系三元组,只要将主语和谓语正确地找出,就可以直接从知识库中找到匹配的三元组并得到答案.这一类问题被称为简单问题(SimpleQuestions),与之相对应的是复杂问题(ComplexQuestions),复杂问题在知识库中没有直接对应的三元组作为答案,往往包含多个实体和多个关系,需要处理多跳关系、集合操作、属性比较等情况.处理这些复杂问题要求计算机具备一定的推理能力,所以我们将针对复杂问题的问答任务称为推理问答,而针对简单问题的问答任务称为简单问答.图 1 中分别给出了简单问答和推理问答的例子.近年来,关于知识问答的研究关注重点渐渐从简单问答转移到更具挑战性的推理问答上来.

简单问答与推理问答虽然都是基于知识库的问答任务,但它们所用的研究方法有较大差异.在简单问答中,主要关注的是如何尽可能准确地识别问题中的实体和关系,以从知识库中匹配到正确的三元组.而在推理问答中,主要关注的是如何理解复杂问题的语义信息,将其解析为机器可理解的逻辑形式,进而让机器进行推理得到问题答案.本文分别对两类任务各自的代表性方法进行综述和分析.

### 1.3 现有综述

目前,关于知识库问答的代表性综述有文献[5-8].其中,文献[5]把知识库问答分为五个子任务,分析比较不同模型在五个子任务上的实现;文献[6]详细分析了知识库问答任务的挑战,总结当时已有的知识库问答方法;文献[7]对基于神经网络的知识库问答方法进行介绍;文献[8]关注基于表示学习的简单问答.与以往的知识库问答综述相比,本文除了简单问答之外,还对近年来较受关注的复杂问题的推理问答方法进行了详细的介绍和分析.另外针对所用的方法,本文除了神经网络和表示学习技术之外,还涵盖了多种类别的方法,对不同方法进行了深入的分析和对比.

### 1.4 本文结构

本文在第二部分分别阐述了基于知识库的简单问答和推理问答这两类任务各自存在的问题和面临的挑战.在第三部分中,我们将综述近年来与知识库问答相关的数据集,并对已有的数据集进行多维度的分析和比较.第四部分和第五部分分别对简单问答和推理问答的主流方法进行综述,并分析各种方法的优缺点.第六部分对未来的研究方向进行展望.

## 2 问题与挑战

### 2.1 简单问答

在知识库简单问答任务中,问题中往往包含一个三元组的主语和谓语,要求计算机准确地将其识别出来并链接到知识库中.其主要的难点在于:

知识库规模较大,链接难度较高.大规模的知识库往往包含百万级的实体和上千种关系,如 Wiki-data 中约有 9 千万实体和 7 千种关系.如何从如此大规模的知识库中准确选择问题的主语和谓语,是一个很有挑战性的问题.

同一指称对应知识库中的多个实体,消歧具有难度.如问题“Who started Apple”中的“Apple”是指“Apple Inc.”这个公司,而不是“apple”这种水果.

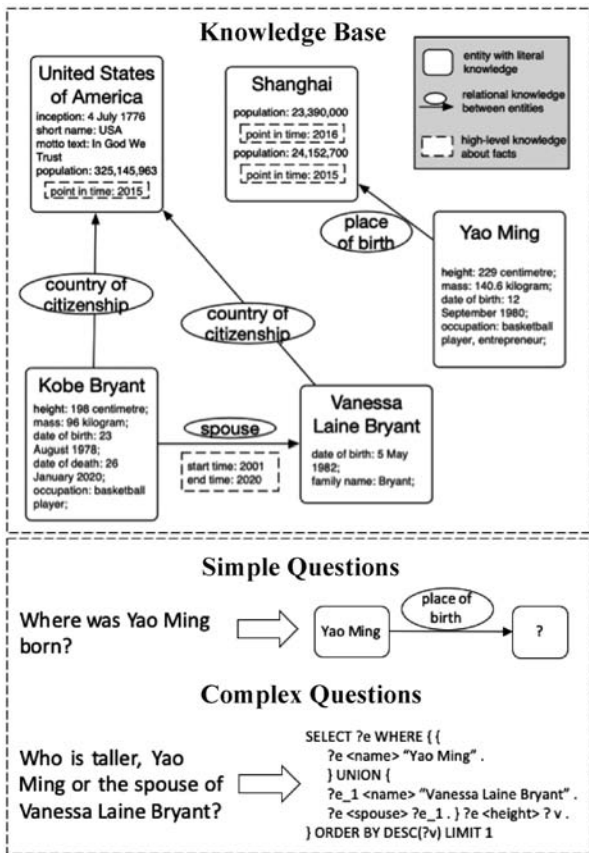


图 1 知识库问答示例

为问题中的实体指称进行消歧,从而将问题中的主语准确链接到知识库中,是一个较为困难的任务。

自然语言表达多样,谓词识别较为困难。知识库中的一种关系可以由多种自然语言描述来表示。如“Who is the founder of Apple”和“Who started Apple”都对应到知识库的“founder\_of”这个关系。准确理解问题表达的关系并将其链接到知识库中是一个较难解决的问题。

## 2.2 推理问答

在知识库推理问答任务中,问题中往往包含多个实体和多个关系,需要计算机具备多跳路径、集合操作、属性比较等推理能力,比简单问答更具挑战性。推理问答可以看做简单问答的升级版,除了以上简单问答所面临的挑战之外,推理问答还有以下难点:

数据标注困难,标注成本高昂。要研究一个任务,一般需要先有相应的数据集作为基础,以便对当前方法进行训练和评测。对于简单问答,要想得到自然语言问题,只需要将单个三元组展示给标注人员,要求标注人员针对宾语进行提问即可,标注难度低。而对于推理问答则要困难得多,需要将一个复杂的子图展示给标注人员,标注难度大大提升。一种可行的方案是先使用模板自动生成问题和答案,然后再由标注人员对问题进行转写,在不改变原意的基础上,使得问题尽量多样和自然。这种方案虽然可以获得大量复杂问题,但是往往需要人工定义大量的模板,缺乏可扩展性,问题的多样性也比较有限。

问题的语义复杂,难以正确理解。推理问答的问题中往往包含很多信息,要想让计算机准确地捕获这些语义信息是非常困难的。即便是对于人类来说,很多复杂题目也需要反复读题才能正确理解题意,复杂问题的理解对于人类都有一定困难,对于计算机来说更是如此。语义理解一直是自然语言处理领域的核心问题,即便已经有了 BERT<sup>[9]</sup> 等具有强大表达能力的预训练语言模型,语义理解仍然是没有被完全解决的问题。

需要的推理技能多,传统模型往往难以胜任。推理问答要求计算机同时处理多跳路径、集合操作、属性比较等不同的情况,传统的神经网络结构(如多层感知机,卷积神经网络等)往往难以胜任。如果使用规则和符号引擎进行推理,虽然可以解决各种情况,但是鲁棒性较低,输入稍微有所改变就可能导致结果出错(如查询的国家从 USA 变为 U. S. A.)。

推理的中间步骤多,准确率难以保证。在处理复杂问题时,一旦中间有一个步骤出错,最终的答案有

很大概率也会出错。所以要想保证问答的准确率,必须确保中间步骤尽量不能犯错,这对模型提出了更高的要求。与此同时,我们希望在机器犯错时,可以由人类定位错误所在,从而进行纠正,这就要求推理的中间步骤和结果都是可以被人理解的,可解释性也是当前研究的一大挑战。

## 3 数据集

为了研究一个任务,一般需要相应的数据集用作训练和评测。从 2011 年开始,多个知识库问答数据集相继公布,如较早的 ATIS<sup>[10]</sup>, GeoQuery<sup>[11]</sup>, QALD<sup>[12-18]</sup>, Free917<sup>[19]</sup>, SimpleQuestions<sup>[20]</sup> 和最近的 LC-QALD<sup>[21]</sup>, KQAPro<sup>[22]</sup> 等等。对于不同的知识库问答数据集,我们从问题的数量、知识的来源、知识的种类、问题是否需要推理、问题是否为自然语言、问题是否有对应的逻辑形式等六个方面进行考量,对比结果如表 1 所示。

ATIS<sup>[10]</sup> 和 GeoQuery<sup>[11]</sup> 是 20 世纪 90 年代被提出的两个数据集,都是针对专门领域构建的,前者针对航班信息知识库,后者针对地理信息知识库。它们的规模都比较小,包含大约几百到几千个问题。

Question Answering over Linked Data (QALD) 是多语言的链接数据问答 (multilingual question answering over linked data) 系统的评测竞赛活动,在 ESWC (European Semantic Web Conference) 上开展,旨在为知识库问答建立一个统一的测评基准。这个评测竞赛活动每年 100 个问题左右,从 2011 年开始,到目前共举办了 9 届。其数据来源包括 DBpedia、YAGO 和 MusicBrainz。主要任务有三类,包括基于 DBpedia 的多语种问答,基于链接数据融合不同数据集信息的问答,以及基于知识库和文本的混合问答。

在 2013 年, Cai 和 Yates<sup>[19]</sup> 基于知识库 Freebase 构建了数据集 Free917, 包含了 917 个问题和对应的逻辑表达式,覆盖 600 多个 Freebase 中的关系。

在同一时期, Berant 等人<sup>[23]</sup> 基于 Freebase 构建了标准数据集 WebQuestions。具体方法是,首先使用 Freebase,通过 Google Suggest API 爬取候选问题,得到 5810 个问题 (3778 个训练数据, 2032 个测试数据),再利 Amazon Mechanical Turk 众包标注得到答案 (问题的答案在 Freebase 中可以找到)。尽管 WebQuestions 比 Free917 的规模大,但

它有两个缺点:(1)只提供了问题和答案,而没有问题对应的逻辑表达式,因此缺乏可解释性;(2)大多

数问题都是简单问题,不涉及较为复杂的推理,难以检验模型解决复杂问题的能力。

表 1 知识库问答数据集汇总

| 数据集                                 | 问题数量      | 知识                     |      | 问题     |            |          |
|-------------------------------------|-----------|------------------------|------|--------|------------|----------|
|                                     |           | 知识库                    | 知识种类 | 是否需要推理 | 自然语言还是模板生成 | 是否包含逻辑形式 |
| ATIS <sup>[10]</sup>                | 5410      | Official Airline Guide | 多种   | 是      | 自然语言       | 是        |
| GeoQuery <sup>[11]</sup>            | 880       | Geobase                | 多种   | 是      | 自然语言       | 是        |
| QALD <sup>[12-18]</sup>             | 50-500    | DBpedia and YAGO       | 多种   | 是      | 自然语言       | 是        |
| Free917 <sup>[19]</sup>             | 917       | Freebase               | 多种   | 是      | 自然语言       | 是        |
| SimpleQuestions <sup>[20]</sup>     | 108442    | Freebase               | 仅关系型 | 否      | 自然语言       | 否        |
| LC-QALD <sup>[21]</sup>             | 5000      | DBpedia                | 仅关系型 | 是      | 自然语言       | 是        |
| KQA Pro <sup>[22]</sup>             | 117970    | Wikidata               | 多种   | 是      | 自然语言       | 是        |
| WebQuestions <sup>[23]</sup>        | 5810      | Freebase               | 多种   | 是      | 自然语言       | 否        |
| WebQuestionSP <sup>[24]</sup>       | 4737      | Freebase               | 多种   | 是      | 自然语言       | 是        |
| ComplexQuestions <sup>[25]</sup>    | 2100      | Freebase               | 多种   | 是      | 自然语言       | 否        |
| GraphQuestions <sup>[26]</sup>      | 5166      | Freebase               | 多种   | 是      | 自然语言       | 是        |
| WikiMovies <sup>[27]</sup>          | ~ 106000  | OMDb and MovieLens     | 仅关系型 | 否      | 模板生成       | 否        |
| MetaQA <sup>[28]</sup>              | 400000    | OMDb and MovieLens     | 仅关系型 | 是      | 模板生成       | 否        |
| LC-QALD2.0 <sup>[29]</sup>          | 30000     | Wikidata and DBpedia   | 多种   | 是      | 自然语言       | 是        |
| Movie Dialog <sup>[30]</sup>        | ~ 3500000 | OMDb and MovieLens     | 仅关系型 | 否      | 模板生成       | 否        |
| DBNQA <sup>[31]</sup>               | 894499    | DBpedia                | 仅关系型 | 否      | 模板生成       | 是        |
| ComplexWebQuestions <sup>[32]</sup> | 34689     | Freebase               | 多种   | 是      | 自然语言       | 是        |
| SimpleDBpediaQA <sup>[33]</sup>     | 43086     | DBpedia                | 仅关系型 | 否      | 自然语言       | 否        |

为了解决第一个问题,Yih 等人<sup>[24]</sup>构建了 WebQuestionSP,为 WebQuestions 中的 4737 个问题标注了对应的 SPARQL 查询语句。为了增强问题的复杂性和多样性,Bao 等人<sup>[25]</sup>和 Su 等人<sup>[26]</sup>在 WebQuestionSP 的基础上,进一步发布了数据集 ComplexQuestions 和 GraphQuestions,通过引入类型约束、显式或隐式的时间约束、聚合操作等对数据集 WebQuestionSP 进行增强。

上述数据集规模相对较小,只能覆盖知识库中很少一部分实体和关系,而当今深度学习技术需要大规模的训练数据,这些小规模的数据集无法满足训练需要。为此,Bordes 等人<sup>[20]</sup>构建了数据集 SimpleQuestions,包含约 100k 个问答对,每个问题对应知识库中一个三元组的头实体和关系,以尾实体作为问题的答案。SimpleQuestions 在一定程度上缓解了训练数据规模小的问题,成为简单问答任务使用范围最广的数据集。

Miller 等人<sup>[27]</sup>发布的数据集 WikiMovies 由电影领域的知识库和约 100k 个问答对组成。这个数据集的缺点是只包含简单问答对,不涉及较为复杂的推理,而且问题都由模板生成,缺乏多样性。

Zhang 等人对 WikiMovies 进行扩展<sup>[28]</sup>,构建

的数据集 MetaQA 以 WikiMovies 作为一跳(1-hop)问答数据,并增加了 21 种类型的两跳(2-hop)和 15 种类型的三跳(3-hop)问答数据,共包含 400k 多个问答对。除此之外,MetaQA 还提供了(1)机器翻译为法语,再回译为英语的问题;(2)问题对应的音频。MetaQA 规模较大,包含复杂问答对,在一定程度上缓解了 WikiMovies 的问题,然而,MetaQA 只覆盖知识库中的 9 种关系,并且问题采用模板生成,缺乏多样性。目前一些方法<sup>[34,35]</sup>已经在 MetaQA 的三跳数据集上达到了接近 95% 的准确率。

2017 年,Trivedi 等人<sup>[21]</sup>发布了数据集 LC-QALD,包含 5000 个基于 DBpedia 的复杂问题。他们首先为 DBpedia 生成结构化查询,然后利用问题模板对查询进行半自动化表达,最后利用众包将这些基于模板的问题转换为自然语言。Dubey 等人<sup>[29]</sup>采用类似的方法构建了一个规模更大、更加多样化的数据集 LC-QALD2.0,包含了 30000 个复杂问题和对应的 SPARQL 查询语句。

Saha 等人<sup>[36]</sup>和 Iyyer 等人<sup>[37]</sup>针对多轮对话问答任务发布了数据集 CSQA 和 SQA。在多轮对话任务中,回答每个问题需要考虑对话的历史信息,每个训练样例由共享同一上下文的多个问答对构成,而

单个问题的长度通常较短。

总结上述分析,目前面向知识库的问答数据集存在这样一些问题:(1)规模不够大;(2)问题由模板生成,缺乏多样性;(3)只包含问答对,没有对应的逻辑表达式等中间过程作为答案的解释;(4)知识库中知识类型较为单一,忽略了知识库中丰富的属性型知识和事实型知识。为此,Shi 等人<sup>[22]</sup>构建了 KQA Pro,如图 2 所示,KQA Pro 中的每个训练样例由五部分组成,分别是:(1)问题;(2)问句的 SPARQL 查询图;(3)表示问题推理步骤的程序 Program;(4)问题对应的十个选项;(5)最终答案。KQA Pro 包含了约 120k 条数据,问题覆盖了多跳关系、集合操作、属性比较等多种情况,是推理问答领域中质量和挑战性都比较高的一个数据集。

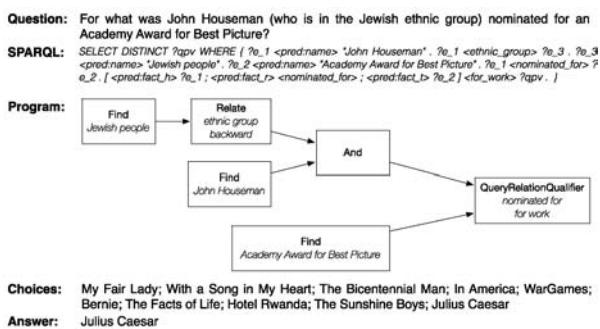


图 2 KQA Pro<sup>[22]</sup>数据集示例

## 4 简单问答研究现状

### 4.1 概述

简单问答的问题往往只涉及一个三元组,所以只要从知识库中找出与问题匹配的三元组,即可得出答案。由于知识库规模巨大,通常思路是先从知识库中选择一个子图,以减小搜索空间,再从子图中将问题对应的三元组找出,从而得出答案。这两步分别称为子图选择(Subgraph Selection)和答案三元组选择(Fact Selection),图 3 给出了简单问答的框架图。对于这两个基本步骤,研究者从不同的角度出发,提出了多种不同的解决方法。本章将会分别对子图选择和答案三元组选择的方法进行介绍和分析,并选择代表论文进行详细讲解。

### 4.2 子图选择

子图选择即选择知识库中的相关三元组作为答案三元组的候选,目的是减小答案三元组选择的搜索空间。子图选择的关键为找到问题在知识库中对应的主题实体,并进一步检索知识库中的相关三元

组,构成子图。

对于一个问题,在寻找知识库中对应的主题实体时,主要难点为:(1)多词一义:实体可能有多个指称,实体的标准名、别名、名称缩写等都可以用来指代该实体,例如 Michael Jordan、MJ 和 Jordan 都可以指代实体 Michael Jeffrey Jordan;(2)一词多义:一个指称可以指代多个实体,例如 MJ 可能指代实体 Michael Jeffrey Jordan,也可以指代实体 Michael I. Jackson。

针对这些问题,研究者提出了两类方法:一是实体链接法,先检测出问题中的实体提及,再用该提及与知识库中的实体进行链接;二是直接匹配法,直接计算问题与知识库实体的语义相似度,找出最相关的实体。

#### 4.2.1 实体链接法

这类方法的通用做法为:(1)首先利用序列标注得到问题中的实体提及;(2)其次通过字符串匹配把实体提及链接到知识库中的实体,链接到的实体称为主题实体;(3)最后抽取以主题实体为头实体的三元组集合,构成子图。

以图 3 为例,对于问题“*What body of water does St. Lawrence flow into?*”,首先通过序列标注得到实体提及“*St. Lawrence*”。然后通过字符串匹配,得到知识库中的实体“*Saint Lawrence*”和“*Saint Lawrence River*”。最后,抽取与这两个主题实体相关的三元组,得到最终的子图。

在序列标注得到实体提及后,通过字符串匹配链接到的主题实体可能较多,得到的子图仍然较大,不利于后续答案三元组的计算。为了解决这个问题,一些模型通过计算主题实体与问题的匹配程度,来进一步对主题实体进行排序和过滤。

Yin 等人<sup>[38]</sup>提出计算主题实体与问题在字符串层面的匹配程度。具体来说,主题实体  $e$  和问题  $q$  的匹配程度为

$$s_e = \alpha a + \beta b + (1 - \alpha - \beta)c.$$

$$a = \frac{|\delta|}{|q|}, b = \frac{|\delta|}{|e|}, c = \frac{p}{|q|}.$$

$$0 \leq \alpha \leq 1, 0 \leq \beta \leq 1, 0 \leq \alpha + \beta \leq 1.$$

其中,  $\delta$  是主题实体  $e$  和问题  $q$  的最长连续公共子序列,  $p$  是  $\delta$  在  $q$  中最后一个单词的位置,  $\alpha$  和  $\beta$  是超参数。也就是,最长连续公共子序列越长,越靠近问题的结尾,对应的主题实体越有可能是答案三元组中的头实体。

以上方法仅考虑实体与问题在字符串层面的匹

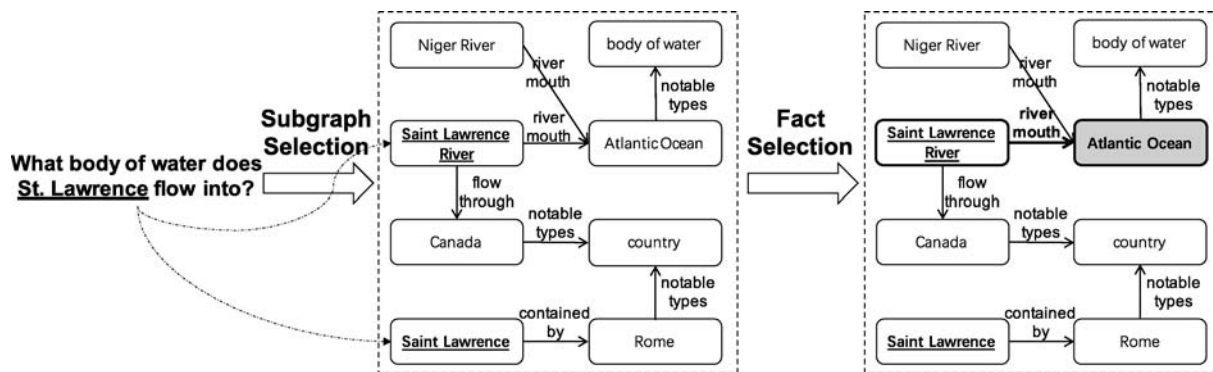


图 3 简单问答框架图

配程度,不能处理同义词等问题.例如,“New York”, “The New York City”和“Big Apple”是同一个实体,然而“Big Apple”与“*What is the population of New York?*”在字符串层面的相似度为 0.为了解决这个问题,文献[39]在文献[38]的基础上,考虑了主题实体和问题在语义层面的相似度.基本思路为利用预训练词向量,计算主题实体和问题各个单词之间的语义相似度,最终得到主题实体和问题的语义相似度.

除了在字符串和语义层面计算实体与问题的匹配程度之外,还有一些工作利用关系检测的结果对主题实体重排序.其基本假设是,如果一个实体与问题不匹配,则与之相连的关系与问题的语义也不匹配,反之亦然.这类方法的代表性工作为文献[40]和文献[41].

除了利用关系检测来帮助实体排序外,另一类方法利用问题模板来得到更加精确的主题实体.例如,文献[40]利用一些启发式规则对问题模板进行纠正,以此来过滤错误的实体提及,从而得到更加精确的主题实体.一个具体的例子为,问题“*What kind of music is on bo diddley*”最初识别的实体提及为“*on bo diddley*”,对应的问题模板为“*What kind of music is <e>*”.通过启发式规则,问题模板“*What kind of music is <e>*”被修正为“*What kind of music is on<e>*”,实体提及则相应地被修正为“*bo diddley*”.

利用搜索引擎,以问题中的实体提及为查询来检索知识库中的实体,也是实体链接的一类方法.文献[42]建立了将实体名称的  $n$ -gram 映射到实体本身的倒排索引,并计算  $n$ -gram 对于实体名称的 TF-IDF 值作为权重.例如,知识库中的实体  $e_i$  为女演员“Sarah Michelle Gellar”,其名称“Sarah Michelle Gellar”有三个 unigram(“sarah”, “michelle”, “gel-

lar”),两个 bigram(“sarah michelle”, “michelle gellar”)和一个 trigram,即实体名称本身(“Sarah Michelle Gellar”). unigram “sarah”和 bigram “michelle gellar”的索引分别为

$$I_{entity}("sarah") \Rightarrow \left\{ \begin{array}{l} node : e_i, \\ score : TF-IDF("sarah, sarah michelle gellar") \end{array} \right\}$$

$$I_{entity}("michelle gellar") \Rightarrow \left\{ \begin{array}{l} node : e_i, \\ score : TF-IDF("michelle gellar, sarah michelle gellar") \end{array} \right\}$$

对于另一个实体  $e_j$  “Sarah Jessica Parker”,也会为“sarah”建立索引:

$$I_{entity}("sarah") \Rightarrow \left\{ \begin{array}{l} node : e_j, \\ score : TF-IDF("sarah, sarah jessica parker") \end{array} \right\}$$

在建立倒排索引后,就可以通过检索实体提及的  $n$ -gram 来得到主题实体.具体来说,对于实体  $e_i$ ,依次检索实体提及本身及其  $n$ -gram ( $n \in \{3, 2, 1\}$ ),在某个  $n$ -gram 得到检索结果时停止操作.实验证明这种方法可以有效提升实体链接效果.

文献[43]和文献[44]使用 Freebase API<sup>[2]</sup>来识别问题中的实体提及并链接到知识库中的实体,这些实体及与之相邻两跳内的实体都被当作主题实体.文献[45]在相邻两跳实体的基础上,进一步将实体的属性加入到子图中,丰富了子图的信息.

#### 4.2.2 直接匹配法

除了先得到问题的实体提及,再利用实体提及进行实体链接外,另一类方法是直接利用问题匹配知识库中的主题实体.

这类方法的通常做法是通过字符串匹配,利用问题中所有可能的  $n$ -gram 来搜索知识库中的实体.代表性工作有文献[20, 46, 47].

以文献[47]为例,Lukovnikov 等人利用问题的  $n$ -gram( $n \in [1, L]$ )来检索相关实体,并设计了四条规则:(1)如果一个较短的  $n$ -gram 包含于一个较长的  $n$ -gram,且后者可以链接到知识库中的某个实体,则较短的  $n$ -gram 被舍弃;(2)如果一个实体的名称可以完全匹配某个  $n$ -gram,则这个实体将作为主题实体的候选;(3)如果对于某个  $n$ -gram,不存在与其完全匹配的实体,则选择与其编辑距离为 1 的实体作为主题实体的候选;(4)如果对于某个  $n$ -gram,有多个实体与其匹配,则以实体在知识库的出现频率为依据对实体进行筛选.在此基础上,模型进一步对候选主题实体进行排序,并利用关系预测结果对主题实体进行过滤.具体来说,模型通过将问题和实体映射到同一个向量空间,并计算向量的余弦相似度来对候选主题实体进行排序.问题由单词级别的门控循环神经网络(Gated Recurrent Unit, GRU)编码;实体表示则考虑实体本身的名称信息和在知识库中的类别信息,其名称信息和类别信息分别由字符级别和单词级别的循环神经网络(Recurrent Neural Network, RNN)编码,两者拼接得到实体表示.模型进一步对问题表示的关系进行预测,过滤掉不能作为关系头实体的候选实体.

在另外一些方法中,文献[48]直接为知识库中的三元组建立倒排索引,以问题为查询来检索相关三元组;文献[49]把问题和知识库中的实体映射到同一个向量空间,利用余弦相似度计算问题和实体的相似度,得分最高的实体即为主题实体;文献[50]利用实体的层次类型信息和与实体相连的关系信息,计算问题和实体的相似度来识别主题实体.

**小结:**子图选择的核心是根据问题从知识库中找到主题实体,主流方法可以分为实体链接法和直接匹配法两大类.实体链接法需要两步,分别是实体识别和实体链接,需要学习两个模型,依赖更多的标注信息,也容易产生错误传递的问题.直接匹配法仅需要一步,但匹配容易受到问题中噪声的影响,准确率较低.文献[38]和文献[42]的实验表明,实体链接法具有更好的性能.

### 4.3 答案三元组选择

答案三元组选择即从子图中选择与问题相匹配的事实三元组,作为最终的答案.为了选择答案三元组,研究者从关系分类和语义匹配两个思路出发,提出了不同的解决方法.关系分类即判断问题表达的关系类别  $r$ ,结合子图选择中得到的主题实体  $h$ ,选择头实体为  $h$ ,关系为  $r$  的三元组作为答案三元

组.语义匹配即把问题和候选三元组映射到同一个空间后计算问题和候选三元组的语义相似度,选择和问题语义相似度最高的三元组作为答案三元组.

#### 4.3.1 关系分类

关系分类即给定关系类型集合,判断问题所表达的关系类别,这里,给定的关系类型集合一般是训练数据集中知识库的所有关系.关系分类法的代表性方法有文献[42, 51-53],核心思路是使用循环神经网络、卷积神经网络(Convolutional Neural Network, CNN)等网络结构对问题进行编码,得到向量表示后再对问题进行分类.

关系分类方法实现较为简单,但面临两个困境:(1)规模较大的知识库通常涉及上千个关系,过大的关系集合导致分类较为困难;(2)知识库往往是动态更新的,知识库中的关系也会随之更新,而关系分类方法的关系类型集合是固定的,无法利用在已有关系数据集上训练好的模型预测新出现的关系.

#### 4.3.2 语义匹配

语义匹配法的基本思路是计算问题和候选三元组的语义相似度,对候选三元组进行排序,选择相似度最高的作为答案依据,在训练过程中,采用基于排序的目标函数.

**考虑三元组整体** 一类方法是把三元组作为整体,直接计算问题和整个候选三元组的语义相似度,代表性方法是文献[48, 54].Bordes 等人<sup>[54]</sup>最早利用基于词向量的表示学习技术,将问题中所有的词向量直接相加得到问题的向量表示,候选三元组的实体和关系向量相加得到三元组的向量表示.Gupta 等人<sup>[48]</sup>采用了三元组孪生混合卷积神经网络(Triplet-SiameseHybrid Convolutional Neural Network, TSH-CNN),联合抽取问题和候选三元组的特征.

**考虑三元组中的尾实体** 不同于把三元组作为整体,更常见的做法是将候选三元组中的尾实体作为候选答案实体,对候选答案实体进行排序.代表性方法有文献[43, 44, 46].

图 4 所示为 Bordes 等人<sup>[46]</sup>提出的 Subgraph 模型,该模型综合了候选答案实体的多种信息来作为排序依据,包括(1)候选实体本身的信息;(2)候选答案的路径信息,即从主题实体到候选实体的一跳或者两跳关系路径;(3)候选答案的上下文信息,即和候选答案直接相连的实体和关系.

Dong 等人<sup>[44]</sup>在文献[46]的基础上又引入了答案实体的类型信息,并且针对答案的路径信息,上下文信息和类型信息,使用三个不同参数的卷积神经



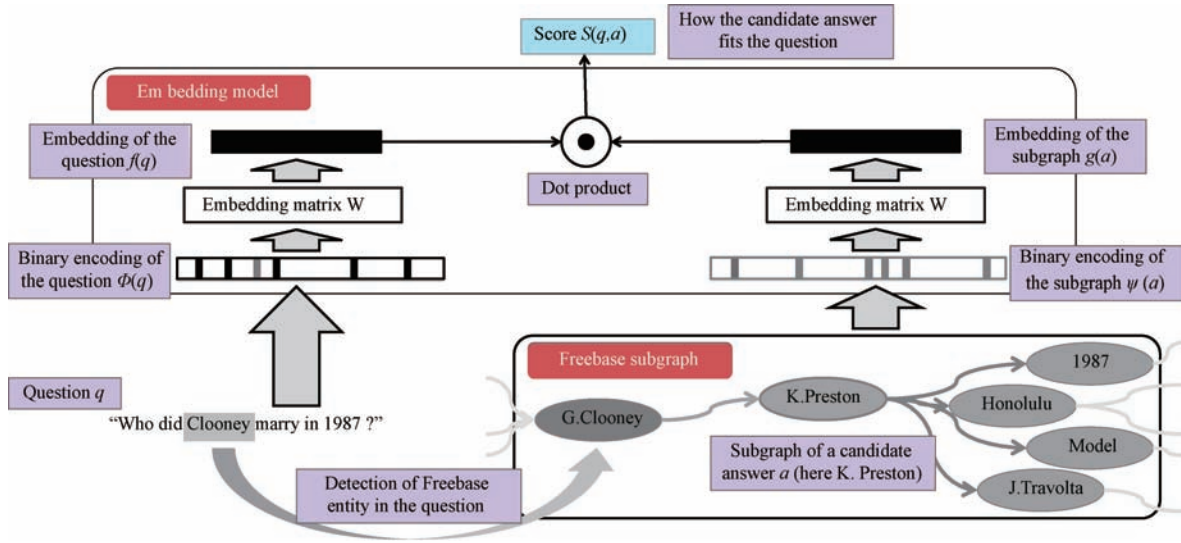


图 4 Subgraph<sup>[46]</sup>模型框架示例

网络编码得到三个不同的问题表示,分别与答案的三种信息计算语义相似度,最后问题与答案的语义相似度为这三种相似度之和.

Hao 等人<sup>[43]</sup>综合了上述四方面的实体信息,即实体本身的信息(answer entity),关系信息(answer relation),类型信息(answer type)和上下文信息(answer context).与文献[44,46]不同的是,他们认为,同一个问题的表示在关注实体不同方面时应有所不同,不同的问题在实体的四个方面也会有不同的侧重.

以问题“where is the carpathian mountain range located?”和候选答案“Slovakia”为例.如图 5 所示,当关注实体本身“Slovakia”时,对问题的侧重点是“carpathian”;而当考虑实体类型“/location/country”时,“where”又是关键词.对于一些问题,可能考虑答案的关系路径最为重要;而对于另一些问题,实体类型是最重要的特征.为了解决这个问题,他们提出了问题和答案信息的互注意力机制,包括答案到问题的注意力和问题到答案的注意力.形式化定义问题  $q$  为单词序列  $\{\tau_1, \tau_2, \dots, \tau_n\}$ ,其中  $n$  是问题的长度;答案信息为  $\{e_e, e_r, e_t, e_c\}$ ,分别对应实体本身,关系信息,类型信息和上下文信息.考虑答案信息  $e_i \in \{e_e, e_r, e_t, e_c\}$  时,问题的表示为

$$q_i = \sum_{j=1}^n \alpha_{ij} w_j,$$

$$\alpha_{ij} = \exp(\omega_{ij}) / \sum_{k=1}^n \exp(\omega_{ik}),$$

$$\omega_{ij} = f(\mathbf{W}^T [\mathbf{w}_j; \mathbf{e}_i] + \mathbf{b}).$$

其中,  $w_j$  和  $e_i$  分别为单词和答案信息的向量表示,

$\mathbf{W}$  和  $\mathbf{b}$  是待学习的参数,  $f(\cdot)$  是非线性激活函数.问题  $q$  和答案信息  $e_i$  的语义相似度为

$$S(q, e_i) = \text{cosine}(q_i, e_i).$$

最后,问题  $q$  和候选答案  $a$  的语义相似度为

$$S(q, a) = \sum_{e_i \in \{e_e, e_r, e_t, e_c\}} \beta_{e_i} S(q, e_i),$$

$$\beta_{e_i} = \exp(\omega_{e_i}) / \sum_{e_k \in \{e_e, e_r, e_t, e_c\}} \exp(\omega_{e_k}),$$

$$\omega_{e_i} = f(\mathbf{W}^T [\bar{\mathbf{q}}; \mathbf{e}_i] + \mathbf{b}),$$

$$\bar{\mathbf{q}} = \frac{1}{n} \sum_j \mathbf{w}_j.$$

考虑三元组中的头实体和关系另一类方法是分别计算问题与候选主题实体的相似度以及问题与候选关系的相似度,二者结合得到问题与候选三元组的相似度,其中候选关系通常为所有与候选主题实体相连接的关系.代表性方法有文献[47,49,55].

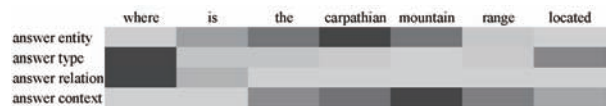


图 5 CrossATT<sup>[43]</sup>模型中实体信息对问题的注意力示例

Lukovnikov 等人<sup>[47]</sup>采用两个不同参数的循环神经网络分别学习与主题实体和关系有关的问题表示.对于候选主题实体,模型用字符级别的循环神经网络来编码实体的标签信息,用词级别的循环神经网络来编码实体的类别信息,两者拼接得到最终的实体表示.问题表示和候选主题实体表示的余弦距离即为问题与候选主题实体的语义相似度.类似地,用词级别的循环神经网络编码关系标签得到候选关系的表示,来计算问题与关系的语义相似度.

Huang 等人<sup>[55]</sup>将知识库中的实体和关系映射

到同一个向量空间中,并预测问题在这个空间中的三元组表示,与之最近邻的候选三元组就是答案三元组. 具体来说,对于知识库  $\mathcal{G}$  中的三元组  $(h, l, t)$ , 有  $e_t \approx f(e_h, p_l)$ , 其中  $e_h, e_t$  和  $p_l$  分别是头实体  $h$ , 尾实体  $t$  和关系  $l$  的向量表示.  $f(\cdot)$  是头实体和关系到尾实体的映射关系, 如知识表示学习模型 TransE<sup>[56]</sup> 中,  $e_t \approx e_h + p_l$ . 如图 6 所示, 在关系学习中, 输入问题  $q$ , 将其映射到关系向量空间中, 得到问题  $q$  的关系表示  $\hat{p}_l$ ; 同样, 采用类似的方法得到问题  $q$  的实体表示  $\hat{e}_h$ . 然后将预测出的关系表示和实体表示代入  $\hat{e}_t \approx f(\hat{e}_h, \hat{p}_l)$  中, 从而得到问题的三元组表示  $(\hat{e}_h, \hat{p}_l, \hat{e}_t)$ . 最后, 在知识库中与  $(\hat{e}_h, \hat{p}_l, \hat{e}_t)$  距离最近的三元组就是答案三

元组.

He 和 Golub<sup>[49]</sup> 则提出了编码解码 (Encoder-Decoder) 框架, 如图 7 所示, 采用基于字符向量的长短时记忆网络 (Long-Short Term Memory Network, LSTM) 对问题进行编码, 再采用基于注意力机制的 LSTM 进行解码, 依次生成主题实体和关系. 在生成主题实体时, 模型采用基于字符向量的卷积神经网络对知识库中的实体进行编码, 并通过计算问题隐向量与实体向量的余弦相似度, 选择相似度最高的实体作为生成的主题实体. 在生成关系时, 将前一步生成的实体作为解码器的输入, 得到新的问题隐向量, 同样计算问题隐向量与候选关系的语义相似度.

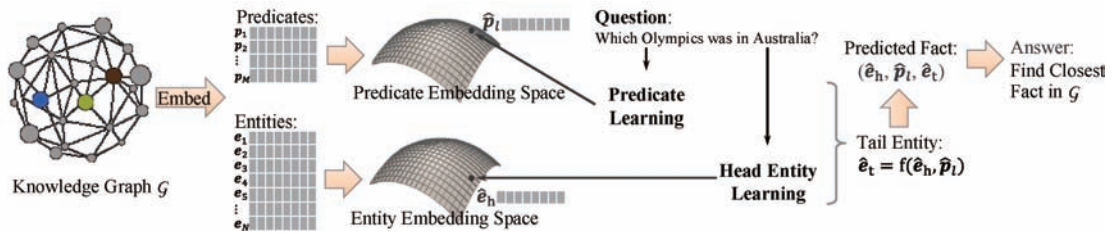


图 6 Huang 等人<sup>[55]</sup> 基于表示学习的知识库问答模型

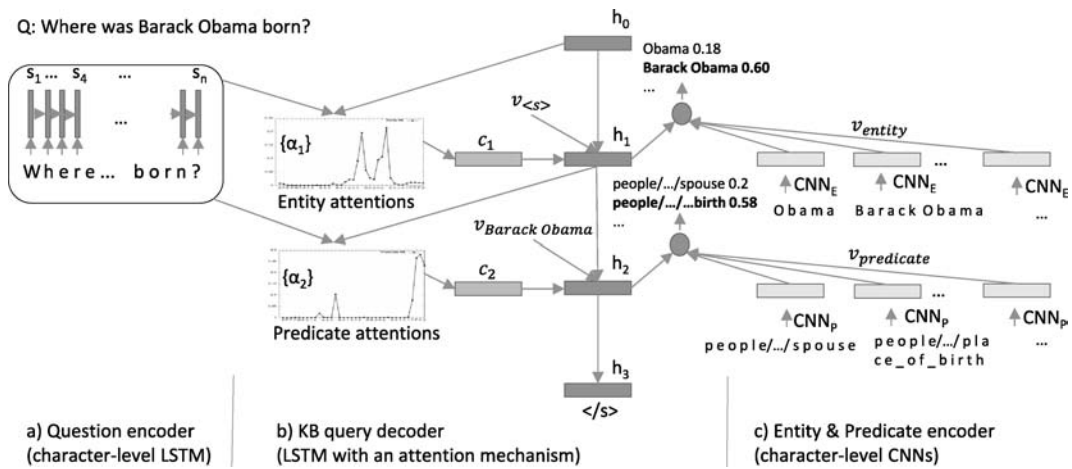


图 7 He 和 Golub<sup>[49]</sup> 的编码解码模型示例

在计算问题与候选关系的语义相似度时, 除了把原问题作为输入, 另一种方法是把问题模板作为输入. 问题模板是通过把问题中的实体提及替换为特殊符号 “<e>” 得到的问题变形. 例如, 问题 “What body of water does St. Lawrence flow into” 的实体提及为 “St. Lawrence”, 则问题模板为 “What body of water does <e> flow into”; “What body of water does Yellow River flow into” 的实体提及为 “Yellow River”, 其问题模板同样为 “What body of water does <e> flow into”. 比较原问题和

问题模板, 后者忽略主题实体, 能够更集中地体现问题的关系, 有利于进行关系匹配. 这类方法的代表工作有文献[38, 40, 57-59].

文献[38]使用基于字符向量的卷积神经网络计算问题和候选实体的语义相似度, 同时使用基于词向量的卷积神经网络计算问题模板与候选关系之间的相似度. 该模型在对问题模板进行卷积操作后, 采用了基于注意力机制的最大池化, 旨在更加关注与候选关系一致的问题关键词.

Yu 等人<sup>[57]</sup> 利用残差学习 (Residual Learning)

增强的层次循环神经网络(Hierarchical Recurrent Neural Network)对问题模板和候选关系进行匹配,并且从关系整体和词序列两个层级对关系进行编码.如图8所示,“starring roles series”在关系整体层面包含两个关系“tv.tv\_actor.starring\_roles”和“tv.regular\_tv\_appearance.series”,在词序列层面为“starring”、“roles”和“series”组成的单词序列.关系整体层级的编码有利于学习关系的整体信息,词序列层级的编码有利于解决未登录关系(即在测试集中出现而未在训练集中出现的关系).

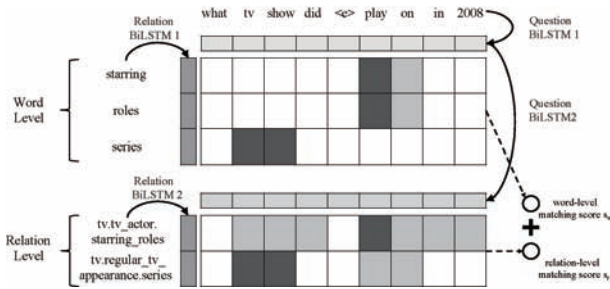


图8 词序列层级和关系整体层级编码示例

文献[40]基于字符向量的循环神经网络对问题中的实体提及和候选主题实体进行编码,计算其语义相似度;对于候选关系,与文献[57]相似,模型从多个角度进行编码,包括把关系作为一个整体单词,把关系名称作为单词序列.另外,模型还考虑了知识库中关系所连头实体的类型.

在其他模型中,Yu等人<sup>[58]</sup>在为关系编码时,考虑了知识库中关系对应尾实体的类型信息;Zhang等人<sup>[59]</sup>使用基于注意力机制的门循环神经网络来计算语义相似度,使用卷积神经网络来计算字符层面的相似度,并通过为词向量加入扰动的对抗训练方式增强模型的鲁棒性.

**小结:**答案三元组选择有关系分类和语义匹配两类方法.关系分类法在特定领域或特定数据集上表现较好,但是只能预测固定的关系集合,无法处理新出现的关系,在实际应用中比较受限.语义匹配法通过计算问题和候选三元组的语义相似度进行选择,计算相似度有多种不同的实现思路,也可以处理首次遇到的实体和关系,适用范围较广,近年来的研究工作也主要集中在这类方法上.

## 4.4 简单问答的挑战

### 4.4.1 错误传播问题

在子图选择和答案三元组选择的级联机制中,前一步的效果会极大影响后一步的效果,即如果子图选择的结果没有包括答案三元组,则答案三元组

选择不可能成功;而如果子图选择的结果包含太多答案三元组以外的冗余信息,则会给答案三元组选择带来噪音和额外的计算开销.

为了缓解这种错误传播(Error Propagation)问题,文献[40,60,61]提出了不同的解决方案.其中,文献[40]设计启发式规则,通过统计信息来查找异常的问题模板,并对问题模板和相应的实体提及进行修正.文献[60]提出了查询检查机制,即通过把关系名称信息引入问题,得到问题的抽象表示,模型考察答案三元组的关系是否可以得到最好的问题抽象表示.如图9所示,左侧关系“activism.activist.area\_of\_activism”得到的问题抽象表示“what area of activism did activist fight for”比右侧关系“people.person.profession”得到的问题抽象表示“what profession did person fight for”表达更加连贯自然,因此,模型认为,包含关系“activism.activist.area\_of\_activism”的三元组更有可能是答案三元组.文献[61]在现有问答框架的基础上,加入了负责对预测关系的可靠性进行评价检验的验证机制,即判断预测关系是否正确的二分类器.

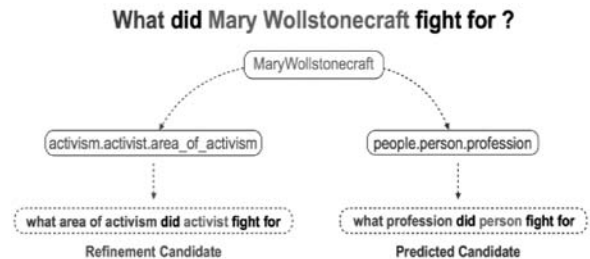


图9 查询检查示例

### 4.4.2 未登录实体和关系问题

简单问答的另一个挑战是未登录实体和关系问题,即在实际应用过程中难以处理训练过程中没有遇到的实体和关系.其解决思路总结为:(1)借助预训练的实体和关系向量,如知识表示学习将实体和关系映射到同一个向量空间,得到的实体和关系向量编码了知识库中事实三元组的先验知识,有助于理解在测试集中首次出现的实体或关系,代表性工作有文献[43,55];(2)对实体或关系采用基于字符向量或词向量的文本编码器,对实体名称或关系名称进行编码,充分利用其名称信息,代表性工作有文献[47,49];(3)在答案三元组选择时,采用基于排序的语义匹配法,而不是给定关系集合的分类法,代表性工作有文献[38,58,59]等.

## 4.5 小结

简单问答的一般思路为首先进行子图选择,再

进行答案三元组选择. 子图选择有实体链接和直接匹配两类方法, 其中实体链接法的准确率较高, 但依赖额外的标注数据, 直接匹配法则相反, 准确率较低, 但不需要实体提及的标注, 更容易在实际中使用. 答案三元组选择有关系分类和语义匹配两类方法, 其中关系分类法比较简单, 适用于给定关系集合的限定领域, 无法处理新出现的关系, 而语义匹配法的适用范围则要广得多, 实现思路也比较多样, 可以综合考虑多种信息, 也可以处理未登录的实体和关系, 近年来受到的关注比较多.

简单问答的关键在于准确识别问题中的实体和关系. 随着近年来深度学习的发展, 实体识别等任务取得了较大进展, 简单问答也达到了接近人类表现的性能.

表 2 中列出了简单问答的代表性方法在常用数据集 SimpleQuestions 上的实验结果. Petrochuket 等人<sup>[53]</sup>提出, SimpleQuestions 数据集由于歧义等问题, 其准确率上限为 83.4%, 他们提出的一种简单的基于 BiLSTM 和条件随机场的基线方法就可以达到 78.1% 的准确率. Zhao 等人<sup>[39]</sup>提出的方法达到了 85.4% 的准确率, 已经达到甚至超过了人类的表现. Ture 和 Jovic<sup>[42]</sup>提出的 RNN-QA 在 SimpleQuestions 上达到 88.3%, 是目前准确率最高的模型之一.

表 2 简单问答代表性方法的实验结果对比

|                                   | SimpleQuestions<br>(Accuracy) |
|-----------------------------------|-------------------------------|
| Golub et al. <sup>[49]</sup>      | 70.9%                         |
| Lukovnikov et al. <sup>[47]</sup> | 71.3%                         |
| BiGRU <sup>[51]</sup>             | 74.9%                         |
| KEQA <sup>[55]</sup>              | 75.4%                         |
| CFO <sup>[52]</sup>               | 75.7%                         |
| AMPCNN <sup>[38]</sup>            | 76.4%                         |
| BiLSTM-CRF <sup>[53]</sup>        | 78.1%                         |
| Hier-Res-BiLSTM <sup>[57]</sup>   | 78.7%                         |
| PatternRevising <sup>[40]</sup>   | 80.2%                         |
| Chung et al. <sup>[39]</sup>      | 85.4%                         |
| RNN-QA <sup>[42]</sup>            | 88.3%                         |

## 5 复杂问题的推理问答研究现状

### 5.1 概述

为了让计算机具备推理和解决复杂问题的能力, 研究者从不同角度出發, 提出了多种不同的解决方法. 已有的针对知识库推理问答的研究方法主要分为以下六个类别:

(1) 有监督的语义解析法. 该类方法将文本形式的问题解析为结构化的形式, 又称为逻辑形式(logical form), 然后使用规则或引擎查询答案. 在知识库问答领域, 最常用的逻辑形式为 SPARQL 语句. 该类方法主要关注如何利用标注数据来学习从复杂问句到逻辑形式的问题解析器;

(2) 弱监督的语义解析法. 该类方法针对逻辑形式标注昂贵、难以获取的问题, 仅根据问题与答案来学习解析器. 相对于有监督的方法, 该类方法主要关注如何搜索合理的逻辑形式, 从而为解析器提供学习信号. 由于搜索空间巨大, 该类方法往往使用较为简单的逻辑形式, 并通过语法约束等方式帮助学习过程收敛;

(3) 关系路径法. 该类方法专门针对多跳问题, 即问题中涉及到一个实体和多个关系, 如“Where was the father of Sylvia Brett’s other half born?”. 该类方法通常先找到问题中的主题实体, 然后从主题实体出发预测关系路径, 沿着路径找到答案实体;

(4) 键值记忆网络. 该类方法是对记忆网络的改进, 引入键值对存储知识库中的三元组, 通过迭代地访问键值对读取知识, 并进行推理;

(5) 可导路径链. 该类方法用权重向量表示实体集合, 用矩阵乘法表示实体集合沿特定关系的转移, 以可导的方式表示集合的交、并等操作;

(6) 图神经网络. 该类方法使用图神经网络, 迭代更新实体的特征向量, 融合图中信息, 隐式地进行推理.

图 10 给出了知识库推理问答各类方法的框架图示例. 下面将会对这些方法进行详细的介绍和分析, 并对每种方法中代表论文进行阐述.

### 5.2 有监督的语义解析法

该类方法通常利用人工标注的训练语料来学习如何将问题从非结构的文本形式解析为结构化的逻辑形式, 语料中包含了成对的自然语言问题和对应的结构化表示. 常用的逻辑形式包括 SQL、SPARQL、lambda-calculus、Prolog 等.

早期的方法<sup>[62-64]</sup>依靠人工构建的词表、语法规则、句法模板、特征工程等方式来构建解析器. 这些方法往往只能针对特定领域和特定逻辑形式, 缺乏可扩展性.

随着深度学习的发展, 基于神经网络的方法逐渐流行, 并在诸多自然语言处理任务中取得巨大进展, 如机器翻译<sup>[65]</sup>. 神经网络方法也逐渐应用到语义解析任务中.

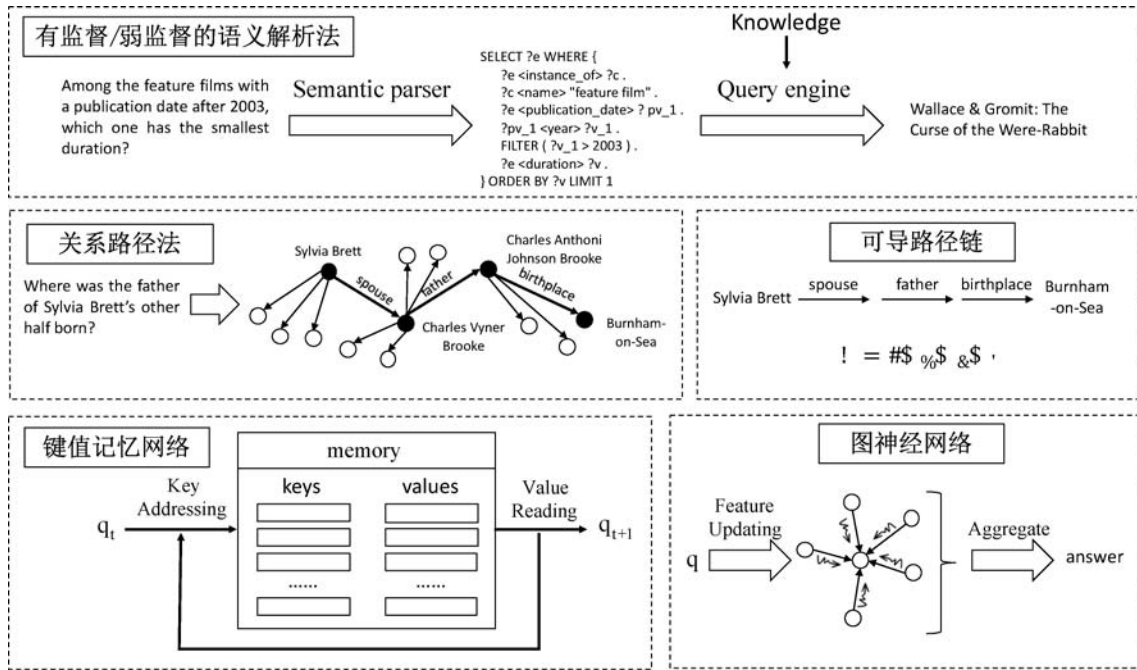


图 10 知识库推理问答六类方法的框架图

5.2.1 序列解析

Dong 等人<sup>[66]</sup>首次提出使用神经编码解码器来处理语义解析任务,他们提出了两种模型,分别是序列到序列模型(sequence-to-sequence, SEQ2SEQ)和序列到树模型(sequence-to-tree, SEQ2TREE),以将问题从文本形式解析为 Prolog、lambda-calculus 等逻辑形式.后者将逻辑形式作为树结构,充分利用了逻辑形式的结构化特征.令  $q = x_1, \dots, x_{|q|}$  表示输入的文本序列,  $a = y_1, \dots, y_{|a|}$  表示逻辑形式,两种模型的目标都是最大化如下概率:

$$p(a | q) = \prod_{t=1}^{|a|} p(y_t | y_{<t}, q).$$

在 SEQ2SEQ 模型中,作者使用层数为  $L$  的 LSTM<sup>[67]</sup>作为编码器,隐向量的计算公式为  $h_t^i = LSTM(h_{t-1}^i, h_{t-1}^{i-1})$ ,其中 LSTM 指 LSTM 函数,  $h_t^0$  是输入单词的词向量.解码器使用了相同结构的多层 LSTM,但参数与编码器相互独立.编码器最后一步的隐向量作为解码器第一步的隐向量,使用解码器最后一层的隐向量  $h_t^i$  预测输出的概率,即  $p(y_t | y_{<t}, q) = softmax(\mathbf{W}h_t^i)$ .该模型的缺点是将逻辑形式看作序列化文本,忽略了其结构特征.

SEQ2TREE 模型考虑了逻辑形式的结构特征,将逻辑形式作为树形结构进行解码.图 11 是该模型以 AB(C)为目标时的解码示例,其中括号内的内容(C)可以看作子树,  $t_1$  到  $t_6$  表示解码的步骤.解码时使用  $\langle n \rangle$  代表子树的根节点,当第一层的  $y_1$  到

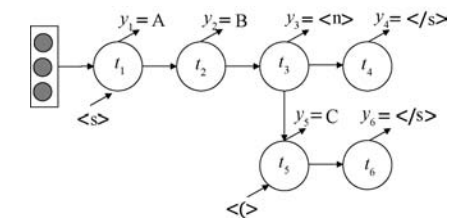


图 11 SEQ2TREE<sup>[66]</sup>模型解码过程示例

$y_4$  解码完成后,会针对  $\langle n \rangle$  所在的位置迭代进行子树的解码,得到  $y_5$  和  $y_6$ .  $\langle s \rangle$  和  $\langle /s \rangle$  作为解码的开始符号和结束符号.该模型使用同一个解码器进行各层的解码.论文在四个数据集上进行了实验,包含了 Prolog、lambda-calculus 等多种逻辑形式,SEQ2TREE 模型的性能均显著地优于 SEQ2SEQ 模型,表明了结构化信息的重要性.

5.2.2 多阶段分解

由于逻辑形式的复杂性,直接学习从文本形式到逻辑形式的映射是比较困难的,所以一些研究者考虑将解析拆分为多个阶段,从而降低任务的难度,提高解析的质量.

Dong 等人<sup>[68]</sup>提出一种粗粒度到细粒度(Coarse-to-fine)的解析方法,将语义解析拆分为两个阶段,第一阶段生成粗粒度的骨架,骨架中忽略了参数、变量等细节,仅考虑大体的逻辑关系.第二阶段以文本问题和生成的骨架作为输入,从文本中提取细粒度的信息,从而将骨架补充完整.图 12 中给出了三个不同的数据集上,问题的文本形式、逻辑形式、骨架

形式的例子,可以看到骨架形式比完整的逻辑形式缺少了一些细节信息.令  $x$  表示文本形式, $y$  表示逻辑形式, $a$  表示骨架形式,作者将原本的学习目标  $p(y|x)$  拆解为  $p(y|x) = p(y|x,a)p(a|x)$ ,然后使用两个不同的编码解码器模型分别进行学习.由于训练集中仅包含文本和逻辑形式,所以需要定义转换规则,将逻辑形式中的细节删掉,从而得到骨架的训练数据.

| 数据集    | 形式 | 例子  |
|--------|----|---|
| GEO    | 文本 | which state has the most rivers running through it?   |
|        | 逻辑 | (argmax \$0 (state:t \$0) (count \$1 (and (river:t \$1) (loc:t \$1 \$0))))                        |
|        | 骨架 | (argmax#1 state:t@1 (count#1 (and river:t@1 loc:t@2)))  |
| ATIS   | 文本 | all flights from dallas before 10am   |
|        | 逻辑 | (lambda \$0 e (and (flight \$0) (from \$0 dallas:ci) (< (departure time \$0) 1000:ti)))           |
|        | 骨架 | (lambda#2 (and flight@1 from@2 (< departure time@1 ?)))   |
| DJANGO | 文本 | if length of bits is lesser than integer 3 or second element of bits is not equal to string 'as', |
|        | 逻辑 | if len(bits) < 3 or bits[1] != 'as':  |
|        | 骨架 | if len ( NAME ) < NUMBER or NAME [ NUMBER ] != STRING :   |

图 12 Coarse-to-fine<sup>[68]</sup>的骨架语句示例

类似地,Zhang 等人<sup>[69]</sup>将语义解析任务拆分为三个阶段,第一阶段将输入的问题分解为若干个子问题,第二阶段对子问题提取细节,第三阶段整合得到最终的逻辑形式.该方法通过子问题分解,可以更好地处理复杂问题.

Sun 等人<sup>[70]</sup>拆分为四个步骤,先将问题解析为自定义的骨架形式,再利用骨架信息构建知识库无关的查询图,再将查询图中的节点和边链接到知识库中,最后对得到的候选查询图进行打分和排序.

Ding 等人<sup>[71]</sup>观察到逻辑形式中包含一些常用的子结构(substructure),提出先预测问题的子结构,再对子结构合并得到复杂的查询图,最后对查询图进行链接和排序.该方法与文献<sup>[69]</sup>的思路类似,都是将复杂问题分解为简单问题,不同之处在于该方法直接考虑子问题的结构化表示,更好地利用了结构信息.

### 5.2.3 重排序

文献<sup>[72,73]</sup>等工作主要关注对已生成的逻辑形式进行打分,经过重排序以得到更好的结果. Shao 等人<sup>[73]</sup>注意到使用 SEQ2SEQ 模型和 beam-search 得到的概率最大的前 5 个结果中,有 88.76%的比例包含目标逻辑语句,但如果只取概率最大的第 1 个结果,则只有 75.3%的比例得到目标语句,所以有必要对 beamsearch 得到的若干候选结果进行重排序,以提高得到正确逻辑语句的概率.文献<sup>[73]</sup>使用多层 Transformer<sup>[74]</sup>结构对逻辑形式提

取特征,并进一步考虑了逻辑形式的结构化特征,根据树形结构中节点的连接关系建立邻接矩阵,作为 Transformer 自注意力机制的掩码(mask).

### 5.2.4 其他方法

文献<sup>[75]</sup>提出了一种连续学习(continuous learning)的框架,在系统给出逻辑形式的初步预测后,人类对预测进行矫正,系统根据人类反馈,生成更加准确的逻辑形式,并用来增强模型,以避免再犯类似的错误.

文献<sup>[76]</sup>关注解析器在不同领域间的迁移性和扩展性,令多个领域共享同一个解析器,在解码的时候除了问题特征外,将领域特征也作为输入.

**小结:**先将问题解析为逻辑形式,再利用逻辑形式查询答案,此过程具有较好的可解释性,当逻辑形式中有错误时,人类也比较容易对错误进行定位.有监督的语义解析法预测出的逻辑形式质量较高,但是需要大量的训练数据作为支撑.逻辑形式本身标注难度大,标注成本高昂.

## 5.3 弱监督的语义解析法

逻辑形式的标注数据是非常昂贵的,所以一些研究者考虑仅依靠问题与答案来学习问题的逻辑形式.该类方法常使用基于状态转移的解析器,将逻辑形式看做图结构,定义一些对图的基本操作,然后逐步进行决策,对图结构进行扩充和完善.还有一些方法使用了模块化的函数,每个函数负责一个简单的功能,将复杂问题解析为函数的组合.由于缺乏直接的训练数据,所以常需要借助搜索、强化学习等方式来优化解析器,常用的做法是搜索得到一些候选的逻辑形式,然后最大化那些能得到正确答案的候选的概率.当逻辑形式变得复杂时,搜索空间会变得十分巨大,所以该类方法常使用较为简单的逻辑形式,并且常使用语法约束等技巧来帮助收敛.

### 5.3.1 问题解析为查询图

查询图是一种图结构的逻辑形式,由节点和边组成,节点通常对应实体、变量或约束,边通常对应节点之间的关系.

STAGG<sup>[77]</sup>模型是一个经典的基于转移的查询图模型,该模型定义的查询图形式包含:一个主题实体(topicentity);一个目标变量节点,对应答案;一条核心推理链(coreinferentialchain),是从主题实体到目标变量的一条路径,该路径上可以包含任意多个中间变量;每个变量节点可以包含任意多个约束,约束分为实体节点和聚合节点两种.图 13 给出了一个示例,其中 FamilyGuy 表示主题实体, $x$  表示

目标变量,  $y$  表示中间变量,  $\text{FamilyGuy} \rightarrow y \rightarrow x$  表示核心推理链,  $y \rightarrow \text{argmin}$  和  $y \rightarrow \text{MegGriffin}$  表示两个约束, 限定了  $y$  的角色是 MegGriffin 且  $y$  的时间是最早的.

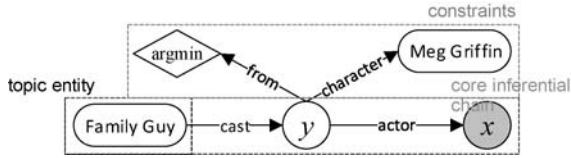


图 13 STAGG<sup>[77]</sup> 中针对问题“Who first voiced Meg on Family Guy?”的查询图结构

STAGG 将该查询图的生成过程定义为阶段式的状态转移过程. 令  $S = \cup \{ \emptyset, S_e, S_p, S_c \}$  表示状态集合, 其中  $\emptyset$  表示空图,  $S_e$  表示仅有主题实体的图,  $S_p$  在  $S_e$  的基础上补充了核心推理链,  $S_c$  又在  $S_p$  的基础上补充了约束, 对应完整的查询图. 令  $A = \cup \{ A_e, A_p, A_c, A_a \}$  表示决策集合, 决策会对图添加节点或边,  $A_e$  表示选择主题实体,  $A_p$  表示选择核心推理链,  $A_c$  和  $A_a$  表示添加实体约束和聚合约束. 图 14 是 STAGG 的状态转移图, 列出了状态和决策的转移关系.

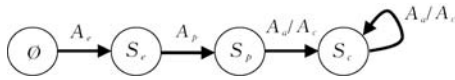


图 14 STAGG<sup>[77]</sup> 状态转移图

STAGG 直接使用实体链接作为  $A_e$ . 生成核心推理链被转化为谓语句的生成, 如图 13 中的  $\text{cast-actor}$ , 具体做法是使用卷积神经网络对问题和谓语句分别进行编码, 然后根据特征向量的相似性, 选择得分最高的谓语句. 约束  $A_c$  和  $A_a$  通过特定规则进行添加, 比如仅当问题中出现某个实体时才将它作为实体约束, 或者仅当问题中出现“first”“latest”等关键词时才使用聚合约束.

STAGG 要学习一个状态奖励函数  $\gamma(s, q)$ , 以评估查询图  $s$  作为问题  $q$  的逻辑形式的可能性. 由于缺乏直接的逻辑形式标注数据, 该模型使用排序法进行优化. 具体来说, 对于两个状态  $s_a$  和  $s_b$ , 执行对应的查询图, 计算得到的结果集合与标准答案集合的 F1 值, F1 值越高, 意味着查询图的质量越高. 该模型使用一个神经网络作为排序器, 手工定义了查询图 and 问题的特征提取方式, 将特征输入排序器中得到  $\gamma(s, q)$  的值. 在测试阶段, 该方法从可行的决策集合中选择能使  $\gamma(s, q)$  最大的决策.

许多研究工作在 STAGG 的基础上做了改进,

对查询图结构、决策空间、搜索算法等进行了优化. 文献[25]提出了 MULCG 的图结构, 扩充了 STAGG 中实体约束的范围和聚合约束的类型, 增强了查询图的表达能力. 文献[78]重新定义了决策空间, 提出了四种基本操作, 分别是连接两个节点、合并两个节点、扩充节点信息、删除无用节点, 利用这四种操作逐步生成查询图, 由于决策的粒度更细, 取得了比 STAGG 更好的性能. 文献[79]提出了一种语义匹配的模型, 将查询图和问题映射到同一个语义空间中, 通过计算特征向量的相似性对查询图进行打分, 从多个候选查询图中选出最好的. 文献[80]与文献[79]思路相似, 但进一步引入了图神经网络来计算查询图的特征向量, 从而更好地考虑了结构信息. 文献[81]针对直接学习复杂查询图比较困难的问题, 提出先将问题分解为若干个子问题, 先生成子问题对应的简单查询图, 再组合成完整的复杂查询图, 从而减小了搜索空间, 提高了结果的质量.

此外还有一些工作<sup>[82]</sup>提出先得到问题的语法依存树 (dependency tree), 再从依存树出发生成对应的逻辑形式, 这样可以利用依存树的结构特征, 提高解析质量. 文献[83]在其基础上进行了扩展, 用一个解析器同时处理多种不同的语言.

### 5.3.2 问题解析为程序

程序是一种序列或树状的逻辑形式, 它由一些基本的函数组成, 每个函数负责一个简单的功能, 如 SQL 语句中的 Select. 每个函数接收一定数量的参数, 如 Select 对应的列名. 将函数按照特定的顺序组合在一起, 就形成了程序, 可以用来实现复杂的功能, 如完整的 SQL 语句. 与查询图的不同之处在于, 查询图主要考虑节点和边, 而程序主要考虑函数、函数的参数、函数的组合顺序等.

早期的研究工作<sup>[23]</sup> 依赖词表匹配将文本中的单词映射到知识库上, 把映射得到的谓语句和实体作为基本函数, 然后定义了四种组合方式, 通过学习如何进行组合, 以得到完整的程序. 具体来说, 假设文本段  $[i_1:j_1]$  被映射为逻辑形式  $z_1$ , 文本段  $[i_2:j_2]$  被映射为逻辑形式  $z_2$ , 那么有四种方式得到文本段  $[\min(i_1, i_2): \max(j_1, j_2)]$  的逻辑形式, 分别是交集  $z_1 \sqcap z_2$ , 连接  $z_1 \cdot z_2$ , 聚合  $z_1(z_2)$ , 或者引入一个新的谓语句  $p$  进行桥接  $z_1 \sqcap p \cdot z_2$ . 通过递归地进行组合, 得到整个文本的逻辑形式. 为了学习如何从这四种方式中进行选择, 作者搜索一些可行的组合方式, 得到对应的逻辑形式, 运行之后把得到的答案与正确答案进行比较, 最大化那些能得到正确答案的概

率. 文献[84]提出了一种基于动态规划的搜索方法, 可以快速找出所有能得到正确答案的逻辑形式, 避免了搜索范围有限的问题.

文献[37]针对 SQL-like 的逻辑形式, 提出了一种基于状态转移的解析方法, 将所有函数(如 Select, Argmax 等)作为决策空间, 每一步根据策略网络进行选取. 在学习策略网络时, 作者通过搜索得到一些候选程序, 然后最大化那些能得到正确答案的概率. 这种搜索方式会得到一些虚假程序(spurious program), 即虽然可以得到正确答案, 但是语义上与问题并不匹配, 这些虚假程序会对解析器的学习造成负面影响. 针对这个问题, 文献[85]提出一种策略整形(policy shaping)的方式, 使用文本和程序中的词汇统计信息作为约束, 对策略网络的输出进行调整, 从而降低了虚假程序出现的概率. 文献[86]也提出了类似的词汇约束, 此外还提出了一种迭代训练的策略, 由于复杂的逻辑形式学起来比较困难, 该论文提出从简单的逻辑形式开始训练, 随着训练的进行, 逐渐增加逻辑形式的复杂性, 从而提高了训练的稳定性.

文献[87]提出针对二叉树形式的程序, 可以使用堆栈(stack)来帮助解析. 该论文使用 FunQL<sup>[1]</sup> 的逻辑形式, 定义了终结符入栈、非终结符入栈、归并出栈三种类型的决策, 并使用 stack-LSTM<sup>[88]</sup> 对堆栈信息进行编码.

文献[89]提出了神经符号机(Neural Symbolic Machine, NSM), 该模型包含一个基于 SEQ2SEQ 的解析器(programmer), 用来将文本序列转换为程序序列, 以及一个针对特定逻辑形式的执行器(computer), 用来执行程序得到答案, 同时为解析器提供语法约束, 以减小程序的搜索空间. 在训练解析器时, 该论文提出了一种强化学习与迭代极大似然(iterative maximum likelihood)相结合的方法. 在强化学习中, 对于一个采样的程序序列, 用执行器得到它的答案, 并与标准答案进行比较, 用两个答案集合的 F1 值作为激励. 由于搜索空间巨大, 直接使用强化学习难以收敛, 该论文又引入了迭代极大似然, 具体做法是找出当前参数下最好的一个程序(即 F1 值最大的), 直接最大化该程序的概率. 文献[90-92]与 NSM 的思路类似, 都使用了基于 seq2seq 的模型作为解码器, 不同之处在于, 文献[90]针对的是对话任务, 在解码时还需要考虑历史信息, 而文献[91, 92]使用了自定义的函数集合, 具有更强的表达能力和更高的自由度, 相应地, 搜索空间更大, 所以

使用了更多的约束来帮助收敛. 文献[93]将 NSM 扩展到了多领域的场景, 提出训练一个通用的解析器来同时处理多个领域的问题.

**小结:**在弱监督的语义解析法中, 常用的逻辑形式可以分为查询图和程序两种, 前者都是将问题解析为图结构, 主要考虑节点、边的添加和删除等, 后者则将问题解析为序列或树状结构, 主要考虑函数的定义、参数、组合顺序等. 在学习解析器时, 最常用的方法是从搜索得到的逻辑形式中, 选择那些能执行得到正确答案的进行优化. 由于搜索空间巨大, 收敛困难, 往往需要一定的约束和训练策略. 与有监督的方法相比, 弱监督的方法不依赖逻辑形式的标注数据, 节省了标注成本, 但其问题在于训练比较困难, 准确率不如有监督的高.

#### 5.4 关系路径法

关系路径法是一种专门针对多跳路径的推理问题的方法, 即问题由一个实体和多个关系组成, 如“姚明效力的球队的老板是谁”. 该类方法通常先找到问题中的主题实体(topic entity)并链接到知识库中, 如“姚明”, 然后在知识库中选择关系, 以跳转到新的实体, 迭代地进行跳转直到找到最终的答案实体.

Zhou 等人<sup>[94]</sup>提出了可解释推理网络(Interpretable Reasoning Network, IRN)用于处理多跳问题. IRN 是一个迭代模型, 包含三个模块, 分别是输入模块、推理模块和答案模块, 模型框架图如图 15 所示. 输入模块用于对问题进行编码, 并根据上一步的推理结果进行更新, 具体公式为  $q^0 = \sum_{i=1}^n x_i$ ,  $q^h = q^{h-1} - M_{rq} \hat{r}^h$ , 其中  $q^h$  表示第  $h$  步的问题特征,  $x_i$  表示输入问题的词向量,  $\hat{r}^h$  是上一步推理模块的输出,  $M_{rq}$  是一个映射矩阵. 推理模块维护一个状态向量  $s^h$ , 每一步预测一个新的关系  $\hat{r}^h$ , 并更新状态向量, 具体公式为  $g_j^h = \text{softmax}(f(q^{h-1}) + g(s^{h-1}))$ ,  $\hat{r}^h = \sum_j g_j^h r_j$ ,  $s^h = s^{h-1} + M_{rs} \hat{r}^h$ , 其中  $r_j$  表示知识库中第  $j$  个关系的特征向量,  $s^0$  使用主题实体的向量表示进行初始化. 推理模块每一步对所有关系计算了一个概率分布, 从而选出所关注的关系. 答案模块用推理模块的状态作为输入, 预测得到的实体, 具体公式为  $e^h = M_{se} s^h$ ,  $o_j^h = \text{softmax}(e_j^T e^h)$ , 其中  $e_j$  表示知识库中第  $j$  个实体的特征向量. 由于迭代计算过程都是可导的, 所以 IRN 可以只根据最终的答案进行训练, 不需要关于中间过程的监督信号. 文献



[95]提出的 UHop 模型在 IRN 的基础上做了进一步改进,支持任意多跳,在空间上也更加高效,可以处理大规模知识库.

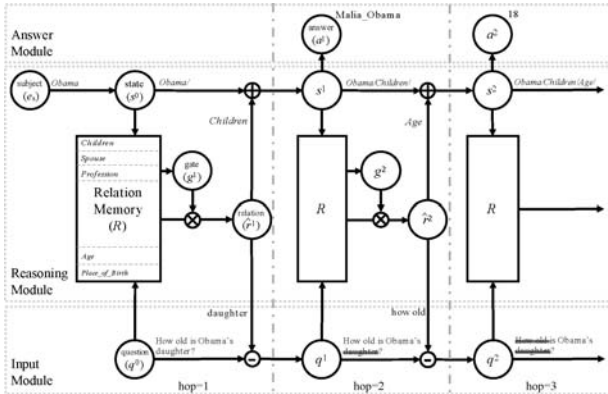


图 15 IRN<sup>[94]</sup> 模型框架图

IRN 必须预先识别出主题实体,这在一定程度上限制了其应用范围. Zhang 等人<sup>[28]</sup>提出了变分推理网络(Variational Reasoning Network, VRN),可以同时推断主题实体和关系路径. 该论文将主题实体作为隐变量,目标函数为

$$\max_{\theta_1, \theta_2} \frac{1}{N} \sum_{i=1}^N \log(P_{\theta_1}(y | q_i) P_{\theta_2}(a_i | y, q_i)),$$

其中  $q_i$  表示问题,  $a_i$  表示答案,  $y$  表示主题实体,  $P_{\theta_1}(y | q_i)$  表示根据问题预测主题实体,  $P_{\theta_2}(a_i | y, q_i)$  表示根据问题和主题实体共同预测答案. 前者的实现比较简单,直接用问题的向量表示做分类即可:  $P_{\theta_1}(y | q) = \text{softmax}(W_y^T f_{ent}(q))$ , 其中  $W_y$  表示实体  $y$  的向量表示,  $f_{ent}$  表示问题编码器. 后者通过遍历以  $y$  为中心的子图实现,具体来说,定义  $G_{y \rightarrow a}$  为包含从  $y$  到  $a$  的所有路径的子图,令  $g(G_{y \rightarrow a})$  表示子图的特征向量,那么答案的概率分布可以由如下式子得到:  $P_{\theta_2}(a | y, q_i) = \text{softmax}(f_{qt}(q)^T g(G_{y \rightarrow a}))$ , 其中  $f_{qt}(q)$  表示另一个问题编码器. 子图的特征以迭代的方式计算,即先找到从  $y$  到  $a$  的路径上,位于  $a$  之前的实体  $a_j$ , 将这样的实体集合记为  $Parent(a)$ , 然后根据  $G_{y \rightarrow a_j}$  的特征来计算  $G_{y \rightarrow a}$  的特征. 具体公式为

$$g(G_{y \rightarrow a}) = \frac{1}{\# Parent(a)} \sum_{a_j} \sigma(\mathbf{V}[g(G_{y \rightarrow a_j}), \mathbf{e}_r]),$$

$$a_j \in Parent(a), (a_j, r, a) \text{ or } (a, r, a_j) \in G_y$$

其中  $\mathbf{e}_r$  表示关系  $r$  的向量表示,  $\mathbf{V}$  表示映射矩阵. 由于需要遍历所有可能的子图,计算过程是比较耗时的. 整个模型通过强化学习进行优化.

文献[96]在 VRN 的基础上做了进一步的改进,不再对所有以  $y$  为中心的子图计算概率,而是

像 IRN 一样采用序列决策过程,每一步预测一个关系,并且使用束搜索来减少候选数量,大大减少了运行时间. 此外还引入了注意力机制和基于势能的激励整形策略(potential-based reward shaping),提高了预测的质量.

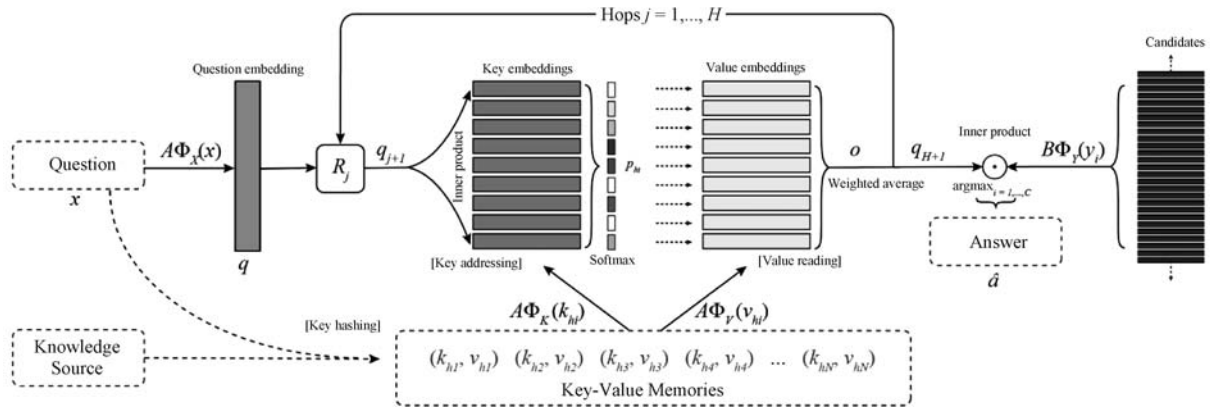
**小结:** 关系路径法适合处理多跳路径问题,可以在预测答案的同时,也预测出中间的关系路径,具有较好的可解释性. 但该方法表达能力较弱,应用范围比较局限,除了多跳关系之外,无法处理其他推理问题,如集合的操作、数值的比较等.

## 5.5 键值记忆网络

键值记忆网络(Key-Value Memory Network, KV MemNet)是记忆网络(Memory Network)<sup>[97]</sup>的一种变体,引入了键值对(key-value pairs)来存储知识.

图 16 给出了经典的键值记忆网络<sup>[27]</sup>的框架图,其中包含三个主要步骤,分别是键哈希(key hashing)、键索引(key addressing)、值读取(value reading). 键哈希是根据当前的问题,从知识库中提取出最相关的知识,构建记忆 memory. 因为整个知识库往往非常巨大,如果将其全部放入 memory 中,既耗费内存又难以优化,所以需要从中提取一个包含少量三元组的子集. 具体做法是将三元组的主语和谓语拼接起来作为 key,将宾语作为 value,用 key 跟输入的问题进行比较,选取那些重合度最高的,放入 memory 当中. 键索引是用当前的查询向量  $q_j$  与所有的 key 计算相似度,得到一个概率分布,公式为  $p_i = \text{softmax}(q_j^T A \Phi_K(k_i))$ , 其中  $\Phi_K$  用来将键  $k_i$  映射为向量表示,  $A$  是一个参数矩阵. 值读取是根据刚刚得到的概率分布,对所有 value 的向量表示计算加权平均,然后用来更新查询向量,公式为  $o = \sum_i p_i A \Phi_V(v_i)$ ,  $q_{j+1} = R_j(q_j + o)$ , 其中  $\Phi_V$  表示值的向量表示,  $R_j$  表示第  $j$  次迭代的参数矩阵. 通过迭代进行  $H$  步键索引和值读取操作,得到最终的查询向量  $q_{H+1}$ , 该向量包含了问题和知识库的信息,将其输入分类器中来预测答案.

键值记忆网络结构简单,容易优化,可以比较方便地处理稍微简单一点的问题,对于复杂问题则表现较差. 针对这个问题,文献[98]主要提出了两点改进. 一是在更新查询向量时,不再简单地使用  $q_j + o$ , 而是将  $q_j$ 、 $o$  和  $\sum_i p_i A \Phi_K(k_i)$  三个向量拼接在一起之后进行融合,这样可以更好地考虑上一步的查询信息,避免进行重复查询. 二是在 memory 中加入一个特殊的 STOP 符号,用来提前终止迭代,避

图 16 键值记忆网络<sup>[27]</sup>框架图

免进行无用查询。

文献[99]提出了一种双向注意力记忆网络,通过多层的注意力机制,可以从知识库中提取出跟问题更相关的信息。但该方法并没有使用键值对,而是直接选出一些候选的答案实体作为 memory。

**小结:**键值记忆网络结构简单,通用性强,但是推理能力较弱,可解释性也较差,迭代更新查询向量的过程难以被人类所理解,也无法定位错误。

## 5.6 可导路径链

可导路径链方法用权重向量表示实体集合,用矩阵乘法表示实体集合沿某个关系的跳转结果,跳转之后得到新的权重向量,可以继续跳转。实体集合的交、并等操作也都以可导的方式实现。这样整个推理过程就可以端到端地进行优化,而且运行效率很高,可以方便地部署到多个 GPU 上进行并行计算。

该类方法最早由 Cohen 等人<sup>[100]</sup>提出。令  $s$  表示实体  $s$  对应的 one-hot 向量,即  $s$  中只有与  $s$  对应的位置是 1,其他位置都是 0。令  $M$  表示关系  $r$  对应的邻接矩阵,即  $M_{i,j}=1$  表示实体  $i$  和实体  $j$  之间存在关系  $r$ ,否则  $M_{i,j}=0$ 。那么实体  $s$  经过关系  $r$  进行跳转之后的结果就可以用向量与矩阵相乘,即  $s \cdot M$  进行表示,返回一个新的 one-hot 向量,表示跳转之后得到的实体。当  $s$  不再是 one-hot 向量,而是变为一个集合在所有实体上的概率分布时,同样可以使用该方式进行计算,且返回的结果也变成了概率分布。计算两个实体集合的并集,可以用  $s+t$  实现,计算交集可以用  $s \odot t$  实现,其中  $\odot$  表示按位相乘。这些操作的输出都是新的实体向量,可以继续输入到下一步操作中,从而形成一条路径链,导数可以沿着该路径链进行传递。

Cohen 在文献[101]中对该方法的具体实现做了优化,进一步提高了运行速度,并且支持在更大规

模的知识库上进行运算。文献[102]引入了 count-min sketch 的数据结构来对实体向量进行压缩,以替代稀疏的权重向量,支持更加紧凑的实体表示。

可导路径链方法与关系路径法有相似之处,都适合用来处理多跳路径问题,可以在预测答案的同时,预测出中间的关系路径。不同之处在于,在关系路径法中,实体沿关系的跳转是离散表示的,无法将导数沿着路径进行传递,所以常常需要借助强化学习等方式进行优化。而可导路径链法中的跳转用矩阵乘法实现,计算是连续的,可以将导数传递至整个推理过程,直接端到端地进行优化。另外,可导路径链法可以处理集合的交、并等操作,这是关系路径法做不到的。

此外,可导路径链方法与弱监督的语义解析法中的程序法也有相似之处。关系跳转、集合操作都可以看作基本函数,通过将这些基本函数进行组合得到程序,以完成复杂功能。

**小结:**可导路径链法适合处理多跳路径和集合操作问题,可以端到端地进行优化,比关系路径法具有更强的表达能力。但是无法处理其他推理问题,如属性查询、数值比较等。

## 5.7 图神经网络

随着 GCN<sup>[103]</sup>、GAT<sup>[104]</sup>等模型的提出,图神经网络获得了广泛的关注,在阅读理解、视觉问答等众多领域取得了较大的进展。以 GCN<sup>[103]</sup>为例,它的全称是图卷积网络(GraphConvolutional Networks),它以如下方式更新节点的特征:

$$\mathbf{H}^{(l+1)} = \sigma(\mathbf{D}^{-\frac{1}{2}} \tilde{\mathbf{A}} \mathbf{D}^{-\frac{1}{2}} \mathbf{H}^{(l)} \mathbf{W}^{(l)}).$$

其中  $\mathbf{H}^{(l)}$  表示第  $l$  层的节点特征矩阵,  $\mathbf{H}^{(0)}$  用输入的节点特征  $X$  进行初始化。  $\mathbf{W}^{(l)}$  表示第  $l$  层的参数矩阵。  $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_N$ , 其中  $\mathbf{A}$  是图的邻接矩阵,  $\mathbf{I}_N$  是

单位矩阵,表示节点的自连接.  $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ ,表示基于节点出度的正则化. 如图 17 所示,GCN 每一层都是将节点的邻居信息集成起来,用来更新当前节点的特征.

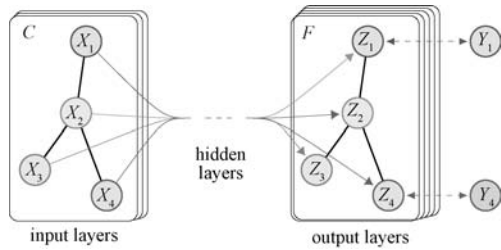


图 17 GCN<sup>[103]</sup>示意图

由于知识库天然就是图结构,应用图神经网络进行推理是一个比较自然的思路.

Sun 等人<sup>[35,105]</sup>使用图神经网络处理推理问答任务,但他们的知识库包含两部分知识,一是结构化的知识库,二是非结构化的纯文本,所以对于特定的问题,需要先从两种知识中选取最相关的信息,然后将选出的文本转化为图,与知识库合并为一个整体的图结构,最后再应用图神经网络. 图 18 给出了将知识库和文本融合为图结构的一个示例.

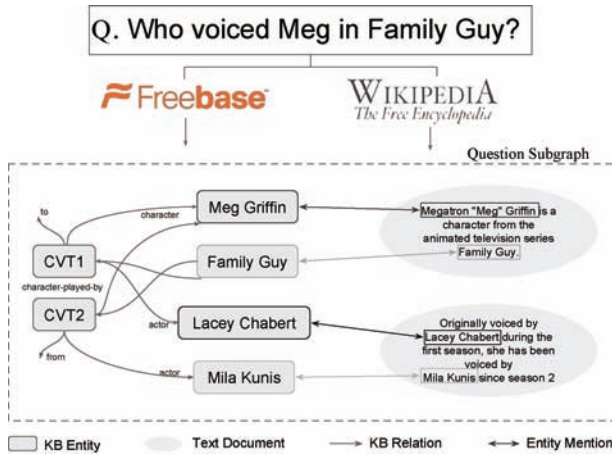


图 18 融合知识库与文本的图结构

文献[105]主要使用启发式的规则进行子图构建,图的质量较低,文献[35]中提出的 *PullNet* 模型针对这个问题,使用学习的方式动态抽取子图,显著提升了子图的质量和答案的准确率. 具体来说,令  $G_q = \{V, E\}$  表示问题  $q$  对应的子图,其中  $V$  表示节点集合,有三种类型的节点,分别是实体节点  $V_e$ , 对应知识库中的实体,文档节点  $V_d$ , 对应文本中的文档,事实节点  $V_f$ , 对应知识库中的三元组. 文档需要进行实体链接(entity linking),从而建立实体节点和文档节点之间的边. 事实节点与它的头实体

和尾实体之间存在边. 令  $G_q^0 = \{V^0, E^0\}$  表示初始的子图,其中  $V^0$  由问题中包含的实体组成,  $E^0$  为空. *PullNet* 要迭代地对  $G_q^0$  进行扩展. 在每次迭代中,先对知识库中的节点进行打分,选出得分超过某个阈值的实体. 对于每个选出的实体  $v_e$ , 使用基于 *IDF* 的信息检索方法从文本中选出与  $q$  最相关且包含实体  $v_e$  的文档集合,使用基于相似度的方法从知识库中选出与  $q$  最相关且包含实体  $v_e$  的三元组事实集合. 然后将这些新得到的节点和相应的边加入到  $G_q$  当中. 迭代若干步之后,得到最终的子图,使用图神经网络对子图提取特征,并对子图中的每个实体节点预测一个得分,最后选择得分最高的实体作为问题的答案. 由于缺乏子图的监督信号, *PullNet* 使用弱监督的方式进行训练,作者先找到从问题中的实体到答案实体的最短路径,以最短路径作为参考来判定子图生成过程的正负例.

*PullNet* 的子图提取过程与弱监督的语义解析法中的查询图法类似,都是从一个初始的图结构出发,迭代地进行扩展. *PullNet* 在训练时缺乏子图的监督信号,也属于弱监督学习. 不同之处在于, *PullNet* 的图是从知识库和文本中提取出的子集,而查询图是问题的语义解析,两种图的含义并不相同.

小结:图神经网络近年来发展迅速,在多个领域取得较大进展. 图神经网络通过迭代的信息整合进行推理,具有较强的表达能力. 知识库非常适合使用图神经网络,但其缺点是当前图神经网络缺乏可解释性,无法给出中间推理过程,当预测的答案出错时,难以对错误进行定位.

### 5.8 其他方法

除了以上介绍的六类方法之外,还有一些其他方法,尝试用别的思路去处理知识库推理问答任务.

文献[106]使用信息传递(message passing)的思路进行推理. 该方法分为两个阶段:第一阶段先将问题中的实体、关系、类别识别出来,链接到知识库中,并计算各自的置信度(confidencescore). 第二阶段将置信度沿着图结构进行传递,更新节点的得分. 文献[107]也将推理分为两个阶段,第一阶段根据问题中的实体和关系抽取候选答案集合,第二阶段引入外部知识库维基百科数据对候选答案集进行验证,筛选出最合理的答案.

文献[108]用表示学习的思路进行推理,即将知识库中的实体和待查询的问题表示到同一个向量空间内,使得答案和问题的距离尽可能接近. 这种思路对于简单问题比较好用,但对于复杂问题,尤其是涉及到集

合的交、并等操作时,就难以得到预期的效果了.针对这个问题,文献[108]提出了 QUERY2BOX 模型,不再将查询问题表示为一个向量,而是表示为向量空间中的一个矩形区域(box),从而可以表示一个实体集合,所有包含在 box 内的实体都属于该集合.集合的交、并也可以方便地通过对 box 的操作得到.

文献[34]针对在稀疏和不完整的知识库上进行推理的任务,提出将知识库表示学习与推理相结合的方法.知识库表示是将知识库中的实体和关系映射到同一个向量空间中,并尽量保证向量的传递性,如在经典的知识库表示模型 TransE<sup>[56]</sup>中,目标函数是  $h+r=t$ ,其中  $h, r, t$  分别对应头实体、关系、尾实体的向量表示.知识库表示常用来做知识库补全,以预测知识库中缺失的链接(linkprediction).通过将知识库表示和知识库推理问答两个任务结合在一起,可以利用链接预测的结果补充额外信息,提高推理性能.

## 5.9 小 结

本章将知识库推理问答的主流方法分为六个类别,表 3 分别总结了每个类别的核心思路和代表性

方法,并分析了每类方法的优点和缺点.

(1)有监督的语义解析法利用标注数据学习如何将问题解析为逻辑形式.逻辑形式容易被人类所理解,也容易定位错误.但标注数据成本很高;

(2)弱监督的语义解析法仅根据答案来学习如何解析逻辑形式.与有监督的方法相比,它节省了标注成本,但训练比较耗时,收敛也比较困难;

(3)关系路径法专门用于处理多跳路径问题,可以预测出关系路径,具有较好的可解释性,但无法处理其他推理问题;

(4)键值记忆网络结构简单,通用性强,但推理能力弱,推理过程无法可视化;

(5)可导路径链以可导的方式处理多跳路径和集合操作,可以直接端到端地进行优化.它与关系路径法类似,可以预测中间路径,但无法处理属性、比较等问题;

(6)图神经网络通用性强,可以用在知识库与文本等其他形式的知识结合的任务中,有较强的推理能力,但推理过程无法可视化.

表 3 知识库推理问答方法分类与比较

| 方法                                | 核心思路  | 优点              | 缺点                           |
|-----------------------------------|---|-----------------|------------------------------|
| 有监督的语义解析法 <sup>[62-76]</sup>      | 设计具有强大表达能力的解析器,采用多步分解、重排序等方式提高解析质量                | 解析质量高           | 需要大量的逻辑形式标注数据,代价昂贵           |
| 弱监督的语义解析法 <sup>[77-93]</sup>      | 基于状态转移或 seq2seq 将问题解析为查询图或程序,借助搜索提供训练信号,添加约束来帮助收敛 | 不依赖逻辑形式标注       | 搜索空间巨大,需要大量的时间和空间,收敛困难       |
| 关系路径法 <sup>[28][94][95][96]</sup> | 从主题实体出发,预测关系路径进行跳转                                | 适合处理多跳关系问题      | 除多跳关系问题外,无法处理其他推理问题          |
| 键值记忆网络 <sup>[27][98][99]</sup>    | 基于知识库构造 memory,根据问题迭代地从 memory 中读取信息              | 结构简单,通用性强       | 无法处理太复杂的问题,缺乏可解释性            |
| 可导路径链 <sup>[100-101]</sup>        | 以可导的方式表示实体集合的跳转、交并等操作,得到的输出继续输入下一步                | 可以端到端优化,计算高效    | 除多跳和集合操作问题外,无法处理其他推理问题,如数值比较 |
| 图神经网络 <sup>[35][105]</sup>        | 使用图神经网络迭代更新知识库中实体的向量表示,用得到的实体特征预测答案               | 图神经网络通用性强,表达能力强 | 缺乏可解释性                       |

在这些方法中,键值记忆网络和图神经网络属于感知系统,依靠神经元结构进行信息传递和隐式推理,而语义解析法、关系路径法和可导路径链则在一定程度上属于认知系统,即通过符号化的逻辑形式对问题进行求解.基于神经网络的感知系统不依赖专家知识,通用性强,而基于符号的认知系统准确率高,推理能力强,也符合人类的思考过程.如何将两类系统进行深入的结合,是一个值得研究的问题.

表 4 中给出了推理问答的代表性方法在常用数据集上的实验结果.早期的研究工作主要集中在 ATIS、WebQuestions 等数据集上,最近的研究工作

则更关注具有 SPARQL 标注的 WebQuestionsSP 和规模更大的 MetaQA.其中 PullNet<sup>[35]</sup>借助图神经网络在 MetaQA 的三跳问题上取得了 91.4% 的准确率,是目前效果最好的推理模型之一.

## 6 未来研究方向

随着近年来深度学习的发展,简单问答已经取得了接近人类表现的效果.Petrochuket 等人<sup>[53]</sup>提出 SimpleQuestions 数据集的准确率上限为 83.4%,而 Zhao 等人<sup>[39]</sup>提出的方法达到了 85.44% 的准确率,已经超过了人类性能.不仅在知识库问答领域,在其

他问答领域,如文本阅读理解任务上,机器在简单问题上的表现也已经达到甚至突破了人类水平,如在广泛使用的阅读理解数据集 SQuAD<sup>[109]</sup>上,人类的

F1 值为 89.45,而最好的模型已经达到了 93.01.因此,近年来研究者的关注重点渐渐转移到更具挑战性的推理问答任务上来.

表 4 推理问答代表性方法的实验结果对比

| 数据集<br>模型                   | ATIS (Accuracy) | WebQuestions (F1) | WebQuestionsSP (F1) | MetaQA (Accuracy) |       |       |
|-----------------------------|-----------------|-------------------|---------------------|-------------------|-------|-------|
|                             |                 |                   |                     | 1-hop             | 2-hop | 3-hop |
| SEQ2TREE <sup>[66]</sup>    | 84.6%           | —                 | —                   | —                 | —     | —     |
| COARSE2FINE <sup>[68]</sup> | 88.0%           | —                 | —                   | —                 | —     | —     |
| STAGG <sup>[77]</sup>       | —               | 52.5%             | 66.8%               | —                 | —     | —     |
| STF <sup>[78]</sup>         | —               | 53.6%             | —                   | —                 | —     | —     |
| SCANNER <sup>[87]</sup>     | —               | 49.4%             | —                   | —                 | —     | —     |
| NSM <sup>[89]</sup>         | —               | —                 | 69.0%               | —                 | —     | —     |
| VRN <sup>[28]</sup>         | —               | —                 | —                   | 97.5%             | 89.9% | 62.5% |
| SRN <sup>[96]</sup>         | —               | —                 | —                   | 97.0%             | 95.1% | 75.2% |
| KVMemNet <sup>[27]</sup>    | —               | —                 | 46.7%               | 95.8%             | 25.1% | 10.1% |
| ReifKB <sup>[101]</sup>     | —               | —                 | —                   | 96.2%             | 81.1% | 72.3% |
| PullNet <sup>[35]</sup>     | —               | —                 | 68.1%               | 97.0%             | 99.9% | 91.4% |

要想让计算机具备像人一样的推理能力是非常困难的.今天的深度学习技术更适用于感知(perception)层面,如从图像中识别物体,从文本中识别实体等.但是像人类一样,利用已有的知识去思考和推理,单纯依靠深度学习是做不到的,还需要借助基于符号的认知(recognition)系统.我们认为,未来的研究可以从这些方向着手:

神经与符号相结合.使用神经网络进行感知,使用符号进行推理,充分发挥两种方式各自的长处.已经有许多工作在这一方向上进行了探索,如文献<sup>[89]</sup>中提出的神经符号机(neuralsymbolicmachine),先用神经网络对问题进行解析,将其转化为符号化的程序,再用程序引擎执行得到答案.语义解析法普遍使用类似的思路,用神经网络作为解析器,再用符号引擎执行逻辑形式.我们认为现有的结合方式还存在很多问题,比如符号引擎鲁棒性很差,解析出的逻辑形式稍有偏差就可能导致结果变错,再比如解析器需要针对逻辑形式的语法特点进行专门设计,缺乏通用性,等等.这些问题都有待进一步的研究.

神经模块网络.神经模块网络(neuralmodule-networks,NMN)最早由文献<sup>[110]</sup>提出,常被应用在视觉推理(visualreasoning)任务中.该方法也属于神经与符号相结合的一种方法,与语义解析法中的程序法类似,利用语义组合性(compositionality),将复杂问题分解为若干个函数组成的序列.不同之处在于,NMN中的函数,又称为模块,是由可训练的神经网络实现,而不是基于规则的符号引擎.由函数组合成的程序是一个可导的神经网络,可以端到端地进行优化,与可导路径链<sup>[100][101][102]</sup>类似.

用神经网络实现的函数比基于规则的函数具有更好的鲁棒性,如函数的输入由“红色”变为“赤色”时,基于神经网络的函数返回结果基本不变,而基于规则的函数则可能无法找到“红色”的结果.NMN在基于文本的推理任务上也有应用<sup>[111]</sup>,但在基于知识库的推理上还没有应用.如何在大规模的知识库上应用神经模块网络,是一个很有价值的研究方向.

推理过程的可解释性.人类在解决复杂问题时,思路是可以逐步列出并被其他人理解的,我们希望机器也是如此,可以将其“思考”过程显式地给出,以方便人类进行理解,当机器出错时也方便对其“思考”过程进行排查,以定位错误出现的原因.在之前介绍的六类方法中,语义解析法、关系路径法、可导路径链等方法具有较好的可解释性,而键值记忆网络、图神经网络的可解释性就比较差.在未来的模型设计中,可解释性应当作为一个重要的考虑因素.

建立通用的可扩展的函数库.函数相当于计算机的基本技能,当计算机掌握某项技能之后,就可以应用该技能去处理更为复杂的任务.我们认为这与人类的学习过程是一致的.例如人类在刚开始学习数学时,要先学习十以内的加减法,将十以内的加减法掌握之后,就可以学习两位数 and 更多位数的加减法.在计算多位数的加减法时,要先按位对齐,然后每一位上都是按照十以内的加减法来计算的,只不过加法需要进位(如 $17+19$ ,先计算个位的 $7+9$ ,需要向十位进1),减法需要退位(如 $17-9$ ,需要向十位借1).所以多位数的加减法实际上由两个技能组合而成,即十以内的加减法和进退位.随着学习的进一步深入,任意位数的加减法也变成了基本技能,在计

算乘法时需要用到. 随着掌握的技能越来越多, 只要将这些技能进行合理的组合, 人类就可以解决很多复杂的问题. 我们认为这个过程是通向通用人工智能的一条可行之路. 我们需要让计算机掌握越来越多的基本技能, 即学会越来越多的函数, 然后让计算机学会组合这些函数去解决复杂问题. 我们认为, 可以尝试建立通用的函数库, 每个函数负责一个特定的功能. 对于人工智能相关的任何任务, 我们首先考虑能否利用已有函数的组合来解决, 如果不能, 则考虑在函数库中加入新的函数, 使得该任务可以被解决. 每个函数的内部实现需要人工设计, 并且可以随着技术进步而不断优化, 如用于图像分类的函数, 从简单的 CNN 发展到 AlexNet<sup>[112]</sup> 再到 ResNet<sup>[113]</sup>. 这种思路的难点在于: 一是函数的定义难统一, 每种函数有不同的输入输出形式, 限制了函数之间的组合; 二是对于新的任务, 缺乏如何利用现有函数的组合将其解决的监督信息, 可能需要大量的人类标注; 三是难以判断何时需要在库中加入新的函数, 以及新函数的功能和实现如何设计. 虽然这些困难都非常具有挑战性, 但我们相信这个方向是走向通用人工智能的一条可行之路.

从面向数据集的研究, 转变为面向应用的研究. 现有的很多研究工作将注意力集中在某个特定的数据集上, 针对数据集的特点进行很多专门的设计, 导致得到的模型虽然在该数据集上表现很好, 但很难应用到实际场景中. 不单单是知识库问答领域, 这个问题在很多人工智能研究领域都普遍存在, 这既限制了技术的落地进程, 又浪费了大量的研究资源. 我们认为, 未来的研究要更多地考虑实际应用, 而不仅仅是拟合某个特定的数据集. 数据集是研究的工具, 而不是研究的目的. 未来应当构造更贴近实际场景的数据集, 并且构造更大规模的测试基准数据集, 以此为基础, 提高模型的通用性, 推动模型的真正落地.

人机结合的持续学习. 人类在学习过程中必然会犯错, 通过对错误进行纠正, 人类可以积累经验, 避免将来再犯类似的错误. 俗话说“熟能生巧”, 任何学习都不是一蹴而就的, 人类需要经过十几年甚至几十年的持续学习, 才能在某个领域成为专家, 我们认为机器同样如此. 要想实现真正像人一样的智能, 机器也需要大量的持续的学习和练习, 并在这个过程中不断接受人类反馈, 纠正错误, 从而逐步变得完善. 文献[75]已经在这方面做出了一些探索, 我们希望未来可以有更多的研究者关注这个方向, 建立更

好的反馈学习和持续学习机制, 并在实际系统中真正应用起来, 从而获得大量人类用户的持续反馈.

与其他知识源相结合, 如文本、图像. 已经有一些研究工作在这方面取得了一定进展, 如文献[105][35]探索了如何利用知识库和文本共同作为背景知识, 将文本转化为图结构之后与知识库融合起来, 使用图神经网络进行推理, 文献[114][115]利用知识库增强了视觉问答的性能. 在未来的研究中, 我们认为多知识源结合将越来越普遍, 因为实际场景往往是包含多种信息的, 人类可以同时处理视觉、听觉、嗅觉、触觉等多种信息, 做出正确的推理和决策, 我们希望机器也可以如此, 关键在于如何让机器学会从不同来源、不同模态的知识中整合信息. 我们认为, 可以像文献[105][35]一样, 将不同来源的知识都解析为符号化的图结构, 这样既方便融合, 也方便做显式推理.

总的来说, 在知识库问答领域, 未来的研究工作应当以可解释的、可落地的、可持续学习的模型和系统作为研究重点. 我们希望该领域可以作为通用人工智能的一个突破点.

## 7 结 论

本文针对知识库问答领域进行了系统的介绍. 我们整理了近年来该领域常用的数据集, 分析和对比了各个数据集的优缺点. 针对问题的复杂程度, 我们将知识库问答任务分为简单问答和推理问答两类, 其中简单问答只涉及到单个三元组, 推理问答则要复杂得多, 要求计算机具备集合操作、多跳路径、属性比较等多种推理技能. 简单问答的主流方法为子图选择和答案三元组选择的级联任务. 推理问答的主流方法可以分为有监督的语义解析法、弱监督的语义解析法、关系路径法、键值记忆网络、可导路径链、图神经网络六类, 我们分别介绍了这六类方法的核心思路 and 代表性工作, 并分析了每类方法的优缺点. 最后我们对知识库问答领域的未来研究方向进行了展望, 希望在推动该领域发展的同时, 以该领域为着力点, 探索通用人工智能的发展之路.

## 参 考 文 献

- [1] Lehmann J, Isele R, Jakob M, Jentzsch A, Kontokostas D, Mendes P N, Hellmann S, Morsey M, van Kleef P, Auer S, Bizer C. Dbpedia — a large-scale, multilingual knowledge base

- extracted from Wikipedia. *Semantic Web*, 2015, 6(2): 167-195
- [2] Bollacker K, Evans C, Paritosh P, Sturge T, Taylor J. Freebase: A collaboratively created graph database for structuring human knowledge//*Proceedings of the ACM SIGMOD International Conference on Management of Data*. Vancouver, Canada, 2008: 1247-1250
- [3] Suchanek F M, Kasneci G, Weikum G. YAGO: A core of semantic knowledge//*Proceedings of the 16th International Conference on World Wide Web*. Alberta, Canada, 2007:697-706
- [4] Vrande? ić D, Krötzsch M. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 2014, 57(10): 78-85
- [5] Diefenbach D, Lopez V, Singh K, et al. Core techniques of question answering systems over knowledge bases: A survey. *Knowledge and Information Systems*, 2018, 55(3): 529-569
- [6] Unger C, Freitas A, Cimiano P. An introduction to question answering over linked data//*Proceedings of the Reasoning Web International Summer School*. Athens, Greece, 2014: 100-140
- [7] Chakraborty N, Lukovnikov D, Maheshwari G, et al. Introduction to neural network based approaches for question answering over knowledge graphs. *arXiv preprint arXiv:1907.09361*, 2019
- [8] Liu Kang, Zhang Yuan-Zhe, Ji Guo-Liang, et al. Representation learning for question answering over knowledge base: An overview. *Acta Automatica Sinica*, 2016, 42(6): 807-818 (In Chinese)  
(刘康, 张元哲, 纪国良, 等. 基于表示学习的知识库问答研究进展与展望. *自动化学报*, 2016, 42(6): 807-818)
- [9] Devlin J, Chang M W, Lee K, et al. 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, Volume 1 (Long and Short Papers)*. Minneapolis, USA, 2019: 4171-4186
- [10] Hemphill C T, Godfrey J J, Doddington G R. The ATIS spoken language systems pilot corpus//*Proceedings of the Speech and Natural Language Workshop*. Pennsylvania, USA, 1990: 96-101
- [11] Tang L R, Mooney R J. Using multiple clause constructors in inductive logic programming for semantic parsing//*Proceedings of the European Conference on Machine Learning*. Berlin, Germany, 2001: 466-477
- [12] Cimiano P, Lopez V, Unger C, et al. Multilingual question answering over linked data (QALD-3): Lab overview//*Proceedings of the international conference of the cross-language evaluation forum for european languages*. Berlin, Germany, 2013: 321-332
- [13] Unger C, Forascu C, Lopez V, et al. Question answering over linked data (QALD-4)//*Proceedings of the Working Notes for CLEF 2014 Conference*. Sheffield, UK, 2014: 1172-1180
- [14] Unger C, Forascu C, Lopez V, et al. Answering over linked data (QALD-5)//*Proceedings of the Working notes for CLEF 2015 Conference*. Toulouse, France, 2015: 23-24
- [15] Unger C, Ngomo A C N, Cabrio E. 6th open challenge on question answering over linked data (QALD-6)//*Proceedings of the Semantic Web Evaluation Challenge*. Heraklion, Greece, 2016: 171-177
- [16] Usbeck R, Ngomo A C N, Haarmann B, et al. 7th open challenge on question answering over linked data (QALD-7)//*Proceedings of the Semantic Web Evaluation Challenge*. Portoroz, Slovenia, 2017: 59-69
- [17] Usbeck R, Ngomo A C N, Conrads F, et al. 8th challenge on question answering over linked data (QALD-8)//*Proceedings of the International Semantic Web Conference*. Monterey, USA, 2018: 51-57
- [18] Usbeck Ricardo, Gusmita Ria, Saleem Muhammad, et al. 9th challenge on question answering over linked data (QALD-9)//*Proceedings of the International Semantic Web Conference*. Monterey, USA, 2018: 58-64
- [19] Cai Q, Yates A. Large-scale semantic parsing via schema matching and lexicon extension//*Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Sofia, Bulgaria, 2013: 423-433
- [20] Bordes A, Usunier N, Chopra S, et al. Large-scale simple question answering with memory networks. *arXiv preprint arXiv:1506.02075*, 2015
- [21] Trivedi P, Maheshwari G, Dubey M, et al. Lc-quad: A corpus for complex question answering over knowledge graphs//*Proceedings of the International Semantic Web Conference*. Vienna, Austria, 2017: 210-218
- [22] Shi J, Cao S, Pan L, et al. KQA Pro: A large diagnostic dataset for complex question answering over knowledge base. *arXiv preprint arXiv:2007.03875*, 2020
- [23] Berant J, Chou A, Frostig R, et al. Semantic parsing on freebase from question-answer pairs//*Proceedings of the 2013 conference on empirical methods in natural language processing*. Seattle, USA, 2013: 1533-1544
- [24] Yih W, Richardson M, Meek C, et al. The value of semantic parse labeling for knowledge base question answering//*Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Berlin, Germany, 2016: 201-206
- [25] Bao J, Duan N, Yan Z, et al. Constraint-based question answering with knowledge graph//*Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan, 2016: 2503-2514
- [26] Su Y, Sun H, Sadler B, et al. On generating characteristic-rich question sets for qa evaluation//*Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, USA, 2016: 562-572
- [27] Miller A, Fisch A, Dodge J, et al. Key-value memory net-

- works for directly reading documents//Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Austin, USA, 2016: 1400-1409
- [28] Zhang Y, Dai H, Kozareva Z, et al. Variational reasoning for question answering with knowledge graph//Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence. New Orleans, USA, 2018: 6069-6076
- [29] Dubey M, Banerjee D, Abdelkawi A, et al. Lc-quad 2.0: A large dataset for complex question answering over wikidata and dbpedia//Proceedings of the International Semantic Web Conference. Auckland, New Zealand, 2019: 69-78
- [30] Dodge J, Gan E, Zhang X, et al. Evaluating prerequisite qualities for learning end-to-end dialog systems. arXiv preprint arXiv:1511.06931, 2015
- [31] Hartmann A K, Tommaso M E, Moussallem D, et al. Generating a large dataset for neural question answering over the dbpedia knowledge base//Proceedings of the Workshop on Linked Data Management. Vienna, Austria, 2018: 1-8
- [32] Talmor A, Berant J. The web as a knowledge-base for answering complex questions//Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics; Human Language Technologies, Volume 1 (Long Papers). New Orleans, USA, 2018: 641-651
- [33] Azmy M, Shi P, Lin J, et al. Farewell freebase: Migrating the simplequestions dataset to dbpedia//Proceedings of the 27th International Conference on Computational Linguistics. Santa Fe, USA, 2018: 2093-2103
- [34] Saxena A, Tripathi A, Talukdar P. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 4498-4507
- [35] Sun H, Bedrax-Weiss T, Cohen W. PullNet: Open domain question answering with iterative retrieval on knowledge bases and text//Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong, China, 2019: 2380-2390
- [36] Saha A, Pahuja V, Khapra M M, et al. Complex sequential question answering: Towards learning to converse over linked question answer pairs with a knowledge graph//Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence. New Orleans, USA, 2018: 705-713
- [37] Iyyer M, Yih W, Chang M W. Search-based neural structured learning for sequential question answering//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver, Canada, 2017: 1821-1831
- [38] Yin W, Yu M, Xiang B, et al. Simple question answering by attentive convolutional neural network//Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers. Osaka, Japan, 2016: 1746-1756
- [39] Zhao W, Chung T, Goyal A, et al. Simple question answering with subgraph ranking and joint-scoring//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). Minneapolis, USA, 2019: 324-334
- [40] Hao Y, Liu H, He S, et al. Pattern-revising enhanced simple question answering over knowledge bases//Proceedings of the 27th International Conference on Computational Linguistics. Santa Fe, USA, 2018: 3272-3282
- [41] Yu M, Yin W, Hasan K S, et al. Improved neural relation detection for knowledge base question answering//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver, Canada, 2017: 571-581
- [42] Türe F, Jovic O. No need to pay attention: Simple recurrent neural networks work! //Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen, Denmark, 2017: 2866-2872
- [43] Hao Y, Zhang Y, Liu K, et al. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver, Canada, 2017: 221-231
- [44] Dong L, Wei F, Zhou M, et al. Question answering over freebase with multi-column 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 (Volume 1: Long Papers). Beijing, China, 2015: 260-269
- [45] Yao X, Van Durme B. Information extraction over structured data: Question answering with freebase//Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Baltimore, USA, 2014: 956-966
- [46] Bordes A, Chopra S, Weston J. Question answering with subgraph embeddings//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar, 2014: 615-620
- [47] Lukovnikov D, Fischer A, Lehmann J, et al. Neural network-based question answering over knowledge graphs on word and character level//Proceedings of the 26th international conference on World Wide Web. Perth, Australia, 2017: 1211-1220
- [48] Gupta V, Chinnakotla M, Shrivastava M. Retrieve and re-rank: A simple and effective IR approach to simple question answering over knowledge graphs//Proceedings of the First Workshop on Fact Extraction and VERification (FEVER). Brussels, Belgium, 2018: 22-27
- [49] He X, Golub D. Character-level question answering with at-



- tention//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Austin, USA, 2016: 1598-1607
- [50] Qiu Y, Li M, Wang Y, et al. Hierarchical type constrained topic entity detection for knowledge base question answering//Proceedings of the Web Conference 2018. Lyon, France, 2018: 35-36
- [51] Mohammed S, Shi P, Lin J. Strong baselines for simple question answering over knowledge graphs with and without neural networks//Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers). New Orleans, USA, 2018: 291-296
- [52] Dai Z, Li L, Xu W. CFO: Conditional focused neural question answering with large-scale knowledge bases//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Berlin, Germany, 2016: 800-810
- [53] Petrochuk M, Zettlemoyer L. Simplequestions nearly solved: A new upperbound and baseline approach//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Brussels, Belgium, 2018: 554-558
- [54] Bordes A, Weston J, Usunier N. Open question answering with weakly supervised embedding models//Proceedings of the joint European conference on machine learning and knowledge discovery in databases. Berlin, Germany, 2014: 165-180
- [55] Huang X, Zhang J, Li D, et al. Knowledge graph embedding based question answering//Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. Melbourne, Australia, 2019: 105-113
- [56] Bordes A, Usunier N, Garcia-Duran A, et al. Translating embeddings for modeling multi-relational data//Proceedings of the Advances in Neural Information Processing Systems. Lake Tahoe, USA, 2013: 2787-2795
- [57] Sidiropoulos G, Voskarides N, Kanoulas E. Knowledge graph simple question answering for unseen domains. arXiv preprint arXiv:2005.12040, 2020
- [58] Yu Y, Hasan K S, Yu M, et al. Knowledge base relation detection via multi-view matching//Proceedings of the European Conference on Advances in Databases and Information Systems. Budapest, Hungary, 2018: 286-294
- [59] Zhang Y, Xu G, Fu X, et al. Adversarial training improved multi-path multi-scale relation detector for knowledge base question answering. IEEE Access, 2020, 8: 63310-63319
- [60] Yavuz S, Gur I, Su Y, et al. Recovering question answering errors via query revision//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen, Denmark, 2017: 903-909
- [61] Wang Y, Zhang R, Xu C, et al. The APVA-TURBO approach to question answering in knowledge base//Proceedings of the 27th International Conference on Computational Linguistics. Santa Fe, USA, 2018: 1998-2009
- [62] Kate R J, Wong Y W, Mooney R J. Learning to transform natural to formal languages//Proceedings of the Twentieth AAAI Conference on Artificial Intelligence. Pittsburgh, USA, 2005: 1062-1068
- [63] Lu W, Ng H T, Lee W S, et al. A generative model for parsing natural language to meaning representations//Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing. Honolulu, USA, 2008: 783-792
- [64] Kwiatkowski T, Zettlemoyer L, Goldwater S, et al. Lexical generalization in CCG grammar induction for semantic parsing//Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing. Edinburgh, Scotland, UK, 2011: 1512-1523
- [65] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks//Proceedings of the Advances Neural Information Processing Systems. Montreal, Canada, 2014: 3104-3112
- [66] Dong L, Lapata M. Language to logical form with neural attention//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Berlin, Germany, 2016: 33-43
- [67] Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation, 1997, 9(8): 1735-1780
- [68] Dong L, Lapata M. Coarse-to-fine decoding for neural semantic parsing//Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Melbourne, Australia, 2018: 731-742
- [69] Zhang H, Cai J, Xu J, et al. Complex question decomposition for semantic parsing//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, Italy, 2019: 4477-4486
- [70] Sun Y, Zhang L, Cheng G, et al. SPARQA: Skeleton-based semantic parsing for complex questions over knowledge bases//Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence. New York, USA, 2020: 8952-8959
- [71] Ding J, Hu W, Xu Q, et al. Leveraging frequent query substructures to generate formal queries for complex 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). Hong Kong, China, 2019: 2614-2622
- [72] Dong L, Quirk C, Lapata M. Confidence modeling for neural semantic parsing//Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Melbourne, Australia, 2018: 743-753
- [73] Shao B, Gong Y, Qi W, et al. Graph-based transformer with cross-candidate verification for semantic parsing//Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence. New York, USA, 2020: 8807-8814
- [74] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need//Proceedings of the Advances in Neural Information

- Processing Systems. Long Beach, USA, 2017; 5998-6008
- [75] Zhang X, Zou L, Hu S. An interactive mechanism to improve question answering systems via feedback//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing, China, 2019; 1381-1390
- [76] Herzig J, Berant J. Neural semantic parsing over multiple knowledge-bases//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Vancouver, Canada, 2017; 623-628
- [77] Yih W, Chang M W, He X, et al. Semantic parsing via staged query graph generation: Question answering with knowledge base//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). Beijing, China, 2015; 1321-1331
- [78] Hu S, Zou L, Zhang X. A state-transition framework to answer complex questions over knowledge base//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Brussels, Belgium, 2018; 2098-2108
- [79] Luo K, Lin F, Luo X, et al. Knowledge base question answering via encoding of complex query graphs//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Brussels, Belgium, 2018; 2185-2194
- [80] Sorokin D, Gurevych I. Modeling semantics with gated graph neural networks for knowledge base question answering//Proceedings of the 27th International Conference on Computational Linguistics. Santa Fe, USA, 2018; 3306-3317
- [81] Bhutani N, Zheng X, Jagadish H V. Learning to answer complex questions over knowledge bases with query composition//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing, China, 2019; 739-748
- [82] Reddy S, Täckström O, Collins M, et al. Transforming dependency structures to logical forms for semantic parsing. Transactions of the Association for Computational Linguistics, 2016, 4: 127-140
- [83] Reddy S, Täckström O, Petrov S, et al. Universal semantic parsing//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Copenhagen, Denmark, 2017; 89-101
- [84] Pasupat P, Liang P. Inferring logical forms from denotations//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Berlin, Germany, 2016; 23-32
- [85] Misra D, Chang M W, He X, et al. Policy shaping and generalized update equations for semantic parsing from denotations//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Brussels, Belgium, 2018; 2442-2452
- [86] Dasigi P, Gardner M, Murty S, et al. Iterative search for weakly supervised semantic parsing//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). Minneapolis, USA, 2019; 2669-2680
- [87] Cheng J, Reddy S, Saraswat V, et al. Learning structured natural language representations for semantic parsing//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver, Canada, 2017; 44-55
- [88] Dyer C, Ballesteros M, Ling W, et al. Transition-based dependency parsing with stack long short-term memory//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). Beijing, China, 2015; 334-343
- [89] Liang C, Berant J, Le Q, et al. Neural symbolic machines: Learning semantic parsers on freebase with weak supervision//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver, Canada, 2017; 23-33
- [90] Guo D, Tang D, Duan N, et al. Dialog-to-action: conversational question answering over a large-scale knowledge base//Proceedings of the Advances in Neural Information Processing Systems. Montreal, Canada, 2018; 2942-2951
- [91] Ansari G A, Saha A, Kumar V, et al. Neural program induction for kbqa without gold programs or query annotations//Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence. Macao, China, 2019; 4890-4896
- [92] Saha A, Ansari G A, Laddha A, et al. Complex program induction for querying knowledge bases in the absence of gold programs. Transactions of the Association for Computational Linguistics, 2019, 7: 185-200
- [93] Agrawal P, Dalmia A, Jain P, et al. Unified semantic parsing with weak supervision//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, Italy, 2019; 4801-4810
- [94] Zhou M, Huang M, Zhu X. An interpretable reasoning network for multi-relation question answering//Proceedings of the 27th International Conference on Computational Linguistics. Santa Fe, USA, 2018; 2010-2022
- [95] Chen Z Y, Chang C H, Chen Y P, et al. UHop: An unrestricted-hop relation extraction framework for knowledge-based question answering//Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Minneapolis, USA, 2019; 345-356
- [96] Qiu Y, Wang Y, Jin X, et al. Stepwise reasoning for multi-relation question answering over knowledge graph with weak supervision//Proceedings of the 13th International Conference on Web Search and Data Mining. Houston, USA, 2020; 474-482

- [97] Sukhbaatar S, Weston J, Fergus R. End-to-end memory networks//Proceedings of the Advances in Neural Information Processing Systems. Montreal, Canada, 2015: 2440-2448
- [98] Xu K, Lai Y, Feng Y, et al. Enhancing key-value memory neural networks for knowledge based question answering//Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics; Human Language Technologies, Volume 1 (Long and Short Papers). Minneapolis, USA, 2019: 2937-2947
- [99] Chen Y, Wu L, Zaki M J. Bidirectional attentive memory networks for question answering over knowledge bases//Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics; Human Language Technologies, Volume 1 (Long and Short Papers). Minneapolis, USA, 2019: 2913-2923
- [100] Cohen W W, Sieglar M, Hofer A. Neural query language: A knowledge base query language for tensorflow. arXiv preprint arXiv:1905.06209, 2019
- [101] Cohen W W, Sun H, Hofer R A, et al. Scalable neural methods for reasoning with a symbolic knowledge base. arXiv preprint arXiv:2002.06115, 2020
- [102] Sun H, Arnold A O, Bedrax-Weiss T, et al. Guessing what's plausible but remembering what's true: Accurate neural reasoning for question-answering. arXiv preprint arXiv:2004.03658, 2020
- [103] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907, 2016
- [104] Velić ković P, Cucurull G, Casanova A, et al. Graph attention networks. arXiv preprint arXiv:1710.10903, 2017
- [105] Sun H, Dhingra B, Zaheer M, et al. Open domain question answering using early fusion of knowledge bases and text//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Brussels, Belgium, 2018: 4231-4242
- [106] Vakulenko S, Fernandez Garcia J D, Polleres A, et al. Message passing for complex question answering over knowledge graphs//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing, China, 2019: 1431-1440
- [107] Xu K, Reddy S, Feng Y, et al. Question answering on free-base via relation extraction and textual evidence//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Berlin, Germany, 2016: 2326-2336
- [108] Ren H, Hu W, Leskovec J. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. arXiv preprint arXiv:2002.05969, 2020
- [109] Rajpurkar P, Zhang J, Lopyrev K, et al. SQuAD: 100,000+ questions for machine comprehension of text//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Austin, USA, 2016: 2383-2392
- [110] Andreas J, Rohrbach M, Darrell T, et al. Neural module networks//Proceedings of the IEEE conference on computer vision and pattern recognition. Las Vegas, USA, 2016: 39-48
- [111] Gupta N, Lin K, Roth D, et al. Neural module networks for reasoning over text. arXiv preprint arXiv:1912.04971, 2019
- [112] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks//Proceedings of the Advances in Neural Information Processing Systems. Lake Tahoe, USA, 2012: 1097-1105
- [113] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition//Proceedings of the IEEE conference on computer vision and pattern recognition. Las Vegas, USA, 2016: 770-778
- [114] Wang P, Wu Q, Shen C, et al. Explicit knowledge-based reasoning for visual question answering//Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence. Melbourne, Australia, 2017: 1290-1296
- [115] Narasimhan M, Lazebnik S, Schwing A. Out of the box: Reasoning with graph convolution nets for factual visual question answering//Proceedings of the Advances in Neural Information Processing Systems. Montreal, Canada, 2018: 2654-2665



**CAO Shu-Lin**, Ph. D. candidate.

Her research interests mainly include knowledge graph representation and knowledge base reasoning.

**SHI Jia-Xin**, Ph. D. His research interests mainly include knowledge base reasoning, visual reasoning, question

answering, and document summarization.

**HOU Lei**, Ph. D. , assistant professor. His research interests mainly include knowledge graph and news mining.

**LI Juan-Zi**, Ph. D. , professor. Her research interests mainly include semantic web and social web mining.

## Background

As the scale of the Knowledge Bases (KBs) grows, people are paying more attention to seeking effective methods for

accessing these precious intellectual resources. Initially, accessing the knowledge stored in KBs is facilitated through the

use of formal query languages with a well-defined syntax, such as SQL, SPARQL, GraphQL etc, due to their expressive nature and the ability to store and retrieve very nuanced information. However, the use of formal queries to access these knowledge bases pose difficulties for non-expert users because they require users to understand the syntax of the formal query language and the underlying structure of entities and their relationships. KBQA systems therefore aim to provide users with an interface to ask questions in natural language and receive a concise answer generated by querying the KB. Such KBQA systems could be integrated in popular web search engines like Google Search and Bing as well as in conversational assistants including Google Assistant, Siri, and Alexa. However, how to understand questions and bridge the gap between natural languages and structured semantics of knowledge bases is very challenging. In the early research, researchers mainly focused on simple questions, which involve only a single triple. In recent years, with the successful application of deep learning, simple questions have been nearly tackled and the researchers gradually focus more on questions that require complex reasoning skills. In this article, we will review the current re-

search progress of KBQA. For both simple question answering and complex question answering, we summarize the problems and challenges of the task. Then multi-dimensional analysis and comparison of KBQA datasets in recent years are carried out. Next, we introduce the representative methods of simple question answering and complex question answering and analyze their strengths and shortcomings. Finally, we discuss the future research direction of KBQA.

This research is supported by National Key Research and Development Program of China (Grant No. 2017YFB1002101) and NSFC Key Projects (Grant No. U173620 and Grant No. 61533018), which aim to (1) research methods of knowledge acquisition combining representation learning and symbolic logic, (2) support knowledge services for character portraits and positioning analysis, (3) establish basic knowledge facilities for open-domain text computing, (4) integrate logical reasoning and representation learning to support new knowledge discovery and (5) establish a Chinese open-domain question answering system. Our group has been working on this area for several years, published several papers and released a KBQA dataset.