DOI:10.13530/j.cnki.jlis.150030

面向图书馆关联数据的自动问答技术研究*

欧石燕　唐振贵

摘　要　早期针对语义网的自动问答主要是面向单一 RDF 数据集,随着网络上相互关联数据集的急速增加,迫切需要将自动问答扩展到多个 RDF 数据集,但同时在语义标注、答案整合方面也带来了更大的难度与挑战。本文提出了一种面向图书馆关联数据的自动问答新方法,通过将自然语言提问转换为结构化的 SPARQL 查询,从图书馆领域相互关联的五个 RDF 数据集中提取特定答案。该方法的创新点在于,将问句分为涉及一个数据集的简单句和涉及多个数据集的复杂句分别进行处理,又将简单句分为查询属性和查询实例两种类别分别制定 SPARQL 查询构建规则,将复杂句分解成若干个简单句进行处理,有利于 SPARQL 查询的构建和答案的整合。通过实验测评,100 个问句的回答精确率达到 91%,表明这是一种行之有效的问答方法,对于促进关联数据在图书馆中的应用具有重要意义。图 5。表 5。参考文献 27。

关键词　自动问答　关联数据　RDF 数据集　SPARQL 查询　语义标注　本体

分类号　G254

A Question Answering Method over Library Linked Data

OU Shiyan & TANG Zhengui

ABSTRACT

Since the advent of Linked Data, more and more structured data have been published on the Web in Linked Data format, including a large amount of bibliographic data, academic information and controlled vocabularies from libraries and other related institutions. Therefore, the issue of how to effectively access these interlinked RDF data becomes of crucial importance. SPARQL provides a standard way to query RDF data; however, it is very difficult for ordinary users to construct SPARQL queries. Question answering, which can provide an easy-to-use natural language interface, is undoubtedly an ideal solution. Earlier question answering research on the Semantic Web is oriented to a single RDF dataset. With the growth of interlinked RDF datasets on the Web, there is an urgent need to extend question answering from a single RDF dataset to multiple RDF datasets, which thus causes more problems and challenges in semantic annotation and answer integration.

This paper proposes a novel question answering method over Library Linked Data, which transforms a natural language question into a structured SPARQL query to retrieve answers from five interlinked RDF datasets in libraries, including bibliographic data, thesauri, events, people/organizations and locations. The

* 本文系国家社科基金项目"基于 SOA 架构的术语注册和服务系统构建与应用研究"(编号:11BT0023)的研究成果之一。(This article is an outcome of the project "Research on the construction and application of SOA-based terminology registry and terminology services" (No. 11BT0023) supported by National Social Science Foundation of China.)

通信作者:欧石燕,Email:oushiyan@nju.edu.cn,ORCID:0000-0001-8617-6987 (Correspondence should be addressed to OU Shiyan, Email:oushiyan@nju.edu.cn, ORCID:0000-0001-8617-6987)

question answering procedure includes three main steps: 1) Index construction: extract instance names (i. e. named entities) from RDF data and the lexical labels of ontology classes and properties from OWL files, and offline construct two indexes (one for named entities and one for ontology terms) using the open source information retrieval toolkit LUCENE; 2) Question preprocessing: perform Chinese word segmentation, named entity recognition, and semantic annotation based on the constructed indexes, categorize questions into two categories, i. e. simple questions involving a single RDF dataset and complex questions involving multiple RDF datasets, according to the number of the involved ontologies and the number of the classes and their relationships, and furthermore categorize simple questions into two types, i. e. the A type querying attributes and the B type querying names; 3) Question answering: for a simple question, construct a SPARQL query based on the pre-defined rules; for a complex question, decompose it into several simple sub-questions, process each sub-question using the simple question method, and then combine the results of the sub-questions to construct a SPARQL query for the whole complex question.

The innovation of this proposed question answering method lies in transforming question answering over multiple RDF datasets into the one over a single RDF dataset in order to facilitate the construction of SPARQL queries and answer integration, by decomposing a complex question into several simple questions based on its dependency parsing result. The experiment results show that this is an effective question answering method which greatly simplifies the processing of complex questions and obtains an answer accuracy of 88% for complex questions and 91% for both simple and complex questions. However, this method can only be used to answer the questions which are stated explicitly in RDF datasets, and is not able to answer the questions which require reasoning and computing, for example, those containing "more" and "the most".

Question answering provides a straightforward and easy-to-use manner of accessing Linked Data. It is a key step in the application of Linked Data in the real world. Thus, the research content of this paper has a very significant value to facilitate the application of Linked Data in libraries. It is an earlier study about Chinese question answering over Linked Data, and also an earlier study focusing on Library Linked Data. 5 figs. 5 tabs. 27 refs.

KEY WORDS

Question answering. Linked Data. RDF dataset. SPARQL query. Semantic annotation. Ontology.

0 引言

自2006年伯纳斯·李首次提出"关联数据"以来[1]，作为一种在网络上发布结构化数据的方式，关联数据受到学术界和企业界的极大关注，越来越多的机构开始在网络上发布自己的关联数据集。关联开放数据云(Linked Open Data Cloud)已由2007年的12个RDF数据集发展到现今的近600个，呈井喷式增长[2]。图书馆拥有且一直持续不断地生成大量高质量的结构化数据，是关联数据的天然实践者与提供者。目前，图书馆数据已经成为关联数据云的一个重要来源，大量书目数据(如LIBRIS、WorldCat)、词表数据(如LCSH、AGROVOC)和学术论文数据(如DBLP、CiteSeer)被发布为关联数据[3]，约占整个关联数据云的9.5%[2]。

随着关联数据在网络上不断增多，如何直接、有效地查询和访问这些结构化数据成为亟须解决的问题。基于RDF数据模型的关联数据需采用SPARQL语言才能进行查询，要求用户既要了解底层数据所使用的描述词汇和结构，

又要具备构建复杂 SPARQL 查询的能力，这对于普通用户来说是极为困难的。近年来，研究者一直在探索新的关联数据查询方式，使用户既能够受益于语义网出色的表达能力，同时又能屏蔽掉其复杂性。自动问答，能够提供简单友好的自然语言查询界面，无疑是一种理想的解决方案，因而受到了极大关注[4]。致力于信息访问系统开发与评测的“评估论坛会议和实验室（Conference and Labs of the Evaluation Forum，CLEF）”自 2011 年起每年都开设“关联数据自动问答（Question Answering over Linked Data，QALD）”专题①，就面向关联数据的自动问答展开评比与讨论。自动问答技术为关联数据的访问提供了一种直接、易用的查询模式，是关联数据走向实际应用的一个重要环节。

在自动问答的发展历史中，一直存在着两种几乎平行发展的系统：基于自由文本的问答系统和基于知识的问答系统[5]。前者主要利用信息检索技术或搜索引擎从文本集合中检索到相关网页，并从中提取出答案；后者则直接从结构化数据中抽取答案，因此可用于处理更复杂的提问，并获得更精确的答案[5]。最早的基于知识的问答系统可追溯到 20 世纪 70 年代面向关系型数据库的自然语言查询[6]，如 LUNAR[7]、PRECISE[8]等，后来发展到基于专属知识库的开放域自动问答，如 START②[9]、Wolfram Alpha③、Evi④等。2007 年以来，随着语义网的出现与发展，面向 RDF 数据的语义问答逐渐成为知识问答的主流。起初，有关这类自动问答的研究主要是针对单一 RDF 数据集，如 AquaLog[10]、PANTO[11]、QACID[12]等，采用的方法是 SPARQL 查询构建法，即对用户问句进行分析与处理，将其转换为本体兼容的 SPARQL 查询，用于从 RDF 数据集中提取答案。随着关联数据的急速发展，迫切需要将面向单一 RDF 数据集的自动问答扩展到相互关联的多个 RDF 数据集，这也带来了更大的难度与挑战，主要在于：①用户提问可涉及多个数据集，使其在句法和语义上都变得更加复杂；②在对问句进行语义标注时，需将问句中的自然语言词汇与多个本体中的语义元素相映射，使得标注时的歧义问题更加突出，标注难度增大；③需从不同数据集中提取信息并进行整合才能形成最终答案。当前，针对多个 RDF 数据集自动问答的研究还比较少，仅有的一些研究也主要是解决多数据集（即多本体）环境下的问句语义标注问题，如 PowerAqua[13]，以及 He[14]、Shekarpour[15]等人的研究，关于复杂 SPARQL 查询的构建以及答案整合的研究几乎没有。

本研究提出了一种新的面向图书馆关联数据的自动问答方法，面对图书馆中相互关联的多个 RDF 数据集（即图书馆书目数据、叙词表数据、事件数据、个人/组织机构数据和地点数据），通过采用 SPARQL 查询构建的方法实现自动问答。该方法的创新之处在于，采用问句分解的方法，将一个涉及多个数据集（本体）的复杂问句分解成若干个只涉及单一数据集的简单问句，从而将针对多个数据集的自动问答转换为针对单一数据集的自动问答，有利于 SPARQL 查询的构建和答案的整合。

1 相关研究综述

根据数据源的多少，面向 RDF 数据集的自动问答可分为三种类型：①面向单一 RDF 数据集；②面向多个相互关联的 RDF 数据集；③面向整个数据网（即关联数据云）。第一类问答系统通常只适用于受限领域，以单一本体作为领域知识模型，从单一数据集中抽取答案。这类系

① http://greententacle.techfak.uni-bielefeld.de/~cunger/qald

② http://start.csail.mit.edu/index.php

③ http://www.wolframalpha.com

④ 原 True Knowledge，见 http://www.evi.com

统的关键是利用本体分析和解释用户提问，并将其转换为本体兼容的结构化查询。第二类系统是真正面向关联数据的自动问答，同第一类系统一样，其关键点也是由自然语言提问向结构化查询的转化，但这类系统因涉及多个采用不同模式（schema）的 RDF 数据集，需同时利用模式和实例层面的链接构建一个跨越不同数据集的联合查询，因此这一过程变得更加复杂和困难。第三类问答系统事实上还几乎不存在，目前面向整个关联数据云的检索系统主要还是基于关键词的语义网搜索引擎，如 Swoogle[16] 和 Sindice[17]。因此，我们只对前两种自动问答方法进行总结与述评。在 RDF 数据自动问答中，主流方法是通过各种手段将自然语言问句转换为结构化的 SPARQL 查询，我们将这些方法概括为如下六类。

最简单的一类方法是构建受控用户问句，譬如 GINSENG[18]。问答系统提供指导性用户界面，帮助用户采用受控词汇构建自然语言的用户提问，这种在受控条件下生成的问句，其结构和语义规范而合理，很容易处理并转换为结构化的 SPARQL 查询。但该类方法的缺陷是所支持的问句表达受限，灵活性不够。

第二类方法是直接利用本体对问句进行语义标注，譬如 ORAKEL[19] 和 Pythia[20]。问答系统直接利用通用词汇和领域相关词汇（即本体中的元素）对自然语言提问进行语义标注，将其转换为逻辑表示，然后再通过深层次的语言处理将逻辑表示转换为结构化查询。该类方法没有独立的语言分析层，可移植性差；但因为是将问句中的词汇直接与本体中的元素相映射，映射精确度较高，再加上深层次的语言分析，能够回答复杂的提问。

在上述方法之上的一类改进方法是增加领域无关的通用语言处理层，譬如 AqualLog[10] 和 PANTO[11]。该类方法首先采用通用的语言分析工具对自然语言提问进行分析和处理，将其转换为通用的、领域无关的结构化语言表示，然后利用词汇资源（如 WordNet）将语言表示中的自然语言词汇与本体中的元素相映射，将语言表示转化为本体兼容的表示，最后生成 SPARQL 查询。该类方法拥有领域无关的语言分析层，因此具有一定的可移植性。

第四类方法是采用模式识别将问句与问题模板相匹配，譬如 NLP-reduce[21]、Querix[22] 和 QACID[12]。这类方法预先构建三元组模板或问句模板，每个模板与特定的 SPARQL 查询模板相关联，然后将用户问句与三元组模板或问句模板进行匹配，基于匹配结果生成 SPARQL 查询。该类方法的优点是避免了复杂的问句语言处理过程，但所能回答的问题受模板的局限很大。

第五类方法是直接构建 SPARQL 查询，譬如 LODQA[23] 和 TBSL[24]。这类方法通过对用户问句进行语言分析和处理（如句法解析），生成 SPARQL 查询模板，然后再从问句中抽取语义实体填充查询模板，生成完整的 SPARQL 查询。其优点是适用于难以直接转换为三元组的问句（如含有大于、最多等比较词汇的问句），缺点是 SPARQL 查询模板与问句的句法结构密切相关，一旦问句结构与知识库（或本体）结构不相匹配，容易产生错误。

最后一类方法是通过人机对话对用户提问进行辨析，譬如 FREyA[25]。当用户提问中出现意义模糊不清的地方时，问答系统通过对话或者其他用户交互方式来澄清用户意图。实验证明，用户交互方式能够提高问答系统的准确率和召回率，但其缺点是：新用户往往因为对底层数据模型和本体词汇不了解而无法对歧义消解给予帮助，而且随着数据源的增多，向用户给出建议的匹配并对其进行排序变得更加复杂。

上述研究中，第二与第三类方法使用比较普遍，主要是针对单一 RDF 数据集。我们采用的问答方法属于第二类，但针对的是多个 RDF 数据集，研究重点是如何将语义标注后的问句转换为结构化查询，尤其是对于涉及多个数据集的复杂问句。

虽然国外对关联数据自动问答技术的研究如火如荼，但国内这方面的研究还非常少。国

内关于关联数据中文自动问答的研究，仅见许德山等人在《中文问句与 RDF 三元组映射方法研究》中提出中文问句与 RDF 三元组的映射方法[26]。因此，本研究是较早关于关联数据中文自动问答的研究，目前看来也是较早以图书馆关联数据为对象的自动问答研究。

2 系统概览

本研究中构建的面向图书馆关联数据的自动问答系统架构如图 1 所示，该系统主要包含四个部分。

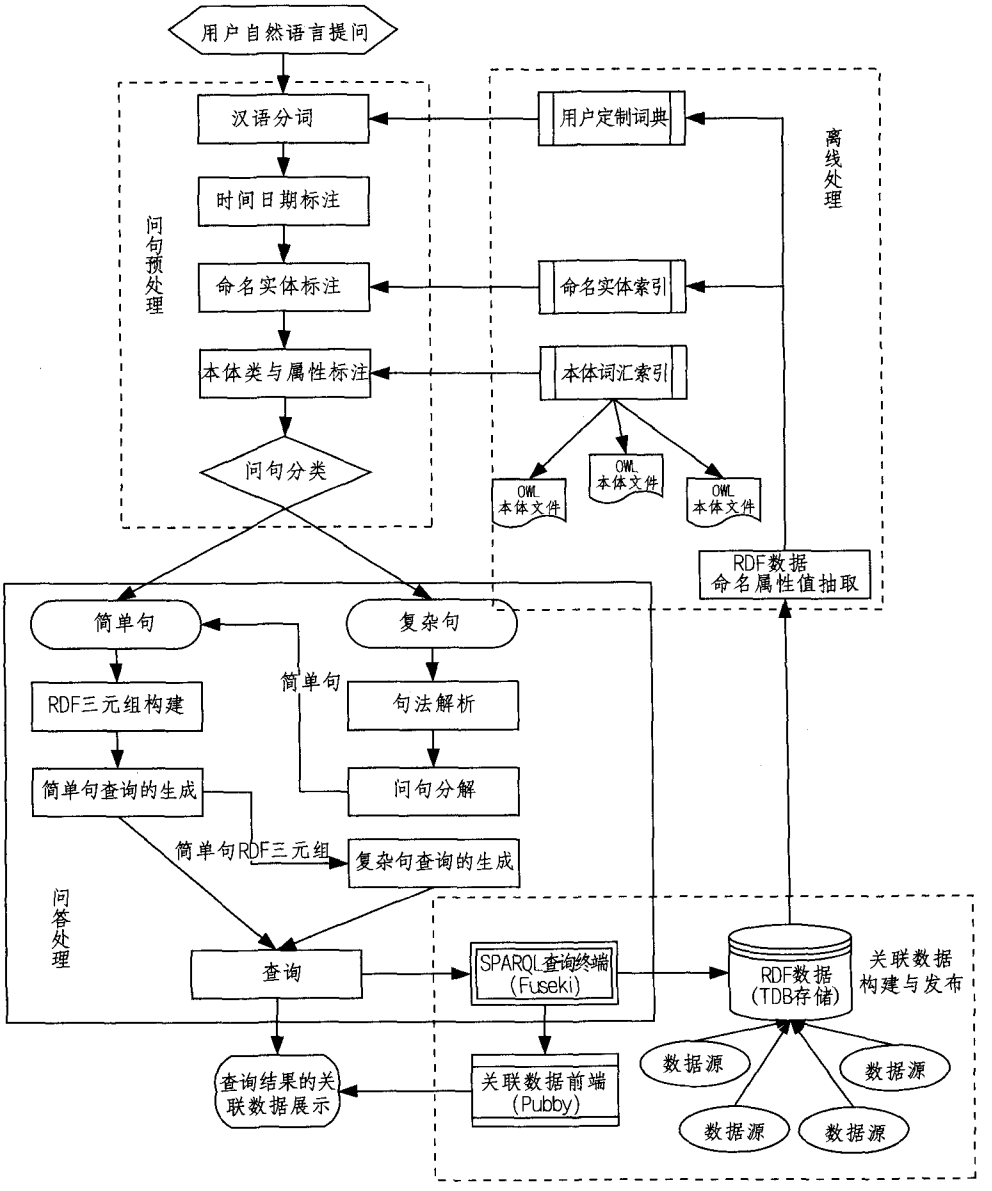

图 1 自动问答系统架构图

(1)图书馆关联数据的构建与发布:图书馆关联数据是通过对文献资源及其相关资源(如受控词汇、作者、地点、事件等)进行语义描述并建立语义关系而形成的五个相互关联的 RDF 数据集,每个数据集都基于一个本体进行描述[3]。上述 RDF 数据采用三元组存储器 Jena TDB 进行存储,由 Jena 的 SPARQL 查询终端 Fuseki 提供访问接口,最后通过 Pubby 发布为网络可访问的关联数据[27]。

(2)索引的构建:在自动问答中,一个重要步骤是采用本体中的元素对问句中的自然语言词汇进行语义标注,从而为构建本体兼容的 SPARQL 查询做准备。在标注过程中,为了提高对本体元素的检索速度,我们事先对其建立索引。首先从 RDF 数据中抽取出本体实例(即命名实体),从本体 OWL 文件中抽取出本体词汇(即类与属性);然后采用开源信息检索工具包 LUCENE 构建命名实体与本体词汇的倒排索引。

(3)问句预处理:问句预处理包括对自然语言提问进行汉语分词和命名实体识别,并基于构建的索引对问句进行语义标注,然后对标注好的问句进行分类。根据问句中所涉及的本体数量和本体中类的数量及关系远近,将问句分为简单句和复杂句。简单句只涉及单一本体中的一个类,而复杂句则涉及一个或多个本体中的多个类。

(4)问答处理:这部分是整个问答系统的核心。对于简单句,直接根据规则构建 SPARQL 查询;对于复杂句,则根据句法解析结果将其分解为若干个简单句,然后按照简单句方式处理每个切分出来的子句,最后将子句的处理结果合并后生成复杂句的 SPARQL 查询。

在下文中,我们将对后三个部分进行详细描述,关于第一部分"图书馆关联数据的构建与发布"可参见相关研究论文[3,27]。

3 索引构建

在对用户提问进行语义标注时,需将问句中的词汇与 RDF 数据中的实例和本体中的词汇进行匹配,为了加快检索速度,我们采用开源信息检索工具 LUCENE,以离线方式分别构建命名实体索引和本体词汇索引。

3.1 命名实体的抽取与索引构建

在 RDF 数据中,我们将描述实例名称(即命名实体)的属性称为"命名属性"。一个实例可以有一个或多个命名属性,不同类别的实例有不同的命名属性,譬如,人与组织机构的命名属性有 foaf:name(名称)和 foaf:nick(别名)。我们采用语义网开源工具包 Jena 中的 RDF API 对 RDF 数据进行解析,抽取每个实例的 URI 标识符、所属的类、命名属性及其值作为索引中的域值。其中,前三项都是 URI 的形式,是非索引项,只对它们进行存储但无需索引,最后一项"命名属性的值"是索引项。命名实体是不能分解的,要作为一个单独的语汇单元被搜索,因此对该索引项进行存储并索引,但并不进行分析。

此外,从实例中抽取的信息同时也以 JSON 格式存储。在随后进行汉语分词时,JSON 数据中存储的命名实体将会被提取出来加入到分词软件的用户词典中,从而保证在分词过程中正确切分出命名实体。

3.2 本体词汇的抽取与索引构建

图书馆关联数据所涉及的本体有:CORE 核心元数据本体、FOAF 本体、EVENT 本体、geoNames 本体、DCTERMS 元数据规范和 SKOS 本体[3,27]。在对上述本体进行解析前,我们首先对其 OWL 文件进行精炼,只保留与实例数据有关的类与属性,删除其他不相关的类与属性,而且保证每个保留的类与属性都有相应的中文标签。针对本体,我们采用语义网开源工具包 Jena 中的 Ontology API 对其进行解析,抽取出本体中类与属性的 URI 标识符、属性的领域①、属性的

① 对于没有指定领域和值域的属性,如果是子属性则领域和值域继承其父属性,其他的则由程序自动赋值为"Any"。

值域、类与属性的文字标签、属性类型以及来源本体作为索引中的域值。前三项都是 URI 的形式,后两项是受控词汇,这五项都是非索引项,只对它们进行存储而无需索引,只有类与属性的文字标签是索引项,对其进行分词处理和同义词扩展。同义词主要来源于哈尔滨工业大学开发的《同义词词林扩展版》①和《图书情报词典》②。在索引期间进行同义词扩展,虽然索引所占用的磁盘空间会比较大,但能够加快搜索速度,这对于问答系统是至关重要的。此外,抽取出的本体词汇同样也以 JSON 格式存储,以供汉语分词时构建用户词典。

4 问句预处理

问句预处理包括汉语分词、语义标注和问句分类三部分。

4.1 汉语分词与命名实体识别

本研究采用中科院计算机技术研究所开发的 ICTCLAS③ 汉语分词系统对用户提问进行分词处理。ICTCLAS 虽然能够对通用的人名、地名和组织机构名进行识别,但却无法识别特定类型的命名实体,譬如图书情报领域所涉及的大量书名、刊名、文章名、学术会议名等。此外,ICTCLAS 通常切分出的是 2—3 个字的短词汇,对于较长的本体词汇(如"相关概念""出版周期""修改日期"等)是无法正确切分的。为了保证在汉语分词过程中实现对命名实体和本体词汇的正确切分,我们自定义了用户词典。用户词典包括从 RDF 数据中抽取出的命名实体(见 3.1 节)、从本体中抽取出的类与属性的中文标签以及它们的同义词(见 3.2 节)。用户词典被引入到分词系统后,分词时会优先查询该词典,从而对问句中的命名实体和本体词汇实现正确切分。

4.2 时间日期的标注

在分词结果中,首先根据词性标签判断时间日期表示。因为分词过程中常将一个完整的时间日期表示(如"今年 2 月 4 日")切分成若干短语,此时要将这些被切分的短语重新合并成一个完整的时间日期表示,然后将问句中各种时间日期表示转换为规范的表示格式,本研究采用 XML Schema 中定义的时间日期表示格式。对于不明确的时间日期表示(如去年、今年等),则根据当前系统日期推断其确切时间。

4.3 命名实体的标注

在分词结果中,对于每个被标注为"ne"的命名实体词汇,在命名实体索引中查找与其精确匹配的索引项(即命名属性的值),并提取出所对应的实例资源(URI)、命名属性和所属的类。一个字符串有可能同时是多个命名实体的名称,譬如作为书名的"信息检索"和作为概念名的"信息检索",因此针对一个查询词会返回多个查询结果。此时,用所有的查询结果对命名实体进行标注,这些标注结果之间是并列可选关系。在后续处理中,将会根据整个句子的标注情况过滤不符合语义与逻辑的标注结果。

4.4 本体词汇的标注

为了减少搜索索引的次数,在分词结果中首先过滤掉问句中的助词、叹词、语气词等停用词,只保留有用的实词,然后在本体词汇索引中查找与每个保留词相匹配的索引项。这些保留词是经过分词处理后切分出的词汇,因此在搜索索引时无需再对它们进行分词处理。索引项是经过分词处理和同义词扩展的,因此检索结果有完全匹配和部分匹配两种情况。譬如,查询词为"书籍"时,会返回"图书""图书章节""编辑图书"三个查询结果。在默认情况下,

① http://www.datatang.com/data/42306

② http://mall.cnki.net/reference/read_R200908039.html

③ 关于 ICTCLAS 的详细介绍见官方网站 http://www.ictclas.org

Lucene是通过关联评分对匹配结果进行降序排列，因此第一个查询结果的关联度最高。在本研究中，我们只选取第一个查询结果“图书”（对应#Book类）对查询词“书籍”进行标注，而忽略另两个查询结果。

问句语义标注的最终结果以XML格式输出，图2所示为例句1“去年苏新宁写了哪些关于信息检索主题的书籍？”的标注结果。

```
<question>
    <token id="1" Type="DateTime" Value="2014">去年</token>
    <token id="2" Type="NamedEntity" Class="http://xmlns.com/foaf/0.1/Person" Predicate="http://xmlns.com/foaf/0.1/name">苏新宁</token>
    <token id="3" Type="ObjectProperty" OntResource="http://purl.org/dc/terms/creator" Domain="http://purl.org/ontology/core#Document" Range="http://xmlns.com/foaf/0.1/Agent">写</token>
    <token id="4">了</token>
    <token id="5">哪些</token>
    <token id="6">关于</token>
    <token id="7">
    <annotation Type="NamedEntity" Class="http://purl.org/ontology/core#Book" Predicate="http://purl.org/dc/terms/title"/>
    <annotation Type="NamedEntity" Class="http://www.w3.org/2004/02/skos/core#Concept" Predicate="http://www.w3.org/2004/02/skos/core#altLabel"/>信息检索</token>
    <token id="8" Type="ObjectProperty" OntResource="http://purl.org/dc/terms/subject" Domain="http://purl.org/ontology/core#Document" Range="http://www.w3.org/2004/02/skos/core#Concept">主题</token>
    <token id="9">的</token>
    <token id="10" Type="NamedClass" OntResource="http://purl.org/ontology/core#Book">书籍</token>
    <token id="11">？</token>
</question>
```

图2 例句1的语义标注结果

4.5 问句分类

问句分类是指将用户输入的问句区分为简单句和复杂句的过程。这里所说的简单句和复杂句是指句子在语义和本体逻辑上的复杂度，即问句所涉及的本体的数量和本体中类的数量及相互间关系的远近，因为这会直接影响SPARQL查询的构建。在本研究中，我们定义：简单句是只围绕单一数据集中某一个或某一类个体（即实例）进行提问的问句，也就是说，简单句只涉及单一本体中的单一类及其属性，属性可以是数据类型属性，其值是文字值，也可以是对象属性，其值是相关联类的命名属性。简单句可进一步细分为两类。

（1）A类：已知个体的名称，查询其某一属性的属性值。①如果所查询的属性是数据类型属性，则属性值是文字值。譬如，“信息检索这本书是什么时候出版的？”②如果所查询的属性是对象属性，则属性值是对象的URI地址。譬如，“信息检索这本书的作者是谁？”

（2）B类：已知个体某些属性的属性值，查询其名称（限定1到3个属性）。①已知属性是对象属性，譬如，“关于信息检索这一主题的图书有哪些？”②已知属性是数据类型属性，譬如，“2014年出版的图书有哪些？”问句中数据类型属性的值（即文字值）比较难以标注，且这类问题在实际中也较少见，而且大多数数据类型属性都可以转换为对象属性，因此在本研究中，我们只处理“时间日期”类的数据类型属性，对含有其他数据类型属性的问句暂不做回答。③已知属性是两种属性的混合，譬如，“2014年出版的作者是王军的书有哪些？”

在本研究中，我们采用基于规则的分类开

发了问句分类算法,根据问句中语义标注结果对问句的类别进行判定。对于一个用户输入,首先判别它是否为一个语义完整的有效问句。有效问句需满足下列条件之一:①问句中含有至少一个命名实体和至少一个属性;②问句中含有至少一个时间日期和至少一个时间日期属性(以时间日期为值域的数据类型属性);③问句中含有至少一个时间日期和一个类。如果问句是一个有效问句,则继续判别它是否为简单句;如果是无效问句,则直接忽略。

简单句的判别通过计算问句中的命名实体、类与属性的数量,以及相互间关系来实现。在问句语义标注过程中,命名实体的标注是直接进行精确字符串匹配,标注准确度最高;其次是本体的命名类,主要是对名词词汇进行精确、包含或同义词匹配;本体属性的标注则最为困难,涉及名词、动词等多种词汇,准确度最差。为了减少因语义标注错误而引起的误判,在简单句的判别算法中要尽量优先考虑命名实体,其次是本体类,最后是本体属性。因篇幅所限,这里对具体的判别算法不予详述。

除简单句外的其他有效问句都属于复杂句,一个复杂句可以分解为两个或多个简单句。譬如,例句2"信息检索的作者所写的书籍都是关于哪些主题的?"可分解为:

子句1:信息检索的作者?

子句2:XXX所写的书籍?(XXX为子句1的答案)

子句3:YYY都是关于哪些主题的?(YYY为子句2的答案)

5 简单句的问答过程

简单句的问答过程分为两部分:①基于问句的语义标注结果生成相应的本体兼容三元组;②将三元组进行合并构建完整的SPARQL查询。

5.1 本体兼容三元组的生成

基于问句中标注的命名实体、本体类、本体属性和时间日期,生成本体兼容三元组的具体规则如下。

(1)基于标注的命名实体生成三元组。这类三元组被称为实例三元组,共有两个:一是描述实例类型的三元组,二是描述实例名称的三元组。如果命名实体具有多重标注,则基于每种标注生成多组并列可选的实例三元组。

(2)基于标注的本体属性生成三元组。这类三元组只有一个:以本体属性为谓语,主语为未知对象。该三元组被称为属性三元组,是查询中的核心三元组。

(3)基于标注的本体类生成三元组。这是一个描述主语类型的三元组,称为类三元组。

(4)基于标注的时间日期生成三元组。如果问句中含有时间日期属性,则直接将时间日期作为时间日期属性三元组中的宾语。如果问句中不含时间日期属性,则仅依靠时间日期标注无法生成三元组,还需考虑问句中标注的本体类。对于本体类,如果它具有时间日期属性,则以未知对象为主语,该类的时间日期属性为谓语,规范的时间日期表示为已知的宾语,构建一个时间日期三元组;如果一个类有多个时间日期属性,或者有多个本体类具有时间日期属性,则构建多个并列可选的时间日期三元组。

5.2 简单句SPARQL查询的构建

针对不同类型的简单句,我们制定了不同的合并规则,将本体兼容的三元组进行组合,填入到预先构建的SPARQL SELECT查询模板的查询条件项中。

针对A类简单句,无需考虑类三元组,只需对实例三元组和属性三元组进行组合,其查询目标是属性三元组中的未知宾语。如果问句中含有本体类,则检测该类与实例类之间的关系。如果该类与标注的实例类中其中一个相同,则只保留基于该种标注构建的实例三元组。采用实例三元组中的主语替换属性三元组中的未知主语,然后将实例三元组和属性三元组进行组合后填入到SPARQL查询模板。如果没有过滤

或过滤后仍有多组并列可选的实例三元组，则每组实例三元组分别与属性三元组进行组合，不同集合之间以 OPTIONAL 连接。

针对 B 类问句，需要将实例三元组、属性三元组、类三元组和时间日期三元组进行组合，其查询目标是属性三元组中的未知主语。如果问句中含有多个本体类，检测本体类与属性领域的关系，只保留与属性领域相同或其子类的类三元组和时间日期三元组。采用实例三元组中的主语替换属性三元组中的未知宾语，采用类三元组中的主语替换属性三元组中的未知主语和时间日期三元组中的未知主语，然后将所有剩下的三元组进行组合后填入 SPARQL 查询模板。如果有多个属性三元组和多个实例，采用所属类与属性值域相同或为其子类的实例替换属性三元组中的未知宾语。如果有多组并列可选的实例三元组，过滤掉与所有属性值域均不相同或不在其范围内（即子类）的实例三元组。如果没有过滤或过滤后仍剩下多组实例三元组，将每组实例三元组分别与其他三元组进行组合，不同集合之间以 OPTIONAL 连接。例句 1 的 SPARQL 查询如图 3 所示。

```
命名空间前缀（略）
SELECT ?Subject                                  查询目标是属性三元组中的未知主语
FROM <http://resource/test>
WHERE {
?Instance1 rdf:type foaf:Person.                 } 实例三元组 1
?Instance1 foaf:name "苏新宁"@zh.                }
?Subject dcterms:creator ?Instance1.             属性三元组 1（核心三元组）
?Instance2 rdf:type skos:Concept.                } 实例三元组 2
?instance2 skos:prefLabel "信息检索"@zh.         }
?Subject dcterms:subject ?Instance2.             属性三元组 2（核心三元组）
?Subject dcterms:date "2007"^^xsd:gYear.         时间日期三元组
?Subject rdf:type core:Book.                     类三元组（该类为属性领域的子类）
}                                                结束括号
```

图 3 例句 1 的 SPARQL 查询

6 复杂句的问答过程

复杂句的特征主要有两点：一是句子的句法结构复杂；二是句子的语义结构复杂，会涉及来自多个本体的多个类及属性。针对复杂句，如果还是根据语义标注的结果直接生成 SPARQL 查询，其难度和错误率都会增加很多。因此，在本研究中我们提出了一种新的处理方法，通过句法解析将复杂句分割成若干个简单句，针对每个简单句基于上文所述的规则直接生成本体兼容的三元组，最后将所有的三元组组合起来构建针对复杂句的 SPARQL 查询。

6.1 问句分解

问句分解的功能是把一个复杂问句分割成若干个简单句。分解过程需把关系密切的词语分割在同一个简单句中，这样才能保证分解后问句的语义不发生变化。在本研究中，我们利用问句词汇间的依赖关系对问句进行切分。

首先，采用斯坦福大学自然语言处理小组开发的 Stanford Parser 对复杂句进行句法解析并输出词汇间的依赖关系，然后基于依赖关系生成一个被称之为斯坦福依赖树的树状结构。图 4 所示为例句 2“信息检索的作者所写的书籍都是关于哪些主题的?”的斯坦福依赖树。在一棵依赖树中，具有密切依赖关系的词语通常位于

同一棵子树中，而关系疏远的词语则被分割在不同的子树中，因此，基于依赖树中的子树，可将一个复杂句分割成若干个简单子句。复杂句的分割算法如下。

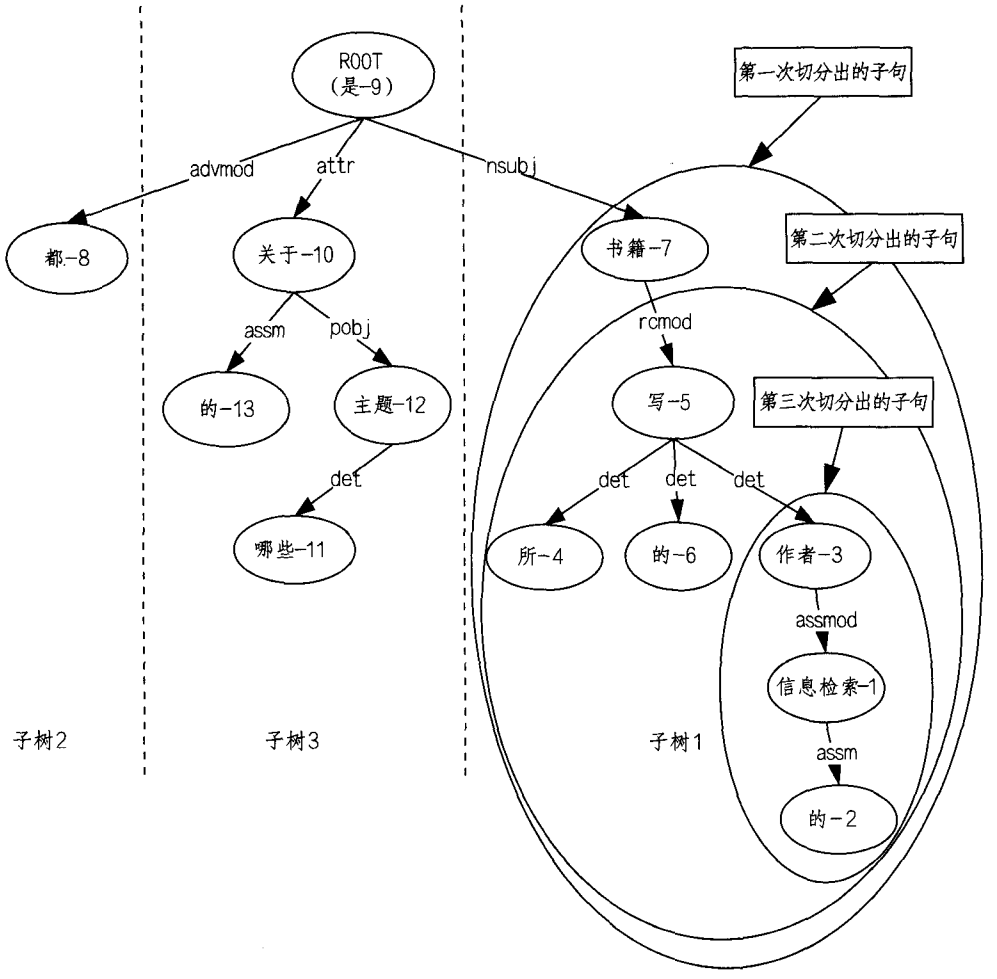

图 4　例句 2 的斯坦福依赖树及其分割过程

(1)将整个解析树记为 G(n)，n(n>0)为树中节点数，从树的根节点开始，自上而下，依次遍历 G 中的每个非终节点 i (0<i≤n)；

(2)将节点 i 处的分支数记为 m (m≥1)，该节点下的子树记为 F_{ij}(1≤j≤m)，依次判别每个子树 F_{ij}是否为有效句，如果 F_{ij}为有效句，将其作为一个子句切分出来；

(3)当节点 i 下的所有子树都判别结束，将解析树 G 中未切分的剩余部分作为一个子句输出；

(4)对于从 G 树中切分出的每个子句 F_{ij}(含剩余部分构成的子句)，首先判别它是简单句还是复杂句，如果是简单句，将其作为一个子句输出；如果是复杂句，则继续对其进行分割；

(5)将切分出来的一个复杂子句 F_{ij}看作是一个完整子树，重复上述过程，直至切分出的子句中没有复杂句。切分出的所有子句如表 1 所示。

表 1 例句 2 基于斯坦福依赖树切分出的子句

根节点下的子树	子句			
子树 1	#1	[信息检索-1]→[的-2]→[作者-3]→[所-4]→[写-5]→[的-6]→[书籍-7]		
		#1-1	[信息检索-1]→[的-2]→[作者-3]→[所-4]→[写-5]→[的-6]	
			#1-1-1	[信息检索-1]→[的-2]→[作者-3]
			#1-1-2	[所-4]→[写-5]→[的-6]
		#1-2	[书籍-7]	
子树 2	#2	[都-8]→[是-9]→[关于-10]→[哪些-11]→[主题-12]→[的-13]		
根节点				
子树 3				

注：根节点下的子树按照在句子中出现的顺序排列。

复杂句切分完成后，切分出的子句均为简单句或无效句，但有些子句过于琐碎（譬如，子句#1-2“书籍”只是一个词汇而非句子），因此还需对它们进行合并，使最后输出的每个子句都是语义完整的句子。子句合并算法包含三个主要步骤：①将切分出的子句按照在句子中出现的顺序进行排列，然后将前一个子句的答案（即实例）代入到后一个子句中补充其语义；②如果子句中不含有本体属性（譬如，子句#1-2），即使代入实例也无法形成有效句，则将这类子句与相邻子句进行合并，形成语义完整的有效句，合并原则是“邻近优先和同一母句优先”。譬如，根据邻近优先原则，子句#1-2 可以并入到子句#1-1-2 或#2 中，再根据同一母句优先原则，优先将其并入到属于同一母句（即子句#1）的子句#1-1-2 中；③有的无效句代入上一个句子的答案（即实例）后，从一个无效句变成一个复杂句，此时重新调用前文所述的问句分割算法对其进行分割。根据子句合并算法，对表 1 中的子句进行合并，结果如表 2 所示。

表 2 将分割后的子句进行合并的结果

序号	切分出的子句	合并前子句类型	合并后的子句	合并后子句类型
#1-1-1	[信息检索-1]→[的-2]→[作者-3]	有效简单句	[信息检索-1]→[的-2]→[作者-3]	原始简单句
#1-1-2	[所-4]→[写-5]→[的-6]	只含属性	<Instance$_{(\#1-1-1)}$>→[所-4]→[写-5]→[的-6]→书籍	衍生简单句
#1-2	[书籍-7]	只含类		
#2	[都-8]→[是-9]→[关于-10]→[哪些-11]→[主题-12]→[的-13]	只含属性	<Instance$_{(\#1-1-2)}$>→[都-8]→[是-9]→[关于-10]→[哪些-11]→[主题-12]→[的-13]	衍生简单句

注：①切分后的子句按照在句子中出现的顺序排列；②为了与原始简单句相区别，代入上一个句子的答案后形成的简单句被称为衍生简单句。

6.2 复杂句 SPARQL 查询的构建

对于切分出的子句，首先按照简单句的处理规则生成各自的 SPARQL 查询，然后对这些 SPARQL 查询进行组合，生成针对复杂句的完整 SPARQL 查询。整个组合过程采用迭代递归算法，具体如下。假设一个复杂句 Q 被分解为 n(n

>0)个简单子句 $SQ_{(i)}$(i=1,2,…n),对于每个子句 $SQ_{(i)}$,首先判别其类型(A 类或 B 类简单句),然后根据类型对子句的三元组进行组合,生成 SPARQL 查询并确定其查询目标。如果查询目标是文字值,则忽略;如果查询目标是实例对象,则将查询目标实例 Instance(i)及其所属类别 Class(i)传递到下一个子句 $SQ_{(i+1)}$,使子句 $SQ_{(i+1)}$成为一个有效简单句,然后重复上述过程。当到达最后一个子句 $SQ_{(n)}$时,将该子句的查询目标作为整个复杂句的查询目标。例句 2"信息检索的作者所写的书籍都是关于哪些主题的?"SPARQL 查询的构建过程如图 5 所示。

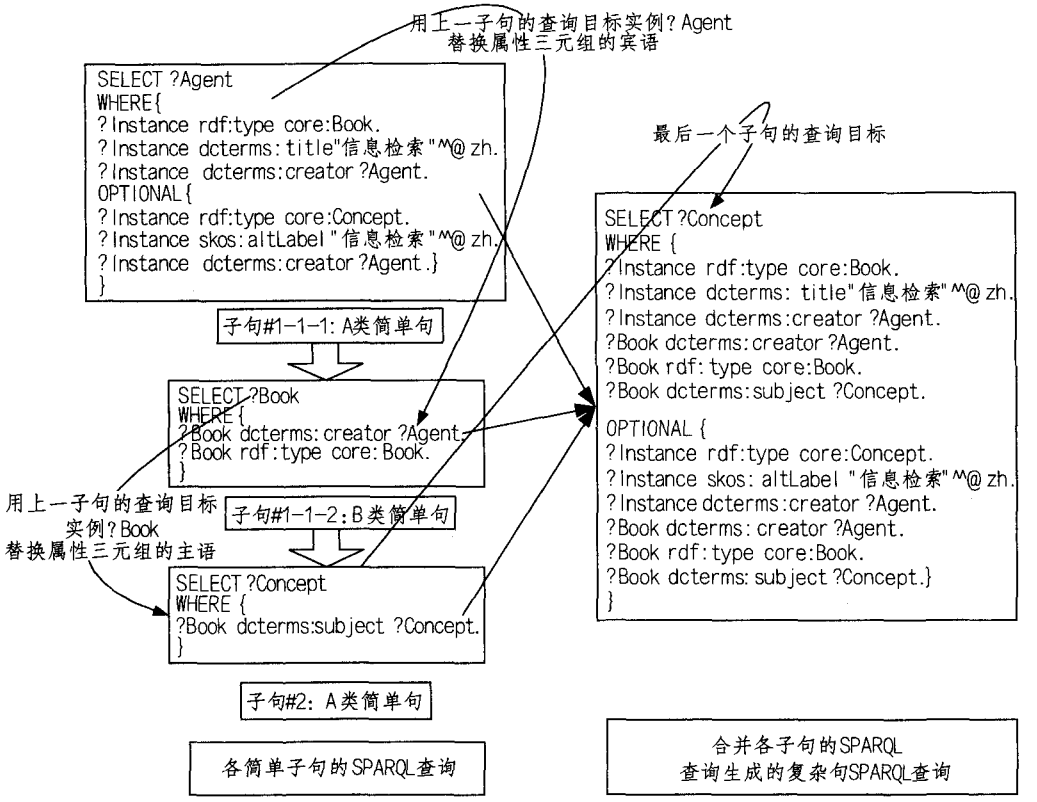

图 5 合并简单子句的 SPARQL 查询生成复杂句 SPARQL 查询的过程

7 实验测评

本研究采用 100 个测试问句对构建的自动问答系统进行实验测评。测试问句由来自图书情报学科的 5 名本科生和 5 名研究生基于领域知识构建。这 10 个用户构建的原始问句存在大量重复,而且有些问句语义不完整,句法不规范,或者在数据集中根本找不到答案,经人工过滤掉上述问句后,最后保留 50 个简单句和 50 个复杂句作为测试问句,部分问句如表 3 所示。针对 RDF 数据,只要有正确的 SPARQL 查询就能检索到正确的答案,因此只需检测系统是否能够将问句转换为正确的 SPARQL 查询,而无需检测问句的最终答案。此外,本研究还对自动问答过程中的四个主要步骤,即分词与语义标注、问句分类、复杂句分解、SPARQL 查询构建分别进行测评,以便深入了解问答过程中产生错误的主要原因,为进一步提高问答系统的精度提供依据。

表3 用户手工构建的部分测试问句

ID	简单句	复杂句
1	基于文本蕴含的语义网数据自动问答系统这篇文章来源于哪本会议论文集？	基于CSSCI本体的学科关联分析的作者还发表了哪些论文？
2	情报学报的出版周期有多长？	指导学位论文数据挖掘与竞争情报系统的人的个人主页是什么？
3	面向关联数据的语义数字图书馆资源描述与组织框架设计与实现发表于哪家期刊？	ICADL2011会议出品的会议论文集是由哪家机构出版的？
4	数据挖掘与竞争情报系统这篇学位论文的指导老师是谁？	来源于第13届亚太数字图书馆国际会议论文集的论文都是关于哪些主题的？
5	数据挖掘有哪些相关概念？	由苏新宁写的关于信息检索主题的文章发表于何处？
6	研究方向为数字图书馆的学者有哪些？	数据仓库和数据挖掘的作者还写过哪些主题的书？
7	苏新宁的书有哪些？	信息分析丛书所包含的书籍都是由哪些人写的？
8	王军写过哪些有关数字图书馆这一主题的书？	以情报检索的下位词为主题的文献有哪些？
9	在ICADL2011会议上发表的有关信息检索主题的文章有哪些？	苏新宁指导的学位论文都是关于哪些主题的？
10	由南京大学授予的有关数据挖掘主题的学位论文有哪些？	面向关联数据的语义数字图书馆资源描述与组织框架设计与实现所属的那期刊物还同时包含了哪些论文？

在对问句进行汉语分词时，本研究分别采用三种不同的用户词典，即命名实体词汇、命名实体词汇+本体词汇、命名实体词汇+本体词汇+本体词汇的同义词。因为每种分词词典都包含命名实体词汇，而命名实体的语义标注是采用字符串精确匹配的方式进行，其标注正确率在三种情况下都是100%，因此，在实验中我们只测试三种不同分词情况下本体类与属性的标注准确率、召回率以及整个句子的标注精确率，结果如表4所示。实验结果表明，汉语分词对问句的语义标注结果有明显影响，采用扩展的用户词典能够极大提高问句中领域词汇（即本体词汇）切分的正确性，从而提高语义标注的正确性，最好的标注结果是采用“命名实体词汇+本体词汇+本体词汇的同义词”作为用户定制词典进行分词时获得的，100个测试句中只有5个句子未能被正确标注，整句标注精确率达到95%。在语义标注时，通过使用LUCENE同义词分析器，基本上能够正确标注问句中采用不同自然语言词汇表达的本体概念与关系，标注错误主要在于不能正确识别隐含属性、非词组表达的属性、一词多义和互逆属性。

表4 采用三种不同分词词典时问句的语义标注结果

分词字典	本体类的标注			本体属性的标注			整句标注准确率(%)
	准确率(%)	召回率(%)	F值(%)	准确率(%)	召回率(%)	F值(%)	
命名实体词汇	83.5	82.5	83	86.7	70.1	77.5	40
命名实体词汇+本体词汇	100	100	100	91.0	79.0	84.6	66
命名实体词汇+本体词汇+本体词汇的同义词	100	100	100	97.6	97.6	97.6	95

在问句语义标注正确的情况下，分别测试问句分类、复杂句分解和SPARQL查询构建三个步骤的精确率，结果如表5所示。在100个测试问句中，只有3个复杂句被误判为B类简单句，其余97个问句的类别都被正确识别；50个复杂句中有48个问句被正确分解为简单句；在语义标注、问句分类和复杂句分解都正确的前提下，无论是简单句还是复杂句都能够基于规则全部正确地构建SPARQL查询。这说明，本文所提出的基于规则的问句分类算法、基于解析树的复杂句分解算法、基于规则的SPARQL查询构建算法都是有效的。

问答系统的最终精确率取决于问句语义标注、问句分类、复杂句分解和SPARQL查询构建四个步骤的累积精确率。经过上述步骤的处理，在100个测试问句中最终有3个简单句和6个复杂句未能生成正确的SPARQL查询，因此简单句的回答精确率为94%，复杂句为88%，合计为91%，如表5所示。

表5 问答过程中主要步骤的精确率及最终结果

测试问句		每一步单独测试的结果				累计结果
种类	数量	语义标注精确率(%)	问句分类精确率(%)	复杂句分解精确率(%)	SPARQL查询构建精确率(%)	回答精确率(%)
简单句	50	94	100	-	100	94
复杂句	50	96	94	96	100	88
合计	100	95	97	-	100	91

注：回答精确率为最终生成正确SPARQL查询的问句在测试问句中所占的比率。

8 结论与展望

本文提出了一种面向图书馆关联数据的自动问答新方法，通过将自然语言提问转换为结构化的SPARQL查询，从图书馆领域相互关联的五个RDF数据集中提取特定答案。该方法的前提是采用本体(即RDF数据集)中的语义元素对问句进行正确语义标注，本研究目前采用基于信息检索的标注方式将问句中的命名实体与本体中的实例相映射，将问句中的其他词汇与本体词汇及其同义词相映射。这一标注方法能够解决绝大部分问题，但对隐含属性、非词组表达的属性、多义词表达的属性还无法解决，而且对本体的依赖程度较高。该方法的创新点在于，将问句分为涉及一个数据集的简单句和涉及多个数据集的复杂句分别进行处理，又将简单句进一步分为查询属性(A类)和查询实例(B类)两种，分别制定SPARQL查询构建规则，复杂句则分解成若干个简单句进行处理。实验结果表明，这是一种行之有效的自动问答方法，大大简化了复杂句的处理过程，使复杂句SPARQL查询的构建达到88%的精确率，全部测试问句(含简单问和复杂问)的回答精确率达到91%。

在下一步研究中，我们将对语义标注方法进行改进，拟增加通用的语言分析层以提高标注方法的领域通用性，并重点提高属性的标注精度。此外，此方法目前只能回答在RDF数据集中有明确表述的信息，对于需要推理和计算才能得出答案的提问(如比较级、最高级问题)还无法回答，下一步我们将对此类问句进行探索。

参考文献

[1] Berners-Lee T. Linked data-design issues [EB/OL]. [2015-04-05]. http://www.w3.org/DesignIssues/LinkedData.html.

[2] Schmachtenberg M, Bizer C, Paulheim H. State of the LOD cloud 2014 [EB/OL]. [2015-04-05]. http://linkeddatacatalog.dws.informatik.uni-mannheim.de/state.

[3] 欧石燕. 面向关联数据的语义数字图书馆资源描述与组织框架设计与实现[J]. 中国图书馆学报,2012,38(6):58-71. (Ou Shiyan. Design and implementation of a linked data-oriented framework for resource description and organization in semantic digital libraries[J]. Journal of Library Science in China,2012,38(6):58-71.)

[4] Lopez V, Uren V, Sabou M, et al. Is question answering fit for the semantic web?: a survey [J]. Semantic Web Journal,2011 (2):125-155.

[5] Mollá D, Vicedo J. Question answering in restricted domains: an overview [J]. Computational Linguistics,2007,33(1):41-61.

[6] Androutsopoulos I, Ritchie G D, Thanisch P. Natural language interfaces to databases-an introduction [J]. Natural Language Engineering,1995,1(1):29-81.

[7] Woods W A, Kaplan R M, Webber B N. The Lunar Sciences natural language information system: final report: BBN Report 2378[R]. Cambridge: Bolt Beranek and Newman Inc,1972.

[8] Popescu A, Etzioni M, Kautz H. Towards a theory of natural language interfaces to databases [C]//Proceedings of the 8th International Conference on Intelligent User Interfaces. New York, NY: ACM,2003:149-157.

[9] Katz B, Felshin S, Yuret D, et al. Omnibase: uniform access to heterogeneous data for question answering [C]// Proceedings of the 6th International Workshop on Applications of Natural Language to Information Systems. London, UK: Springer-Verlag,2002:230-234.

[10] Lopez V, Uren V, Motta E, et al. AquaLog: an ontology-driven question answering system for organizational semantic intranets [J]. Journal of Web Semantics: Science, Services and Agents on the World Wide Web,2007,5(2):72-105.

[11] Wang C, Xiong M, Zhou Q, et al. PANTO: a portable natural language interface to ontologies [C]//Proceedings of the 4th European Conference on the Semantic Web: Research and Applications. Heidelberg, Berlin: Springer-Verlag,2007:473-487.

[12] Ferrández Ó, Izquierdo R, Ferrández S, et al. Addressing ontology-based question answering with collections of user queries [J]. Information Processing and Management,2009,45(2):175-188.

[13] Lopez V, Fernández M, Motta E, et al. PowerAqua: supporting users in querying and exploring the semantic web content [J]. Semantic Web,2012,3(3):249-265.

[14] He S, Liu K, Zhang Y, et al. Questioning answering over linked data using Markov first-order logic [C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL,2014:1092-1103.

[15] Shekarpour S, Ngonga A, Auer S. Question answering on interlinked data [C]// Proceedings of the 22nd International Conference on World Wide Web. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee,2013:1145-1156.

[16] Ding L, Finin T, Joshi A, et al. Swoogle: a search and metadata engine for the semantic web [C]//Proceedings of the 13th ACM International Conference on Information and Knowledge Management. New York, NY: ACM,2004:652-659.

[17] Oren E, Delbru R, Catasta M, et al. Sindice.com: a document-oriented lookup index for open linked data[J]. International Journal of Metadata, Semantics and Ontologies,2008,3(1):37-52.

[18] Bernstein A, Kaufmann E, Kaiser C. Querying the semantic web with ginseng: a guided input natural language search engine [C/OL]//Proceedings of the 15th Workshop on Information Technology and Systems, 2006: 112-126 [2015-03-12]. http://www.merlin.uzh.ch/contributionDocument/download/2283.

[19] Cimiano P, Haase P, Heizmann J, et al. Towards portable natural language interfaces to knowledge bases-the case of the ORAKEL system [J]. Data & Knowledge Engineering, 2008, 65(2): 325-354.

[20] Unger C, Cimiano P. Pythia: compositional meaning construction for ontology-based question answering on the semantic web [C]//Proceedings of the 16th International Conference on Natural Language Processing and Information Systems. Berlin, Heidelberg: Springer-Verlag, 2011: 153-160.

[21] Kaufmann E, Bernstein A, Fischer L. NLP-Reduce: a "naïve" but domain-independent natural language interface for querying ontologies[C/OL]//Demonstrations of the 4th European Semantic Web Conference, 2007[2015-03-12]. http://www.merlin.uzh.ch/contributionDocument/download/2304.

[22] Kaufmann E, Bernstein A, Zumstein R. Querix: a natural language interface to query ontologies based on clarification dialogs [C/OL]//Proceedings of the 5th International Semantic Web Conference. Berlin, Heidelberg: Springer-Verlag, 2006: 980-981 [2015-03-12]. http://www.merlin.uzh.ch/contributionDocument/download/2186.

[23] Kim J, Cohen K. Natural language query processing for SPARQL generation-a prototype system for SNOMEDCT [C/OL]//Proceedings of BioLINK SIG 2013. Berlin, Germany: The BioLINK Special Interest Group, 2013: 32-38[2015-03-12]. http://biolinksig.org/proceedings/2013/biolinksig2013_Kim_Cohen.pdf.

[24] Unger C, Bühmann L, Lehmann J, et al. Template-based question answering over RDF data [C]//Proceedings of the 21st International Conference on World Wide Web. New York, NY: ACM, 2012: 639-648.

[25] Damljanovic D, Agatonovic M, Cunningham H. FREyA: an interactive way of querying linked data using natural language [C]//Proceedings of the 8th International Conference on the Semantic Web. Berlin, Heidelberg: Springer-Verlag, 2012: 125-138.

[26] 许德山,张智雄,赵妍. 中文问句与RDF三元组映射方法研究[J]. 图书情报工作, 2011, 55(6): 45-48. (Xu Deshan, Zhang Zhixiong, Zhao Yan. Research on Chinese interrogative sentences and RDF triples mapping methods[J]. Library and Information Service, 2011, 55(6): 45-48.)

[27] 欧石燕,胡珊,张帅. 本体与关联数据驱动的图书馆信息资源语义整合方法及其测评[J]. 图书情报工作, 2014, 58(2): 5-13. (Ou Shiyan, Hu Shan, Zhang Shuai. An ontology & linked data driven semantic integration method of library information resources and its evaluation[J]. Library and Information Service, 2014, 58(2): 5-13.)

欧石燕 南京大学信息管理学院教授,博士生导师。江苏 南京 210023。

唐振贵 南京大学信息管理学院博士研究生。江苏 南京 210023。

(收稿日期:2015-06-24)