西安郵電大學

硕士研究生学位论文

基于语义理解的智能问答系统关键技术研究

作　　者：李兆兆

指 导 教 师：翟社平 副教授

学科（专业）：计算机应用技术

提交论文日期：二〇一九年六月

单位代码 11664 学 号 1603210029
分 类 号 TP393 密 级

西安邮电大学
硕士研究生学位论文

题（中、英文）目 基于语义理解的智能问答系统关键技术研究

Research on Key Technologies of Intelligent Question Answering System Based on Semantic Understanding

作 者 姓 名 李兆兆 指导教师姓名、职务 翟社平 副教授

学 科 门 类 工学 学 科（专业） 计算机应用技术

提交论文日期 二〇一九年六月

摘要

大数据、人工智能等技术的蓬勃发展使得网络数据规模急剧增长，传统搜索引擎根据用户输入的关键字返回相关网页链接的无序列表，检索结果包含大量无关冗余信息，无法满足当前用户的需求。智能问答系统综合自然语言处理、知识管理等技术理解用户查询意图，将精确答案以自然语言形式返回用户。然而现有问答系统大多采用大型语料库作为知识源，数据冗余且维度单一，同时受同义词、多义词的影响，用户问句不能清晰反映其查询意图，这造成了系统检索术语错配、返回出错等情况。语义 Web 技术的快速发展给以上问题提供了良好的解决思路，越来越多的研究者尝试将语义 Web 中的关键技术如本体构建、语义查询等应用于问答系统，以解决问答系统缺乏语义理解的问题。

本文将语义 Web 技术与问答系统相结合，使用本体作为问答系统的知识源，在语义和知识层次上描述信息，增加了语义理解和知识推理，解决传统问答系统中问句匹配度差、用户查询意图模糊的问题，旨在优化智能问答系统的语义理解能力，提供准确、全面的问答服务。本文创新工作如下：

（1）多特征融合的句子语义相似度计算。在句子相似度计算中综合考虑句子的结构信息与语义信息，提取句子词形特征、词序特征及句长特征，使用层次分析法进行权重分配，计算结构相似度；给本体中概念间关系赋予权重，使用权重计算语义距离，基于概念间语义距离描述语义相似度；最终使用加权融合方法计算整体相似度。相较于传统余弦相似度算法和基于编辑距离的相似度算法，多特征融合的相似度计算方法明显地提高了计算精度。

（2）基于语义的问句查询扩展。用户查询经问句处理后得到查询关键词序列，使用概念占有率思想将查询关键词映射到领域本体，改进最小生成树算法生成最小查询扩展子树，利用有效路径扩展查询生成树，使用语义相似度进行扩展词集的筛选重排，最终生成用户查询的关键词扩展词集，从语义层面更加清晰的描述了用户的查询意图，提高基于本体的查询效率。实验结果表明，与无扩展方法和关键词扩展方法相比，基于语义的查询扩展获得了更高的 F - 度量值。

（3）搭建基于语义理解的智能问答系统。以图书领域为背景，构建图书领域本体、常用问题库和寒暄问题库，设计常用问题库问答和语义问答结合的问答策略，实现对用户提问的准确回答。系统同时提供可视化界面对相似度计算、查询扩展和数据源进行展示，验证了上述算法的可行性和有效性。

关键词：智能问答；语义 Web；句子相似度；查询扩展

ABSTRACT

The rapid development of big data and artificial intelligence has led to a sharp increase in the scale of network data. Traditional search engine returns unordered list of matched web pages, and the results searched contain a large amount of irrelevant information, which is no longer sufficient for the current users. The intelligent question answering system integrates natural language processing, knowledge management and other technologies to understand query intent, and returns the exact answer to the user in natural language. But most of the existing question answering systems use the common corpus as a knowledge source, the data is redundant and the dimension is single. At the same time, influenced by synonyms and polysemous words, the query intent cannot be reflected by the input questions, which causes the term mismatches and some other errors. The rapid development of semantic Web provides a good solution to the above problems, more and more researchers try to apply the key technologies such as ontology construction and semantic query in the semantic Web to the question-answering system, in order to solve the problem of lack of semantic understanding.

This paper combines semantic Web with question answering system, using ontology as the knowledge source, describing information in the semantic level and knowledge level. The semantic understanding and knowledge reasoning are added to solve the problem of poor question matching degree and fuzzy user query intention in traditional question-answering system, aiming at optimizing the semantic comprehension ability and to provide accurate and comprehensive question and answering services. Main innovation theory and research results have been proposed as following:

(1) Sentence semantic similarity calculation of multi-feature fusion. In the calculation of sentence similarity, the structure information and semantic information of sentences are considered synthetically. The length features, morphological features, and word order features are extracted, and the analytic hierarchy process are used to calculate the structure similarity. The weights of the concepts in the ontology are given to calculate the semantic distance, and then the semantic similarity is described based on the semantic distance between concepts. And finally, the weighted fusion method is used to calculate the overall similarity. Compared with the traditional cosine similarity algorithm and the similarity algorithm based on editing distance, the similarity calculation method of multi-feature fusion obviously improves the calculation accuracy.

(2) Semantic-based question query expansion. The query keyword sequence is obtained after the processing of user query, and the query keyword is mapped to the domain ontology according to the concept of occupancy rate. The improved minimum spanning tree algorithm is used to generate the minimum query spanning tree, and the query spanning tree is extended by the effective path. The semantic similarity is used to

filter and rearrange the extended words set, and finally the keyword extended words set of the user query is generated. The query intent is described more clearly from the semantic level, and the ontology-based query efficiency is improved. Experimental results show that the semantic based query extension obtains a higher F-measure than the no-extension method and the keyword extension method.

(3) The building of an intelligent question answering system based on semantic understanding. Based on the background of book, the book domain ontology, the database of frequently asked questions and the database of greetings questions is constructed. The question answering strategy of the combination of the frequently asked questions and semantic is designed to realize the accurate answer of the user's questions. The system also provides a visual interface to display the similarity calculations, query expansion and data sources, verifies the feasibility and effectiveness of the above algorithm.

Keywords: Intelligent Question Answering; Semantic Web; Sentence Similarity; Query Expansion

目录

第 1 章　绪论

1.1 研究背景及意义

互联网的迅速发展和广泛普及使得人们获取信息的方式呈多样化和便捷化发展趋势，随着大数据、人工智能等技术的发展，网络信息爆炸性增长，人们很难准确、快速地获取有价值的信息。搜索引擎为人们从海量网站中迅速查找有效信息提供途径，然而其使用效果却不尽如人意[1]。世界各大调查机构和搜索引擎公司曾就搜索引擎的使用现状展开调研，并分别发布了调研结果[2]。其中，英国独资市场调查研究公司MORI的民意调查结果表明，能在网络上查找到可用信息的用户数量只占18%，大部分用户都不满意现有的搜索引擎，占总数的68%。美国知名调查公司Roper Starch调查指出，71%的用户在使用搜索引擎时遇到过麻烦；搜索过程中46%的错误都是因为网页链接出错；86%的互联网用户表示应当出现更有效、准确的信息搜索技术。数据分析提供商Keen的调查报告显示，使用搜索引擎在网上查找答案的用户数占到了31%，找寻到满意答案的平均耗时是每周8.75个小时。

从这些调查数据中不难看出，尽管诸如谷歌、百度等此类优秀的搜索引擎花费了很多时间和精力用于核心搜索技术的研发，但是传统的搜索引擎依旧存在一些缺陷，比如检索结果语义不相关、信息冗余等[3]。网络数据资源的爆炸式增长使得用户对信息的需求内容及需求方式发生了极大改变，传统基于关键词匹配的信息检索技术已经不能满足当下用户的需求[4]。用户通过常规搜索引擎已经很难一次性获得期望结果，主要有以下两点原因：

①常用搜索引擎都是基于关键词组合的输入，检索算法多为索引、匹配算法，无法触及语义信息。然而用户本身对于检索内容的要求较为复杂，这使得简单关键词的硬性排列无法清楚完整地表达用户的检索意图[5]。

②传统搜索引擎只能根据用户的关键词输出一系列按照匹配度大小排列的非结构或半结构化的网页，没有经过加工，检索结果包含大量冗余信息[6]。用户需要对检索结果进行人工筛选，筛选过程耗时耗力，有时也得不到精确结果。

基于以上背景，智能问答系统应运而生。智能问答系统是指系统接收用户以自然语言形式描述的提问，使用数据处理、查询扩展等技术从大量异构数据源中搜索出能回答提问的精准答案的信息检索系统[7]。智能问答系统的实例如表 1.1 所示。

表 1.1　智能问答实例

问题	答案
苹果的英文是什么？	Apple
世界上的四大洋都有哪些？	太平洋、大西洋、印度洋和北冰洋
什么是数据结构？	数据结构是计算机存储、组织数据的方式。

智能问答系统允许用户以自然语言的形式进行提问，在问答系统中数据源被深度加工和管理，通常以结构化形式存储。为了满足用户精确的查询需求，问答系统提供了类似人机自然交互的方式，能根据用户所输入的内容，智能给出与人类行为类似的回答。问答系统能够提供用户真正有用、精确的信息，不同于传统基于关键词查询且返回文档链接集合的搜索引擎，智能问答系统涉及知识表示、IR（Information Retrieval，信息检索）、NLP（Natural Language Processing，自然语言处理）等领域，能更有效地帮助用户从海量信息资源中提取出有效信息，同时具有更智能的人机交互体验，目前已经成为国际上一个新兴的研究热点[8]。智能问答系统与传统搜索引擎具有较大差异，表 1.2 从四个方面展开对比，说明了问答系统和搜索引擎的区别。

表 1.2　问答系统与搜索引擎的区别

	智能问答系统	传统搜索引擎
系统输入	自然语言提问	关键词组合
系统输出	准确的答案	相关文档的列表
所属领域	涉及 NLP 和 IR 等领域	纯 IR 领域
信息确定性	用自然语言提问表示，需求明确	用关键词组合表示，需求很模糊

智能问答系统主要包含问句分类、知识抽取及表示、问句语义理解、知识推理、答案匹配等关键技术，目前常见的问答系统建立大规模常用问题库，其浅层问答的完成主要采用自然语言处理、信息检索等技术[9]。然而，基于语义理解的智能问答系统的设计仍旧面临着各种挑战，用户问题和问题库问句的语义相似度计算仍是一个关键所在，如何有效结合上下文信息进行有效的推理、进而设计精确的形式化查询语言来完成问答操作也是亟待解决的问题之一。

语义 Web 中本体概念的提出给出了这个问题的一个解决思路，语义 Web 通过给互联网上的文档增加可被计算机理解的语义节点信息，使整个互联网成为一个通用的信息交换媒介。本体能在语义和知识层次上描述信息，具有良好的概念层次结构和较强的语义表达能力[10]，在计算机科学领域如数据表示、通信、互操作等方面都得到了广泛应用。近年来，基于本体的信息检索和智能问答也成为本体应用的热点领域。

因此，将语义 Web 中的知识本体库与智能问答系统相结合，以本体作为问答系统的数据源，在语义和知识层次上描述信息系统，解决诸如问句答案的语义表示、问句及答案间的语义匹配等问题，进而实现基于语义理解的垂直领域智能问答系统，就显得十分必要。

1.2 国内外研究现状

1.2.1 问答系统

上世纪 50 年代，图灵在论文《Computing Machinery and Intelligence》中提出“机器智能”的概念[11]，问答系统初具雏形。图灵用“learn by experience（通过经验学

第三章为多特征融合的句子相似度计算方法研究。使用本体作为知识库充分挖掘句子语义信息，将句子相似度分为句子结构特征和语义特征两个部分，提出一种多特征融合的句子语义相似度计算方法，通过结构特征和语义特征的加权融合解决句子之间的相似度计算问题。

第四章为基于语义的问句查询扩展方法研究。将问句查询扩展描述为三个步骤，首先将用户查询关键词映射到领域本体，之后改进最小生成树算法生成最小查询扩展子树，最后通过语义相似度进行扩展词集的筛选生成最终查询关键词。

第五章为智能问答原型系统。基于前述研究工作设计并实现了基于语义理解的智能问答原型系统，给出了原型系统的架构，阐述了系统核心技术及功能模块的实现，最后展示了原型系统的测试及实现。

第六章为总结与展望。详细总结本文的工作重点和研究成果，说明了研究存在的不足之处，对以后的工作进行展望。

第 2 章　智能问答系统理论及关键技术

2.1 语义 Web

2.1.1 语义 Web 概述

语义 Web 提供了在应用、企业和社区之间共享数据的通用框架，通过给万维网上的文档添加能够被计算机所理解的语义（元数据），从而使整个互联网成为一个通用的信息交换媒介[49]，其主要愿景是使计算机更能解读万维网“to make the web more accessible to computers”。语义 Web 包含两层含义，“语义”指符号和其所指的对象之间的关系，也可理解为符号的指称，“Web”则表示语义 Web 是对已有 Web 的一个延伸而并非创建新的 Web。

W3C（World Wide Web Consortium，万维网联盟）给出了一系列语义 Web 相关标准，并且说明了语义 Web 体系结构，各层分别有不同的含义和功能，如图 2.1 所示。

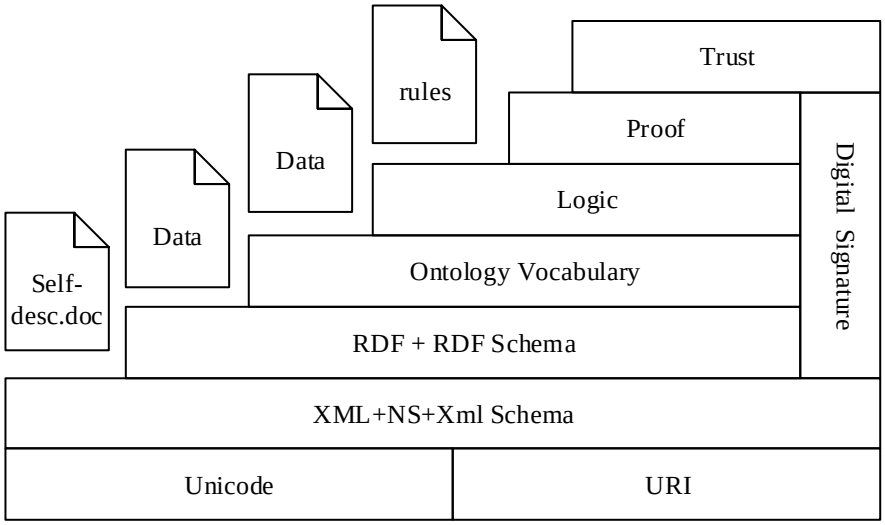

图 2.1　语义 Web 体系结构图

第一层：Unicode 和 URI。Unicode（国际编码）提供世界各种语言和符号的编码库。URI（Uniform Resource Identifier，统一资源标识符）可以给网络上信息资源一个固定唯一的标识，保证了概念的唯一性。

第二层：XML + NS + Xml Schema。XML（Extensible Markup Language，可扩展标记语言）将数据的内容、结构和表现形式分离，数据存储不受显示格式的制约，解决了如何利用语法层次来表示数据的问题；NS（Name Space，命名空间）是 XML 中 URI 的聚类依据；XML Schema 可以精确地描述 XML 文档结构，引入数据类型、命名空间，同时提供完整机制约束 XML 中标记的使用，具有较好的可扩展性和规范性。

第三层：RDF + RDF Schema。RDF（Resource Description Framework，资源描述框架）可以把领域的元数据信息描述成为三元组的方式，是一种表达本体的基本语言，主要包括图数据模型、基于 URI 的词汇、数据类型、简单事实表达式等。而 RDF Schema

则是对 RDF 的一个扩展，定义了类和属性，可以用所定义的类和属性描述其他类和属性，增强了 RDF 对资源的描述能力。

第四层：Ontology Vocabulary。Ontology Vocabulary 即本体词汇，是语义 Web 的核心层。本体不仅在知识系统的数据处理中扮演着重要的角色，而且能够在异构环境中提供互操作性，本体词汇层中主要定义了特定领域中的分类关系、概念以及概念间的关系。

第五层至第七层：分别是 Logic 层、Proof 层和 Trust 层。Logic 层负责提供公理和推理规则，增强本体语言并允许描述面向特定应用的声明式知识，为智能推理提供基础；Proof 层则包含逻辑的演绎过程，使用更低层次的万维网语言来表达及验证证明，该层执行 Logic 层的逻辑，结合 Trust 的信任机制进行评价；Trust 层提供信任机制，保证用户在 Web 上提供个性化服务，常伴随数字签名一起使用。

语义 Web 的基础技术[50]包括以下三个方面。

①使用带标签的图作为对象及其关系的数据模型，图中将对象作为节点，对象间的关系表示为边。在语义 Web 中通常使用 RDF 的形式化模型来描述这种图结构。

②使用 URI 来标识出现在数据集中的单个数据项以及它们之间的关系。

③使用本体作为数据模型来形式化地表达数据的隐含语义。诸如 RDF schema 和 OWL（Web Ontology Language，万维网本体语言）的形式化模型被用于该目的，同样也使用 URI 来表示类型和他们的属性。

语义网络通常以图的形式表达，图中结点代表实体，结点间的边代表两个实体具有某种关系，边上的谓词进一步说明了具体关系。如图 2.2 所示，该语义网络是电视节目《Family Guy》语义网的一个子图，图中结点“cvt1”与结点“Mila Kunis”之间具有关系“*actor*”。

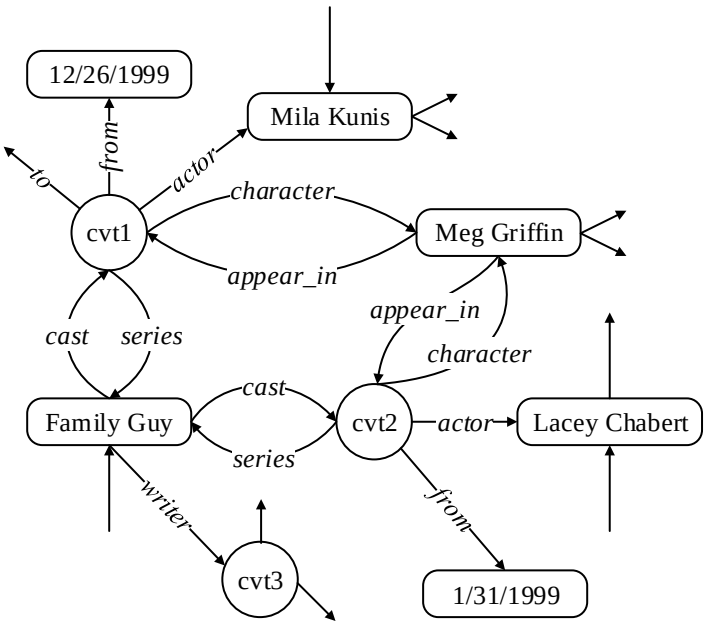

图 2.2　语义网络实例图

2.1.2 本体概述

（1）本体定义

本体在语义 Web 的层次结构中处于核心位置，是语义 Web 的重要技术之一。本体是对客观存在的一个系统的说明解释，用它来解释世界的本原以及存在的性质，之后被引入到计算机领域，来表达人们对领域的一致理解。目前关于本体的定义还未达成一致共识，人工智能领域关于本体的定义是“给出构成相关领域词汇的基本属于和关系，以及利用这些术语和关系构成的规定这些词汇外延的规则的定义”。另外不少信息系统、知识系统等领域的学者也针对本体展开深入研究，较为广泛的定义是“共享概念模型的明确形式化的规范说明”。

本体描绘了特定领域中的一组概念及概念之间的基本关系属性，并且用非常鲜明的方式表达出了这些概念的内涵。一个本体由一个包含术语及术语间联系的有限列表组成[51]，术语指领域中的重要概念，联系则刻画了概念的层次性和关联性，另外，本体还包含属性、值限制、不相交声明、对象间逻辑关系的说明等。本体具有四个特征，如表 2.1 所示。规范化是指采用了形式化的、计算机可以统一读取的约束条件；“概念化”是对客观世界中相关概念的抽象得到的概念模型，又称概念语义结构，其表现形式是建立在事实结构上的非正式约束规则，如一组概念、定义及其关系，概念不仅包含实体，也包含属性和过程。概念化对象包括概念的静态状态，以及它动态运动的过程。“形式化”主要针对计算机而言，为使得计算机可以高效处理本体，其构建和描述必须“形式化”。“共享”反映的是本体的易移植性，体现了领域内认可的一致知识及相关领域中公认的概念集，即本体针对的是公共范畴而非个体之间的共识。

表 2.1　本体特征含义

特征	具体含义
概念化	客观世界中的抽象模型
明确	必须给出概念和概念间的关系一个明确的定义
形式化	计算机可以理解和处理
共享	被一致认可的知识

（2）本体语言描述标准

W3C 的研究小组在 1999 年提出了 RDF 标准草案，又在 2004 年推出了 OWL 推荐标准，RDF 和 OWL 都是语义 Web 体系结构的概念，它们都是关于本体语言的规格描述的说明。RDF 在 XML 语法的基础上，规定了元数据的存储结构以及相关技术标准；OWL 语言具有更好的良定语法、形式语义学和表达能力，同时提供有效的推理支持，符合 XML 语法标准，可以更加深入详细地描述信息内容。

RDF 即资源描述框架，用于表达关于 Web 资源的元数据，在 Web 上被标识的任何事物的基本信息，都可以用 RDF 来进行描述[52]。RDF 提供了一种灵活且领域无关

的数据模型用以表达信息，并且利用 RDF 表达的信息在应用程序间进行交互的时候还会保持原来的语义。RDF 用 Web 标识符来表示事物，用极其简单的属性和属性的值来刻画资源之间的关系。其基础构件是一个包括了主词、谓词和宾词的三元组，表现形式为“实体 - 属性 - 取值”，声明由谓词表示的、在主词和宾词所指称的事物之间的关系。

OWL 则是 W3C 所倡导的另一种本体语言推荐标准[53]，它的目的是提供一种可以描述 Web 文档中一些类和类之间的固定的关系的语言。OWL 定义了类和类的属性对领域知识进行形式化描述，OWL 同时定义对象和对象的属性以便连接各个对象。OWL 本体主要由三种元素组成，即类（class）、个体（individual）和属性（property）。类指的是领域本体中所要描述的个体的集合，以图书领域为例，领域中所有的图书个体都属于“Book”类。个体指的是领域本体中需要研究的对象，OWL 可以用两个不同的名字来说明一个个体。两个个体之间的关系可以用属性来进行说明，OWL 中有两种主要属性，即对象属性和数据属性。对象属性反映了一对实体之间的关系，通过对象属性可以从一个实体出发连接到另一个实体；数据属性指是实体和某个数据取值之间的关系。

OWL 可以被部分映射到描述逻辑，利用已有的推理引擎进行推理，在一定基础上实现了对类和个体的推理功能。W3C 的 Web 本体语言工作组将 OWL 分成了三个不同的子语言，每个子语言都可以满足特定需求，分别是 OWL-Lite、OWL-DL 和 OWL-Full 子语言，这三种子语言的描述能力逐步提高，推理复杂度也逐步提高。

2.1.3 本体构建及查询

（1）本体构建

本体构建（Ontology Construction）实质是获取知识的过程，最重要任务是为了获取一个具体领域的知识模型，从而把该领域里的定理性的知识例如概念、概念间关系等进行一个综合抽象。由于本体构建标准不一致，因此 Gruber 提出了构建本体的五条准则，即为明确性和客观性、完全性、一致性、最大单调可扩展性、最小承诺和最小编码偏好。

本体构建过程又可以分为手工构建和自动构建，自动构建主要是指使用文本挖掘、实体抽取等技术从大规模语料中自动化抽取实体及实体间关系，从而形成本体，自动构建的本体缺乏规范性和通用性，目前是研究热点之一。手工构建本体方法较为成熟，目前最常用的方法是七步法和循环获取法。七步法是 Stanford 大学医学院研究并开发的领域本体构建方法，主要适用于特定领域本体的构建；循环获取法设计环形结构对本体进行学习并评价，需要领域专家的参与，此处不做详细说明。下面对七步法构建步骤进行详细阐述。

①确定范围。一个本体是一个特定领域的模型，因此首先必须明确所要研究的本体构建的领域，明确该本体的使用和维护对象，确定研究方向。

②考虑复用。随着语义 Web 的广泛部署，目前在公共领域（如社交网络、医学、地理）已经出现一些可用的本体。复用现有一些本体，如复旦大学研发的大规模通用领域结构化本体 DBpedia，可以有效减少人工开销。

③枚举术语。列举所有相关术语的非结构化列表，通常使用名词表示类名，动词或动词短语表示属性名。

④定义分类。定义类和类的等级体系，使③中所列出的术语以某种方式组成分类层次，可以选择自底向上或者自顶向下的方法构建分类。

⑤定义属性。在定义好分类之后，需要定义其中相关分类的属性，同时声明属性的定义域和值域，尽量满足通用性和精确性。

⑥定义限制。定义属性的限制规则，指和属性取值相关的某些特性，主要包含基数及关系特征。基数指为属性指出其是否允许或需要拥有特定数目的不同取值，如“至少 1 个值”和“最多 1 个值”；关系特征表示属性之间的关系如对称性、传递性、互逆属性和函数型取值。

⑦定义实例。根据以上步骤为类创建对应的实例，同时将实例之间用属性一一对应。实例的数目可能超越本体中类的数目几个数量级，通常从其他数据源中获取或从文本语料库中自动抽取。

目前最著名的本体编辑器是斯坦福生物医学信息研究中心开发的 Protégé，具有很好的开放性，本文使用七步法和 Protégé进行本体的开发构建。

（2）本体查询

SPARQL 指 SPARQL 协议和 RDF 查询语言[54]，其核心是以简单图模式为形式的简单查询，并且 SPARQL 还提供了一系列构造高级查询模式的高级函数，用这些函数来过滤最后的查询结果，以此规范输出的格式。SPARQL 的查询结果形式有四种类型，包括 SELECT、CONSTRUCT、DESCRIBE 和 ASK，这四种查询所返回的结果格式略有不同。SPARQL 语法和 SQL 语法有一定相似之处，查询语句由两部分组成，SELECT 部分限定查询结果返回的内容和格式，WHERE 部分是查询的限定条件，前一部分被称作结果集，后一部分被称作图模式。SPARQL 引入了查询变量来规定查询图模式中应该作为返回结果的部分，同时也规定了查询结果的表示格式。

以下给出一个简单的 SPARQL 查询的例子。

```
PREFIX ex:<http://example.org/>
SELECT ?title ?author
WHERE {?book ex:publishedBy     <http://crc-press.com/uri>
   ?book ex:title  ?title
   ?book ex:author ?author }
```

该查询包含了三个主要部分，分别以大写的关键字PREFIX、SELECT和WHERE表示。关键字PREFIX声明了一个命名空间，类似于Turtle表示法中的@prefix，与Turtle

表示法不同的是，SPARQL无需句号作为声明的结束。关键字SELECT决定了查询结果的输出格式，SELECT之后的字段声明了查询的剩余部分，列出的名字代表需要获取返回值的变量的标识符，作用是生成一个表作为查询的输出，如上的例子输出结果就是变量? title和? author对应的所有值。查询的实体以WHERE开头，后续是一个以花括号包围的简单图模式，表示在查询处理过程中可能获取到的实际值，如上例中包含的三个Turtle表示法中的三元组，每个三元组包含了URI、QName以及如? book的变量标识符，每个变量都可在多个地方使用，表示特定位置上必须使用相同的变量值。

2.2 智能问答系统关键技术

2.2.1 问答系统体系结构

问答系统可以看作信息检索的一种高级形式，对用户自然语言提出的问题给予准确、简洁的回答。国内外相关学者和不少科研机构都对问答系统的相关技术产生了浓厚兴趣，使得智能问答系统成为每年一度的文本信息检索会议上备受关注的主题之一。目前很多科研机构都在积极探索相关研究，按照类型不同可划分为专家问答系统、问答式搜索引擎、基于自然语言的数据库查询系统、智能问答系统等。

典型智能问答系统通常由数据源分析模块、问题分析模块、知识检索模块、答案生成模块和答案评估模块五个部分组成，如图 2.3 所示。

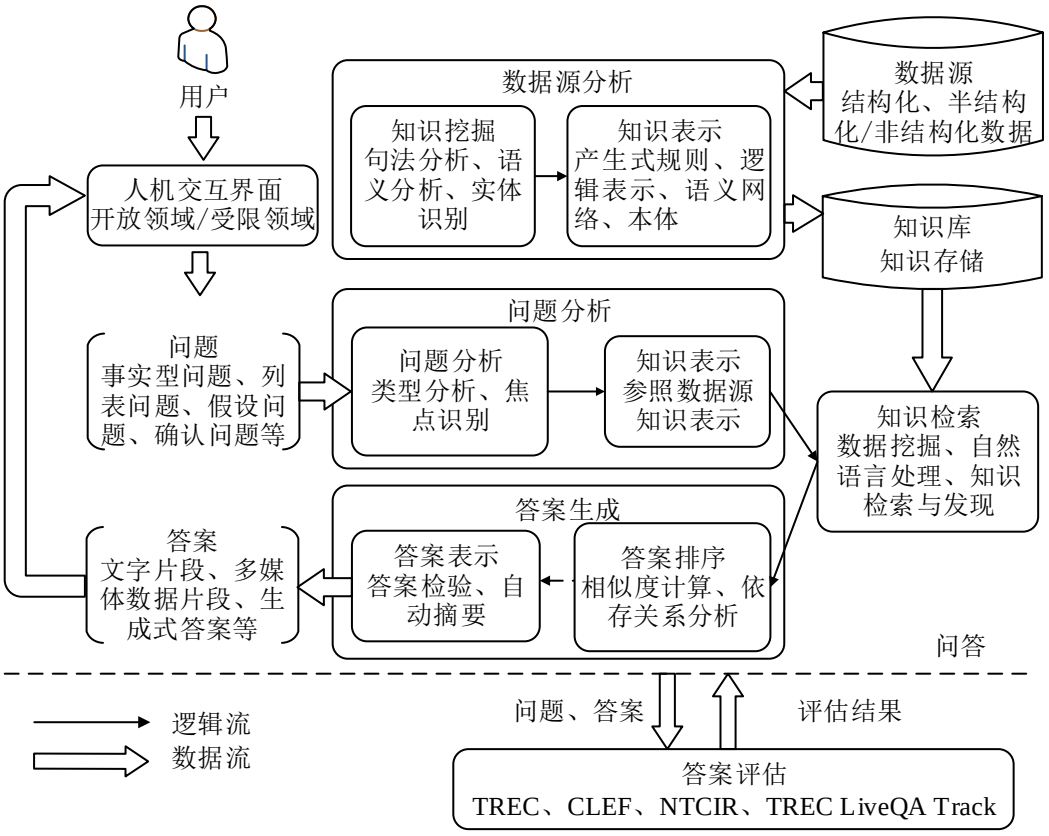

图 2.3　智能问答系统架构图

其中数据源分析模块负责从多种复杂数据源中使用知识挖掘技术提取有用的信

息生成知识表示形式；问题分析模块负责对用户的问句进行语义处理，根据用户的查询意图生成相应的查询语句[55]，此模块涉及到问题分类、问句相似度计算、查询扩展等技术；信息检索模块的主要功能是根据所生成的查询语句检索知识库，获取并解析答案；答案抽取模块则利用信息抽取技术从检索模块所检索到的相关答案中抽取最佳的候选答案返回给用户[56]。答案评估模块是一个补充模块，其对上一步生成的问题答案进行评估，评估结果反馈给其他模块，进而完善和优化整个问答系统。

另外，还有问答系统单独构建 FAQ 库存储用户经常提问的问题及其答案[57]。在基于 FAQ 的问答中，用户问句与 FAQ 库问句进行匹配，无需经过复杂的文本处理就可以快速的给出答案，在提高问答效率的同时也保证答案的正确性。

2.2.2 问句预处理

智能问答系统中对问句进行处理之前必须对问题进行一系列预处理工作，即对用户问题进行文本表示、分词、词性标注、去停用词等，本文主要对文本表示技术和中文分词技术展开阐述。

（1）文本表示

文本的表现形式通常是由文字和标点符号所构成的字符串[58]，由最小语言单元即字或字符组成词语，词语之间组合成为短语，从而形成句、段、节、章、篇等结构。然而这种形式的文本并不能高效地被计算机处理，因此需要对文本进行形式化描述，使其不仅能反映文档的真实内容，如文档主题、文档领域、组织结构等，还可以明确区分不同领域的文档。

目前在自然语言处理领域，最常用的方法是是 20 世纪 60 年代末由 G. Salton 等人提出的向量空间模型表示方法。VSM 最初应用于在信息检索系统中，目前已经成为计算机科学领域对文本处理的常用模型[59]。VSM 中涉及以下基本概念。

①文档。在整个文章中具有相当规模的片段可以成为文档，如句子、句群、段落、段落组直至整篇文章，即一组句群可称为一个文档，整篇文章也可成为一个文档。

②项/特征项。在 VSM 中将文本不可再分的最小语言单元定义为特征项，一般表现为字、词、词组、短语等。基于特征项的定义，可以将一个文档的内容看成是其含有的特征项所组成的集合，形式化描述为 $Document = D(t_1,t_2,\cdots,t_k,\cdots,t_n)$，其中 $t_k(1 \leqslant k \leqslant n)$称为该句子的特征项。

③项的权重：对于文档 $D(t_1,t_2,\cdots,t_k,\cdots,t_n)$，其含有 n 个特征项，根据特定原则给每一个特征项 t_k 赋予一个权重值 w_k，反映该特征项在文档中的重要程度。

基于权重的定义可以进一步将文档 D 表示为特征项及其所对应的权重，即 $D = D(t_1,w_1;t_2,w_2;\cdots;t_k,w_k;\cdots;t_n,w_n)$，简记为 $D = D(w_1,w_2,\cdots,w_k,\cdots,w_n)$，其中 w_k 是特征项 t_k 的权重，满足 $1 \leqslant k \leqslant n$。由此，可以将文档 D 看成是 n 维空间中的一个向量，即向量空间模型，其形式化如定义 2.1 所示。

定义 2.1 向量空间模型 给定一个文档 $D=D(t_1,w_1;t_2,w_2;\cdots;t_k,w_k;\cdots;t_n,w_n)$，$D$ 符合以下两条约定：

①各个特征项 $t_k(1 \leqslant k \leqslant n)$互异；

②各个特征项 t_k 顺序无关，即不考虑文档内部结构。

在以上两个约定下，特征项$t_1,t_2,\cdots,t_k,\cdots,t_n$ 相当于 n 维坐标系，而坐标系中对应的坐标值即为各个权重 $w_1,w_2,\cdots,w_k,\cdots,w_n$。因此，一个文本就可以表示为 n 维空间中的一个向量，称 $D=D(w_1,w_2,\cdots,w_k,\cdots,w_n)$为文本 D 的向量表示或向量空间模型，如图 2.4 所示。

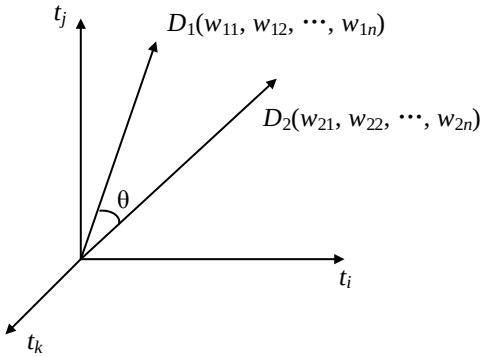

图 2.4 文档向量空间模型示意图

（2）中文分词技术

词是最小的能够独立运用的语言单位，而很多孤立语和黏着语（如汉语、日语、藏语等）的文本不像西方曲折语的文本，词与词之间没有任何空格之类的显式标志指示词的边界，因此，自动分词问题就成了计算机处理孤立语和黏着语文本时面临的首要基础性工作，汉语分词是指通过数据处理等技术让计算机系统在汉语文本中的词与词之间自动加上空格或其他边界标记。

目前成熟的汉语分词方法主要可以分为词典分词方法、统计分词方法、理解分词方法三类[60]。其中词典分词方法主要依据既定的词典结构将待分词字串与之依次匹配，若在词典中找到某词，则匹配成功，输出该词并加以边界标记。该方法受分词算法与词典结构影响较大，目前较成熟的方法有正向最大匹配算法、逆向最大匹配算法、双向最大匹配法等。理解分词方法主要使用神经网络技术同时对文本进行语义分析和句法分析，有效处理文本中的歧义字段，该方法由于使用了神经网络可以较好的适应不断变化的中文文本，同时可以通过不断学习处理未登录词。统计分词方法统计语料库中共现汉字的组合频率将其作为分词依据，常见统计分词方法主要有最大熵分词模型、基于词的 N 元语法模型、有向图模型等。

基于以上方法国内不少高效和科研院所都开发了独立的分词系统，目前较为成熟的几款分词系统主要有 NLPIR 汉语分词系统、SCWS 分词系统、盘古分词、IKAnalyzer、jieba 分词、腾讯分词、语言云等。本文从分词准确度、歧义词切分、分词速度、未登录词等方面对以上系统的性能进行比较，如表 2.2 所示。

表 2.2　主要分词系统性能比较

分词系统	分词粒度	支持处理字符	未登录词识别	词性标注	接口
NLPIR	多选择	中文简体及繁体	有	有	多语言接口
SCWS	多选择	中文	有	有	命令行工具
盘古分词	多选择	中文简体及繁体	有	无	无
IKAnalyzer	多选择	兼容日、韩文字符	有	无	jar 包
Jieba 分词	多选择	中文简体及繁体	有	有	Paython 库
腾讯分词	小	中文简体及繁体	有	有	API
语言云	适中	中文简体及繁体	有	有	API

综合来看，NLPIR 分词系统由中国科学院计算技术研究所开发，使用较为方便，同时具备未登录词识别和词性标注，分词粒度动态可调整，因此本文在问答系统问句预处理阶段选用 NLPIR 分词系统对用户问答进行中文分词操作。

2.2.3 句子相似度计算

句子相似度计算是智能问答系统的重要技术之一，句子相似度指两个句子之间词语的可替换度以及词义的符合程度[61]，是用来评估两个句子之间差异的大小的指标。两个句子若相似，该两者必定具有某些相似或相同的属性，令 $S=Sim(S_1,S_2)$是句子 S_1、S_2 之间的相似度，则 S 满足以下几个条件：

①$S\in[0,1]$且 $S\in R$，表示句子 S_1 和 S_2 相似且相似度为 S；

②$S=0$，当且仅当句子 S_1 和 S_2 之间没有任何相同属性，表示句子 S_1 和 S_2 不相似；

③$S=1$，当且仅当句子 S_1 和 S_2 所具备的属性完全相同，这种情况下 S_1 和 S_2 具备相同的句子结构和语义信息；

④$Sim(S_1, S_2)= Sim(S_2, S_1)$，$Sim(S_1, S_1)=1$，即句子相似度具备对称性及自反性。

Lin 等人认为句子相似度与句子之间的共性有关，共性越大则差异越小，相似度越高，从信息论的角度给出了相似度的计算如公式（2.1）所示。

$$Sim(S_1,S_2)=\frac{\log P(common(S_1,S_2))}{\log P(description(S_1,S_2))} \tag{2.1}$$

式中，$common(S_1,S_2)$是 S_1 和 S_2 的共性信息，$description(S_1,S_2)$描述 S_1 和 S_2 的全部信息，由公式（2.1）可以看出，文本相似度与文本共性正相关。

由于句子相似度涉及到句子结构、语言、句法、词法等因素，近年来不断有新方法涌现，目前较为成熟的相似度计算方法可以分为四大类，分别是基于字符串的方法、基于语料库的方法、基于语义的方法、基于句法分析的方法，如图 2.5 所示。

（1）基于字符串的方法

基于字符串的方法又称字面相似度算法，根据计算粒度的不同可进一步分为基于字符的方法和基于词的方法。其中常见的编辑距离、LCS、N-Gram 等方法都属于基于字符的方法；欧氏距离、Jaccard 等则属于基于词的方法。基于编辑距离的方法主要指 S_1 转化到 S_2 需要删除、插入、替换操作的最少次数，计算准确，但较为费时；

LCS 即最长公共子串，以两个句子共现且最长的子字符串作为相似度，其原理较为简单，但不适用于文档等长文本相似度计算；N-Gram 利用了集合思想，将相似 *n* 元组总量与 *n* 元组总量之比作为相似度，该方法中 *n* 值可调，相对灵活，但同样不适用于长文本。欧氏距离将两个句子间距离直接看作向量，计算向量的自然长度；Jaccard 采用集合思想，对两个句子进行简单的交并运算，这两种方法简单直接，但效果不佳。

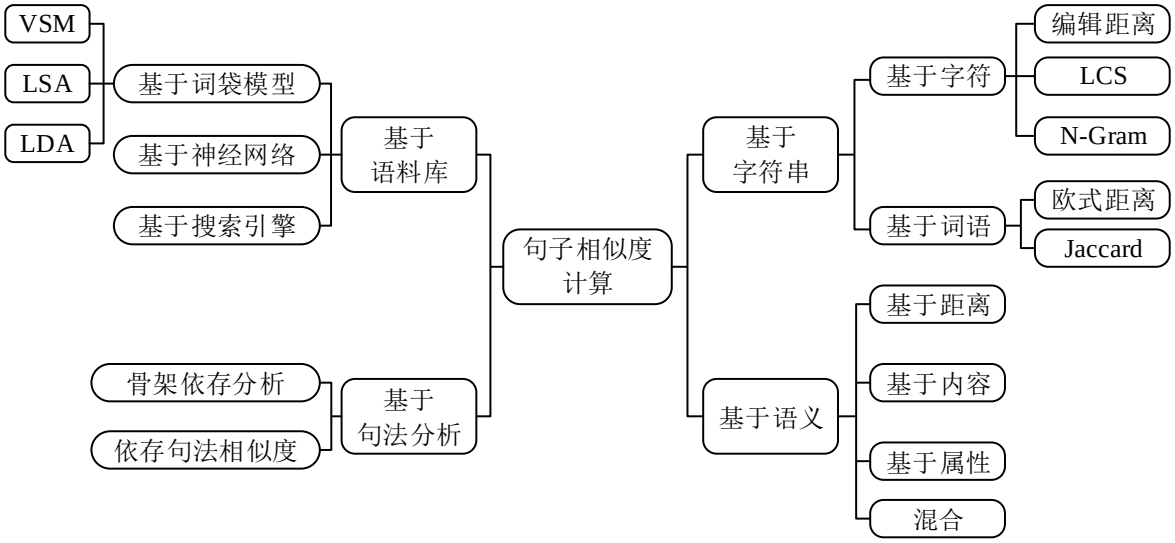

图 2.5　句子相似度计算方法分类

（2）基于语料库的方法

基于语料库的方法从大规模语料库中获取信息计算句子相似度，可进一步分为基于词袋模型的方法、基于神经网络的方法、基于搜索引擎的方法。词袋模型将文本表示成一系列无序词语的组合，较常用的 VSM 相似度就是词袋模型的一种。

余弦相似度是基于 VSM 模型所提出的文本相似度计算方法，将文本置于向量空间，则 VSM 中两个向量的相似度可以用向量夹角的余弦值来度量。根据向量的坐标值将其绘制到向量空间中，利用向量之间的夹角计算夹角的余弦值，使用余弦值表示两个向量的相似性。相似度随夹角大小变化而变化，夹角越小则说明向量方向更吻合，其余弦值越趋于 1，则两个向量越相似。如图 2.6 所示，向量 ***a*** 与 ***c*** 夹角大于向量 ***a*** 与向量 ***b*** 夹角，前者余弦值更小，则 ***a*** 与 ***b*** 具有更高的相似度。

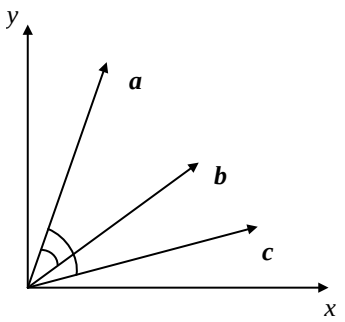

图 2.6　向量的余弦相似度

基于句子描述模型 VSM 将句子 S_1 和 S_2 表示为两个向量：

$S_1 = S_1(w_{11}, w_{12}, \cdots, w_{1n})$

$S_2 = S_2(w_{21}, w_{22}, \cdots, w_{2n})$

则句子的余弦相似度计算如公式（2.2）所示。

$$Sim(S_1, S_2) = \cos\theta = \frac{\sum_{k-1}^{n} w_{1k} \times w_{2k}}{\sqrt{\left(\sum_{k-1}^{n} w_{1k}^2\right) \times \left(\sum_{k-1}^{n} w_{2k}^2\right)}} \tag{2.2}$$

基于神经网络的方法是通过神经网络生成词向量计算本文相似度，与词袋模型不同之处在于表达文本的方式，神经网络将文本表示成向量，向量的维数可以人为修改，所以这类方法效果更好。基于搜索引擎的方法以 Web 数据作为语料库，最为成熟的方法是 Cilibrasi 等人提出的归一化谷歌距离，该方法给定一对搜索关键词，搜索引擎返回相关网页数量，以网页数量差之比计算相似度。之后也有不少学者在归一化谷歌距离的基础上进行改进，然而这类方法计算结果与搜索引擎的效率有直接关系，相似度结果也因搜索引擎不同呈现较大差异。

（3）基于语义的方法

随着语义 Web 等技术的发展，一批学者将语义 Web 技术应用于文本相似度计算，涌现出一些较好的基于语义的相似度计算方法。语义 Web 中本体能够准确的描述概念含义，反映概念间内在关系，利用语义 Web 中的本体技术，以本体中概念间的上下位和同位关系为基础计算概念间语义距离或路径，进而计算语义相似度。由于本体结构存在一定的特殊性，相似度的结果与概念节点在本体中的节点深度、密度、强度等因素有关，目前很少有学者能全方位综合考虑所有特征加以计算，因此也衍生出一系列不同算法。基于距离的方法综合考虑节点密度、深度，语义距离通过本体中概念之间的路径长度来表示，利用语义距离与相似度的反比关系计算相似度；基于内容的方法则采用了不同节点的信息量，使用概念词共享的信息量化其语义距离；基于属性的方法则考虑到了本体结构中概念的属性特征，一般以概念词之间的公共属性的数量表述概念之间的语义相似度；另外也有一些学者将以上三种因素综合考虑，给每种因素赋予一定的权重，但权重设置依赖于本体结构和一些领域专家。

（4）基于句法分析的方法

以上三种方法大多数是以词语为粒度进行计算，较少关注句法层面，即词语的组合方式和组合内涵，然而相同词语的不同组合形式有时也会产生较大差异的内涵，因此基于句法分析的相似度计算也十分重要。基于句法分析的相似度的基本思想是利用自然语言处理的中文分词、词性标注等技术手段对句子进行依存句法分析，得到句子各个成分之间的依存关系，再利用依存关系的有效配对计算相似度。骨架依存分析法就是句法相似度计算的一种，其基本思想是经词性标注后分析句子成分，并以依存树的形式表达，进而比较骨架依存树计算相似度，此方法计算结果较为精确，但由于句子结构复杂，因此框架分析存在一定难度，导致相似度计算结果不甚理想。

2.3 本章小结

本章主要阐述了语义 Web 与智能问答系统相关研究，概述了语义 Web 技术，阐述了语义 Web 分层模型及相关基础技术，针对本体技术展开详细论述，说明了本体的描述语言，详细介绍本体的构建及查询方法。论述了问答系统的体系结构，说明问答系统包含的五个模块即数据源分析模块、问题分析模块、知识检索模块、答案生成模块和答案评估模块。最后针对问答系统中的两个关键技术即问句预处理和句子相似度计算展开详细研究，问句预处理研究了文本表示和中文分词技术，句子相似度计算则对现有相似度计算方法进行分类，并分别展开详细论述。

第 3 章　多特征融合的句子相似度计算方法研究

句子相似度计算是智能问答系统理解用户问题的关键，在基于语义理解的智能问答系统中直接影响着问答系统检索结果的准确率。目前针对句子相似度已经有多种计算方法，如基于编辑距离的方法、基于关键词的方法、基于余弦距离的方法、基于向量空间模型的方法等。这些方法计算简单，可以从简单的结构信息层面进行句子的相似度计算，但却忽略了句子的语义信息，有时甚至得到与事实相悖的结果。本章深入研究句子相似度计算，提出了一种多特征融合的相似度计算方法。

3.1 问题描述

句子相似度计算作为自然语言处理的研究重点，已经广泛应用于文本分类、智能问答、信息检索、文本挖掘等领域。句子的语义相似度是计算语言学中的一个度量，表示依赖于它们的层次关系的两个概念的共性。语义相似度不等同于语义相关度，语义相似度可以看作语义相关度的特例。语义相似的两个概念本身之间就具有某些共性，而语义相关度则指两个概念之间存在某种关系，但有可能概念之间并不相似，概念之间的相关关系一般是通过某些其他关系相关联所形成的，如概念“笔记本”和“计算机”具有语义相似度，“笔记本”和“鼠标”具有语义相关度。

Mihalceal 等人[62]提出使用词袋模型表示句子，Oliva 等[63]提出使用词汇句法信息树表示句子。研究学者基于以上表示方法，设计相似度计算方法计算概念间语义相似度，从而构造特定的函数计算句子的整体相似性，但不能解决如下两个问题。

①词意问题：即由不同概念词构造的相同意义的句子，例如，句子“Peter is a handsome boy”和句子“Peter is a good-looking lad”在其出现的上下文变化不大的情况下具有相同的含义。

②词序问题：句子中概念词出现的顺序会对句子含义造成一定影响。例如，句子“A like B”和句子“B like A”，虽然这两个句子具有相同的词汇，但其概念词出现的顺序不同使得这两个句子表达了完全相反的含义。

由于汉语句子结构多变且存在一词多义的情况，且大部分现有句子相似度计算方法对句子的语义挖掘只停留在句子的关键词层面，存在一定的局限性。针对以上研究背景，本文以句子的形态结构、语序结构、语义信息等特点为核心要素，并以本体作为知识库充分挖掘句子语义信息。将句子相似度分为句子结构特征和语义特征两个部分，提出一种多特征融合的句子语义相似度计算方法，通过结构特征和语义特征的加权融合解决句子之间的相似度计算问题。

3.2 多特征融合的句子相似度算法设计

3.2.1 结构相似度计算

传统基于关键词的相似度计算方法在对句子进行分词、去停等操作后，统计句子的关键词，句子的相似度即关键词之间的相似度。然而除关键词外，其他结构特征如句长、词序、词形等要素也是应考虑的重要因素。本文将句子的词形、词序、句长三个特征相结合，共同计算句子相似度。

两个句子在词语形态上的相似程度即为词形相似度（Morphological Similarity），定义为共有词汇数与句子的长度和之比。词形相似度 $MorSim(S_1,S_2)$的计算如公式（3.1）所示。

$$MorSim(S_1,S_2)=\frac{2\times Com(S_1,S_2)}{Len(S_1)+Len(S_2)} \tag{3.1}$$

其中，$Com(S_1,S_2)$是句子 S_1 和 S_2 经分词结果后共同拥有的特征项个数，如果某一特征项在句子 S_1 和 S_2 中出现超过一次，则以出现次数的最小值作为 $Com(S_1,S_2)$的值。$Len(S_1)$和 $Len(S_2)$即句子 S_1 和 S_2 的长度，分别表示句子 S_1 和 S_2 的总词数（特征项的总个数）。显然，如果句子的共现词汇数目较高，则句子的相似度很可能较高。

词序相似度（Order Similarity）表示的是两个句子中关键词的相对位置关系。将 S_1 和 S_2 中均出现且只出现一次的词汇集合定义为 $Once(S_1,S_2)$，s 代表 $Once(S_1,S_2)$集合中词汇的个数。则词序相似度 $OrdSim(S_1,S_2)$计算如公式（3.2）所示。

$$OrdSim(S_1,S_2)=\begin{cases}1-\dfrac{AIN(S_1,S_2,s)}{s-1} & s>1\\ 0 & s=0\\ \dfrac{1}{2} & s=1\end{cases} \tag{3.2}$$

假设将集合 $Once(S_1,S_2)$中句子 S_1 的词汇顺序定义为标准序列，则 $AIN(S_1,S_2,s)$代表句子 S_2 中词汇的逆序数，假设标准序列为“1432”，则逆序为“43”和“32”。如果共现词汇集合为空，词序相似度定义为 0；如果 S_1 和 S_2 的共现词汇数为 1，词序相似度定义为$\frac{1}{2}$。

在 $MorSim(S_1,S_2)$和 $OrdSim(S_1,S_2)$中，句子关键词是核心要素，不能完整反映句子信息，因此引入句子的句长特征。句子相似度与两个句子长度的差值成反比，长度差值越大，相似度越小。句长相似度 $LenSim(S_1,S_2)$计算如公式（3.3）所示。

$$LenSim(S_1,S_2)=1-\frac{abs(Len(S_1)-Len(S_2))}{Len(S_1)-Len(S_2)} \tag{3.3}$$

式中，$abs(\ )$为绝对值函数，计算两个句子长度差值的绝对值。

综合词形相似度、词序相似度、句长相似度三种特征，得出句子结构相似度

$StrSim(S_1,S_2)$计算模型如公式（3.4）所示。

$$StrSim(S_1,S_2)=\alpha\times MorSim(S_1,S_2)+\beta\times OrdSim(S_1,S_2)+\gamma\times LenSim(S_1,S_2) \quad (3.4)$$

其中，α、β、γ 分别是词形相似度、词序相似度及句长相似度的权重值，$0\le\alpha\le1$，$0\le\beta\le1$，$0\le\gamma\le1$，且满足 $\alpha+\beta+\gamma=1$。

本文采用 AHP（Analytic Hierarchy Process，层次分析法）进行结构相似度中各个特征权值的计算。层次分析法首先将与决策有关的元素分解，形成目标、准则等层次，之后基于层次结构模型进行定性和定量分析[64]。本文运用层析分析法计算权重步骤如下。

（1）建立递阶层次结构模型

依据公式（3.4）将结构相似度计算问题分解为词形相似度、词序相似度、句长相似度，构造出如图 3.1 所示的层次结构模型，分为目标层和准则层，目标层对应句子结构相似度，准则层对应词形、词序及句长相似度。

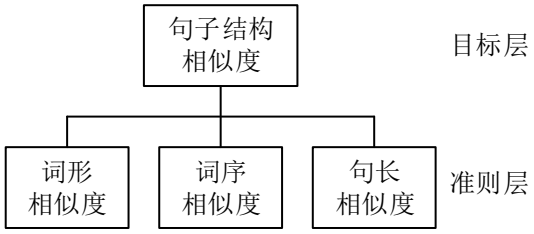

图 3.1　结构相似度层次模型

（2）构造各层次中的判断矩阵

将层次模型准则层中的各个准则相对于结构相似度的重要程度进行对比，得出准则层中各个准则所占比重。词形、词序、句长三个特征相比，此词形和词序特征是影响句子相似度的主要因素，其重要程度相同，相对而言，句长特征对句子结构相似度影响程度较小。根据文献[64]，引用数字 1 至 9 及其倒数作为标度，构造出判断矩阵如表 3.1 所示。

表 3.1　判断矩阵

	词形特征	词序特征	句长特征
词形特征	1	1	5
词序特征	1	1	5
句长特征	1/5	1/5	1

（3）层次单排序及一致性检验

依据表 1 可得判断矩阵

$$A=\begin{bmatrix}1 & 1 & 5\\ 1 & 1 & 5\\ 1/5 & 1/5 & 1\end{bmatrix}$$

计算出该矩阵的最大特征值 $\lambda_{max}=3$，对应特征向量为 $\boldsymbol{p}=[5,5,1]^T$。

定义一致性指标 *CI*（Consistency Index）如公式（3.5）所示

$$CI = \frac{\lambda_{\max} - n}{n-1} \tag{3.5}$$

其中，$\lambda_{\max}$ 是最大特征值，n 是特征向量的维度。依据矩阵 A 计算可得 $CI = 0$。查找平均随机一致性指标 *RI*（如表 3.2 所示）得 *RI*=0.52。

表 3.2　平均随机一致性指标

n	1	2	3	4	5	6	7	8
RI	0	0	0.52	0.89	1.12	1.24	1.36	1.41

计算一致性比例 *CR*（Consistency Ratio）如公式（3.6）所示。

$$CR = \frac{CI}{RI} \tag{3.6}$$

得 $CR = 0$，根据检验系数评判标准可知，当 $CR<0.1$ 时，判断矩阵的一致性可接受，因此上述判断矩 A 所对应的特征向量可以作为相似度权值。对特征向量 $\boldsymbol{p} = [5,5,1]^T$ 做归一化处理，得到权值向量为 $W = [0.455, 0.455, 0.09]$，即公式（3.4）中各个权值分别为 $\alpha = 0.455$，$\beta = 0.455$，$\gamma = 0.09$。

3.2.2 语义相似度计算

句子的语义信息通常由句子中具有实在意义的实词反映，语义 Web 中的本体包含了大量概念节点，蕴含丰富的语义信息并具备一定的推理能力。本文将句子中的实词对应到本体图中的概念节点，使用概念节点之间的语义距离计算概念间语义相似度，句子语义相似度由概念相似度计算得到。

本体树状结构中某两个概念节点所含有的相同上位概念节点个数称为语义重合度，反映了两个概念的相似程度。两个概念节点通路中最短路径的长度定义为语义距离，语义距离与语义相似度呈反比关系，即两个概念的语义距离越大，则语义相似度越小。特别地，当两个概念的语义距离 $SemDis(C_1,C_2) = 0$ 时，语义相似度 $SemSim(C_1,C_2) = 1$；当语义距离 $SemDis(C_1,C_2) = \infty$ 时，语义相似度 $SemSim(C_1,C_2) = 0$。概念节点与根节点的距离越大，概念包含的语义信息越丰富，同时，距离较近的概念之间具有更多的共同特征，语义相似度就越大。例如图 3.2 中，C_1、C_6 间隔 2 条边，C_1、C_3 间隔 1 条边，所以 $SemSim(C_1,C_6) < SemSim(C_1,C_3)$。

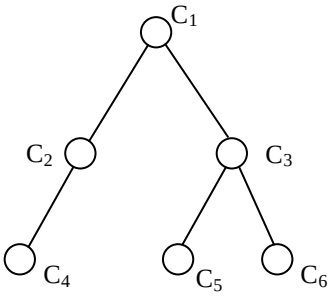

图 3.2　路径长度对语义相似度的影响

由 2.2.2 小节可知，一个句子可以看成由多个项所组成的集合，然而这些项（词）中包含冠词、介词、连词、助词等。这些虚词不包含语义信息，甚至对句子的语义相似度计算造成负面影响。因此在进行相似度计算时，经分词、去停操作，删除无意义虚词，保留实词作为特征项，构成句子的描述模型。基于 2.2.2 小节对现有成熟分词系统性能比较，本文选取中科院计算技术研究所开发的分词系统 NLPIR 进行分词和词性标注。停用词指对句子含义无帮助的一些无意义虚词，如“啊”、“吧”、“的”等，停用词会干扰句子分析结果，在进行处理时应该剔除。假设待比较相似度的句子 S_1、S_2，经分词、去停用词、词性标注等处理后，从剩余关键词中抽取主语和宾语的实词组成集合为 $S_1=\{C_{11},C_{12}\}$，$S_2=\{C_{21},C_{22}\}$，之后本文采用概念间语义距离计算句子间语义相似度。

本文首先计算概念间语义距离，得到概念语义相似度，之后设计算法计算句子语义相似度。给本体中概念间关系分配权重，之后使用最短路径算法计算概念节点与根节点间的最短路径，以最短路径定义语义距离，最终通过语义距离计算句子的语义相似度。主要步骤描述如下。

步骤 1　本体的权重分配

在计算语义距离时，不少方法将本体图中概念节点的权重考虑在内，然而概念节点的权重在一定意义上只反映出概念所包含信息量的多少，并不适应于语义距离计算，因此本文给概念节点之间相连的关系定义权重，反映概念间关系对语义距离的贡献程度。

本体中概念间关系可以分为“*is-a*”关系与“*part-of*”关系，其中，“*is-a*”关系与相似度有关，而“*part-of*”关系与相关度有关，因此本文只考虑“*is-a*”关系，即两个概念为继承关系的情况。使用公式（3.7）给本体中的概念间关系分配权重，概念节点 *m* 和 *n* 之间的关系的权重值记为 *W*(*m*,*n*)。

$$W(m,n)=\left[depth(m)+\frac{Order(n)}{N(G)+1}\right]^{-1} \tag{3.7}$$

式（3.7）中，*m*,*n* 为本体中两个直接相连的概念节点，*depth*(m)代表节点 *m* 的最大深度，*N*(*G*)代表概念节点总数，*Order*(*n*)代表概念节点 *n* 在其兄弟概念节点中的顺序数，*Order*(*n*)≥0 且为整数。*depth*(*m*)是为了确保当前关系的权重总是比起前一个关系的权重值小，定义如下：

$$depth(m)=\begin{cases}0, m\text{是根节点}\\ 1, m\text{是根节点直接相连的节点}\\ i, m\text{是与根节点相距}i\text{个节点的节点}\end{cases}$$

步骤 2　计算最短路径

根据步骤 1 对本体图中的概念间关系的权重分配可知，分配权重之后的本体具有两点特征，即权值严格大于 0 且节点之间关系是“*is-a*”关系。因此将本体图看作一

个带权有向图，使用最短路径算法思想计算概念节点距根节点的最短路径。以公式（3.8）对概念节点关系进行权重初始化，初始节点赋值为 0，其它节点置为无穷，然后使用公式（3.9）计算初始节点与根节点之间的最短路径。

$$W_0(m,n)=\begin{cases}0, m=n\\ \infty, m\neq n\end{cases} \tag{3.8}$$

$$W_{k+1}(m,n)=\min\begin{cases}W_k(m,n)\\ W_{k+1}(m,x)+W(x,n)\end{cases}, 0\le k\le S-1 \tag{3.9}$$

式中，*m,n,x* 代表三个节点，*x* 和 *n* 是直连节点，*S* 为本体中所有节点集合，$W_k(m,n)$ 代表迭代 *k* 处路径(m,n)的权重。

改进后的最短路径算法伪代码如表 3.3 所示。

表 3.3　最短路径算法

算法 3.1　最短路径算法
输入：Ontology //本体图 RootNode //本体图根节点 Node //概念节点 输出：shortestPath //最短路径 1. S[N] ← Ontology //本体图中所有节点存入集合 S 2. FOR i ← 1 TO N DO //初始化 3. IF *S*[*i*]==*Node* THEN 4. *W*(*Node*,*i*)=0 5. ELSE 6. *W*(*Node*,*i*)=∞ 7. END IF 8. *S*[*i-1*]=*NULL* 9. END FOR 10. WHILE *S*!=*NULL* DO 11. *s*[*i*] ← *S* 中最小权重节点 12. FOR *S*[*i*]的每一个后继节点 DO 13. IF *W*(*Node*,*i*)+*W*(*i*,*i+1*)<*W*(*Node*,*i+1*) THEN 14. *W*(*Node*,*i+1*)=*W*(*Node*,*i*)+*W*(*i*,*i+1*) 15. *S*[*i-1*] ← *S*[*i*] 16. IF *S*[*i-1*]==*RootNode* THEN 17. RETURN *shortestPath* 18. END IF 19. END IF 20. END FOR 21. *S*[*N*] ← *S*[*N*]-*S*[*i*] 22. END WHILE

步骤 3　计算语义距离

在语义距离的计算过程中，由于前文计算的最短路径均为概念节点到根节点，因此，两个概念之间的语义距离不能是上述最短路径之和。本文方法是删除最短路径中的共有部分，计算两个概念之间的语义相似度，如式（3.10）所示。

$$SemDis(C_1,C_2)=W_{shortestPath1}+W_{shortestPath2}-2\times W_{comShortestPath} \tag{3.10}$$

其中，C_1,C_2 代表本体图中的两个概念节点，*shortestPathi* 表示概念节点 C_i 距根

节点的最短路径。*comShortestPath* 代表 C_1 和 C_2 之间的第一个公共节点距根节点的最短路径，其计算如式（3.11）所示。

$$W_{comShortestPath}=\sum_{j-1}^{k}W_j(m,n) \tag{3.11}$$

式中，*m,n* 代表 *shortestPath* 中直连的两个节点，*k* 是其中节点关系的集合。

步骤 4　计算语义相似度

根据语义相似度的定义可知，语义相似度与语义距离之间呈反比关系，且满足以下三个条件。

① $\forall(C_1,C_2)\in G:0\le SemSim(C_1,C_2)\le 1$

② $\forall C_1\in G:SemSim(C_1,C_1)=1$

③ $\forall(C_1,C_2,C_3)\in G:$

$if\ \ SemDis(C_1,C_2)>SemDis(C_1,C_3)\ then$

$SemSim(C_1,C_2)<SemSim(C_1,C_3)$

其中，$SemSim(C_1,C_2)$表示概念 C_1 和 C_2 之间的语义相似度，$SemDis(C_1,C_2)$表示概念 C_1、C_2 间语义距离，(C_1,C_2,C_3)为本体图 G 中的三个概念，C_i 表示所给节点与根节点之间最短路径的节点集合。概念 C_1 和 C_2 间的语义相似度定义如公式（3.12）所示。

$$SemSim(C_1,C_2)=\frac{1}{SemDis(C_1,C_2)+1} \tag{3.12}$$

基于以上分析，句子语义相似度算法描述如表 3.4 所示。

表 3.4　句子语义相似度算法

算法 3.2　句子语义相似度算法
输入：S_1,S_2 //S_1,S_2 为待比较相似度的两个句子 Ontology //本体图 RootNode //本体图根节点
输出：SemSimilarity // S_1 和 S_2 的语义相似度
1.　S_1 经分词、去停后得到概念词 a_1,a_2
2.　S_2 经分词、去停后得到概念词 b_1,b_2
3.　FOR i ← 1 TO 2 DO
4.　　FOR j ← 1 TO 2 DO
5.　　IF a_i=b_j THEN
6.　　SemDis ← 0
7.　　ELSE IF a_i 与 b_j 直连 THEN
8.　　SemDis ← W(a_i,b_j)
9.　　ELSE
10.　　SPath1 ← shortestPath(Ontology, a_i, RootNode)
11.　　SPath2 ← shortestPath(Ontology, b_j, RootNode)
12.　　SDis←W_{SPath1}+W_{SPath2}-2*$W_{comShortestPath}$
13.　　END IF
14.　　SemSim ← 1/(SDis+1)
15.　　END FOR
16.　END FOR
17.　SemSimlarity ← SemSim 求和平均
18.　RETURN SemSimlarity

3.2.3 多特征融合的语义相似度计算

句子的结构相似度和语义相似度分别从结构层面和语义层面表达了不同的分析句子的观点，基于各个层次的相似性度量只是单一角度的相似度值。为全面分析句子之间的相似度，本文将以上两种相似度结合，综合考虑句子的结构信息和语义信息，提供句子相似性的整体度量。特征融合的句子相似度计算模型如公式（3.13）所示。

$$Sim(S_1,S_2)=a\times StrSim(S_1,S_2)+b\times SemSimilarity(S_1,S_2) \quad (3.13)$$

其中，a、b 是可调节参数，分别代表结构相似度和语义相似度的权重值，$0\le a\le 1$，$0\le b\le 1$，满足 $a+b=1$。为了验证结构特征和语义特征对句子相似度的影响，此处不采用层次分析法计算权重值，参数 a 与 b 的设定在实验中体现。

3.3 实验验证

3.3.1 实验环境

为了验证本文算法的性能，采用 Java 语言搭建了多特征融合的句子语义相似度计算原型系统，具体实验环境如表 3.5 所示。

表 3.5　实验环境配置

项目	配置
操作系统	Windows 8.1
开发语言及开发平台	Java + My Eclipse
本体建模工具	Protégé
分词工具	NLPIR

3.3.2 实验数据

目前国际上常见的句子相似度计算的测试集多为来自 SemEval 国际语义评测会议的公共测试集，测试集内所有句子均为英文，而中文因灵活性和结构的复杂性目前没有开放的公共测试集。因此本文参照 SemEval 提供的公共测试集，基于手工规则生成句子相似度计算的测试集。SemEval 所提供的相似度语义句子来源于多个领域，包括新闻、翻译评估、社区问答、图书等领域，该测试集中每一类数据源都包含数量不同的句子对，并由领域专家给每对句子进行相似度打分，分数越高，表明两个句子在语义上越接近，句子语义完全相同则其对应的分数为 5 分，分数为 0 则表明句子之间没有任何关系。

本文主要面向图书领域智能问答系统，因此本文筛选出 SemEval 测试集中与图书有关的句子对共 209 对，将其翻译成中文，并进行语义调整，使其合乎中文自然语言的逻辑。同时，本文以图书领域智能问答系统为背景，采集陕西省图书馆网站内读者留言 208 条，人工筛选去除无意义留言，剩余 126 条作为初始数据集。以该测试集中的每条句子为查询条件在百度搜索中进行查询，提取出前 10 条结果，经人工筛选出与初始集合中最相似的句子构成对比数据集，从而得到包含 126 对句子对。将从

SemEval 开放测试集的 209 对句子对与人工组建的 126 对句子对进行合并，构成本文实验的测试集合，共包含句子对 335 对。

本文邀请自然语言处理研究方向的 10 名研究生对这些句子组进行相似度判断，人工标记句子相似度为“0”和“1”，其中“0”表示不相似，“1”表示相似。对 SemEval 的开放测试集，参照原有 0~5 的相似度打分进行人工调整，对手工构建的测试集，进行人工语义判断给出相似度标记。经整理后的测试集格式如表 3.6 所示。

表 3.6　句子相似度计算测试集

ID	句子 1	句子 2	相似度标记
1	有李商隐的诗集有关的书吗？	关于李商隐的诗集，求推荐书籍	1
2	《计算机网络 自顶向下法》是哪个出版社的？	谁写的《计算机网络 自顶向下法》？	0
3	请问杨绛的著作有哪些？	杨绛写了那些书？	1

另外，实验抽取图书领域的实体及其关系，采用 Protégé 构建了一个基于图书领域的领域本体，作为句子语义相似度计算的本体知识库。该本体图共有实体 142 个，关系 506 对，包含图书与图书、图书与出版社、图书与作者等之间的关系对。图 3.3 给出了该本体中一个实体的代码片段，以该出版社实体为例，该实体名称为“作家出版社”，与图书实体“许三多冒险记”之间具有“hasPublish”关系，与作者实体“余华”之间具有“PayMoneyTo”关系。

```
<!-- http://www.semanticweb.org/missing/ontologies/2016/3/untitled-ontology-22#ZuoJiaChuBanShe -->
    <owl:NamedIndividual rdf:about="&BookQuery;ZuoJiaChuBanShe">
        <rdf:type rdf:resource="&BookQuery;Art_Pub"/>
        <rdfs:label rdf:datatype="&xsd;string">作家出版社</rdfs:label>
        <hasPublish rdf:resource="&BookQuery;CaoFangZi"/>
        <PayMoneyTo rdf:resource="&BookQuery;CaoWenXuan"/>
        <hasPublish rdf:resource="&BookQuery;HaoMaMaShengGuoHaoLaoShi"/>
        <hasPublish rdf:resource="&BookQuery;XuSanGuanMaiXueJi"/>
        <PayMoneyTo rdf:resource="&BookQuery;YiJianLi"/>
        <PayMoneyTo rdf:resource="&BookQuery;YuHua"/>
    </owl:NamedIndividual>
```

图 3.3　图书本体片段

3.3.3 实验结果分析

实验首先使用自然语言处理技术每个问句进行分词，经过去停、分词后，选取分词结果的实词作为关键词，之后先使用公式（3.4）计算结构相似度得到 $StrSim(S_1,S_2)$ 值，接着使用算法 2 计算语义相似度 $SemSim(S_1,S_2)$，最后采用公式（3.13）计算多特征融合的句子语义相似度得到 $Sim(S_1,S_2)$的值。公式（3.13）中的权重参数 a 和 b 的值在满足采取 $a + b = 1$ 的条件下动态调节，分别观察结构相似度和语义相似度对句子相似度的影响程度。

图 3.4 展示了参数值 a 从 0 到 1 变化时本文相似度算法在测试集上得出的句子相

似度。由图 3.4 可以看出，句子相似度的值在 a 取 0.3 之前呈增长趋势，在 0.3 之后呈下降趋势。可以得出结论，结构相似度的权重值越高，句子相似度越低；语义相似度的权重也不宜过高，否则句子整体相似度也越低，这也体现出了结构相似度和语义相似度对句子整体相似度的影响程度。因此，综合考虑句子的结构信息和语义信息，在之后的实验中选取权重值 $a = 0.3$，$b = 0.7$。

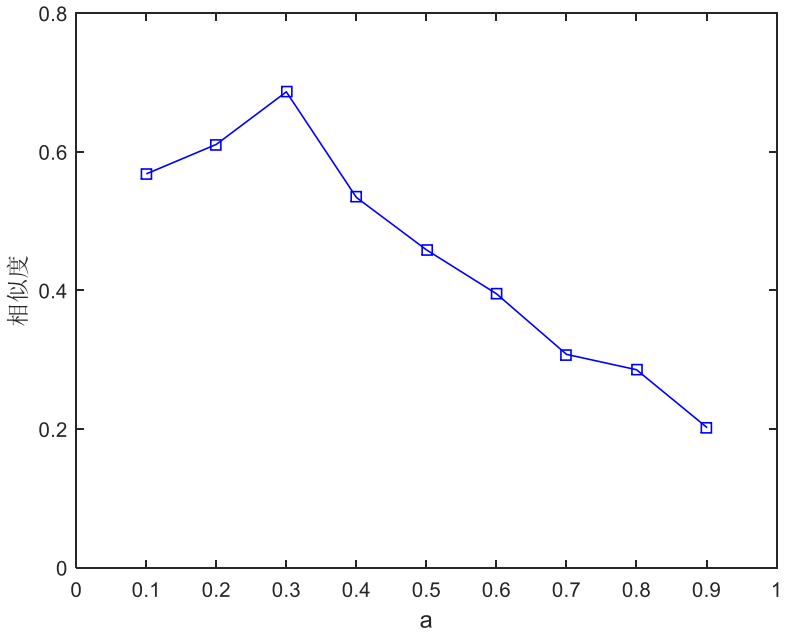

图 3.4　句子相似度结果

为客观体现本文算法的真实性能，同时选取两种具有代表性的相似度计算算法参与对比分析，分别是：

方法 1：余弦相似度算法。如 2.2.3 小节所述，余弦相似度将句子投影至向量空间，得到句子的向量表示，之后采用向量之间夹角的余弦值衡量句子的相似度。

方法 2：基于编辑距离的算法。将两个句子的相似程度定义为由一个转成另一个所需的最少编辑操作次数。

本文随机挑选 5 组句子，给出其相似度计算结果如表 3.7 所示。由表 3.7 可以看出，与方法 1 和方法 2 相比，本文相似度算法计算结果更精确，更接近事实。本文相似度方法计算句子组 4 的相似度远高于方法 1 和方法 2 的相似度，更符合实际认知。这是因为方法 1 将句子向量化，向量结构虽然可以充分考虑关键词信息，但并未考虑到关键词顺序对相似度的影响。而本文算法不仅考虑到句子的结构特征，并且将句子关键词映射至本体中，以概念在本体知识库中的语义距离计算语义相似度，在计算过程中同时考虑到关键词的顺序，循环计算概念之间相似度并求和平均。句子组 5 中两个句子之间以方法 2 计算的差异步骤共 18 步，基于编辑距离的相似度仅为 0.1，这是因为这两个句子虽然表达的语义一致，但是存在较大的结构差异，因此相似度较低。本文相似度算法在句子组 5 的计算结果比方法 1 低，这是因为本体知识库中并未添加

“年少荒唐”概念实例，导致在计算语义距离时出现偏差。另外，可以看出，若两个待比较句子经分词结果得到的多为实词，且可以直接在本体知识库中查询时，本文相似度算法性能更好，更接近实际情况。

表 3.7　句子相似度计算对比结果

测试句子	方法 1	方法 2	本文算法
我想问一下目前有没有《红岩》和《骆驼祥子》？ 《红岩》和《骆驼祥子》哪本书更好？	0.4636	0.3181	0.5465
省图书馆关于旅游方面的书都有哪些？ 关于旅游游记方面的书有哪些推荐？	0.6859	0.4705	0.7956
请问中文版的外国名著在哪里借啊？ 请问外国名著中英文对照做的比较好的出版社有哪些？	0.6196	0.25	0.3240
荣格所著《红书》找不到 荣格写的《红书》这本书没有	0.4622	0.4615	0.9006
《年少荒唐》这本书大概什么时候才能上架？ 什么时候才能有《年少荒唐》？	0.6030	0.1	0.5314

实验采用 F‐度量值（F‐Measure）进行性能评价，F‐度量值是均衡精确率和召回率的评价指标，是信息检索和统计学分类领域的标准指标。其中，准确率（Precision）为计算结果为相似实际也相似的句子数量与所有句子对总数的比率，衡量的是查准率；召回率（Recall）为计算结果为相似实际也相似的句子数量与数据集中所有相似句子对总数的比率，衡量的是查全率。F‐度量值是准确率和召回率的调和平均值，如果 F‐度量值越接近于 1，则说明 Precision 和 Recall 均衡得越好，相反，如果 F‐度量值越接近于 0，这说明两个参数的均衡性越差。

其计算公式如式（3.14）所示。

$$F-Measure=2\times\frac{\mathrm{Pr}ecision\times\mathrm{Re}call}{\mathrm{Pr}ecision+\mathrm{Re}call} \tag{3.14}$$

其中：

$$\mathrm{Pr}ecision=\frac{\text{正确检测到的相似文本数}}{\text{所有检测到的相似文本数}}$$

$$\mathrm{Re}call=\frac{\text{正确检测到的相似文本数}}{\text{实际存在的相似文本数}}$$

考虑相似度阈值对 F‐度量值的影响，动态调节阈值计算 F‐度量值，结果如图 3.5 所示。由图 3.5 可以看出，本文提出的多特征融合的句子语义相似度算法性能最优，余弦相似度算法次之，基于编辑距离的相似度较差。在不同阈值的情况下，余弦相似度结果一直高于基于编辑距离的相似度，这是因为测试集收集图书馆读者留言咨

询问句长度不一，存在很多口语化内容，并且很多句子尽管关键词顺序不同但其表达意思相同，因此词形特征和词序特征使得编辑距离的相似度表现略差。随着阈值的增加，三种算法计算的 F - 度量值都不断增加，在相似度阈值为 0.7 时，三种方法的 F - 度量值都较高，实验效果最好。同时，本文算法由于在考虑词形、词序、句长等结构特征的基础上引入句子的语义信息，F - 度量值略高于其他两者，相似度结果更加精确，更接近于实际情况。

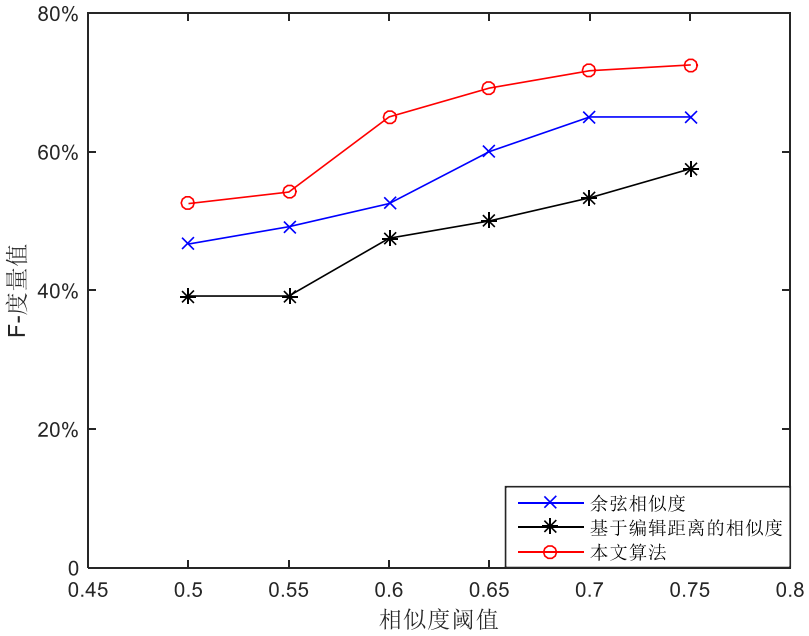

图 3.5　不同方法在不同阈值下的 F-度量值

3.4 本章小结

本章首先就句子相似度计算现存问题进行描述，说明语义相似度计算研究的必要性。提出了一种多特征融合的句子语义相似度计算方法，将句子相似度分为结构相似度和语义相似度，结构相似度中综合了句子的词形、词序和句长特征，语义相似度则使用本体图中概念节点间最短路径定义，两方面特征综合加权计算句子相似度。实验结果表明，与传统余弦相似度算法及基于编辑距离的相似度算法相比，本章所提算法的 F - 度量值可达 72.5%，相似度结果较为精确，更符合事实认知。

第 4 章　基于语义的问句查询扩展方法研究

上一章就句子的相似度计算展开了详细的阐述，所提算法主要用在基于问答对的问答模块。在基于语义理解的问答中，如果出现 FAQ 问题库中没有的问句，则需要在本体库中展开查询。本章提出一种基于本体的问句查询扩展方法，首先将查询关键词映射到本体中，利用改进的最小查询生成树算法构造查询子树生成扩展查询集合，基于相似度对扩展集合进行筛选最终生成查询关键词扩展集合。

4.1 问题描述

针对传统信息检索系统查准率、查全率过低的问题，Rigsbergen 结合计算语言学、信息学等技术提出了查询扩展的思想。查询扩展主要思想是在原始查询中加入与用户查询有关的词[65]，构成一个新的查询集合，进而展开二次查询。通过查询扩展，信息检索领域的查询词错配问题得到了有效解决，弥补了用户查询意图不明确的缺陷，查询系统的性能得到了有效提升。

查询扩展方法可分为全局分析法、局部分析法、基于语义的方法等。全局分析法首先自动分析文档中的全部词和词组，使用相似度计算词组之间的相关程度，将相关程度按照一定的依据进行排序，将与用户查询词关联程度最高的加入原查询，从而生成新的查询词展开二次查询。其缺点是需要构建查询词集合并分析全部文档，计算量较大。为了解决全局分析法中词汇表构建工作量巨大的问题，学者提出了局部分析法，该方法将初次检索之后的相关文档进行排序，选取最相关 *n* 篇作为查询扩展数据源，但是局部分析法的二次扩展查询存在偏离用户原始查询主题的缺点。

随着基于语义 Web 的问答系统的发展，不少学者提出了基于语义的查询扩展。本体可以在语义和知识层面上对数据进行描述，具有良好的概念层次结构，同时也支持逻辑推理，将富有丰富语义信息的本体应用于查询扩展，可通过查询词的语义扩展获得更加全面和准确的结果。使用本体作为查询扩展的知识源，利用本体图中概念间语义关系消除用户查询的语义偏差，从而实现查询扩展。本体知识库的查询语言是 SPARQL 语言，是在 RDF 查询语言如 RDQL 等的基础上发展而来，作为 W3C 推荐标准，SPARQL 受到 Jena 的全力支持，支持在本体中进行推理查询，当用户输入一个查询“《边城》是谁写的？”SPARQL 查询语句如下。

```
PREFIX
BookQuery:<http://www.semanticweb.org/lzz/ontologies/2016/3/untitled-ontology-22#>
SELECT  ?Author  ?Book  ?Publisher
WHERE  {BookQuery:BianCheng  BookQuery:isWritenedBy  ?Author}
```

本章提出一种基于语义的查询扩展方法，用户首先以自然语言形式输入检索问句，使用问句预处理技术对其进行中文分词、去停用词等操作抽取出关键核心概念词，使用本体内部概念与概念间关系进行查询扩展，生成合适的 SPARQL 查询语句进行检索，最终将结果返回用户。

4.2 基于语义的查询扩展算法设计

在基于语义的查询扩展中涉及到本体关系图和查询的概念，为了更加清晰的描述查询，本文给出语义关系图和查询的形式化定义，如定义 4.1 和定义 4.2 所示。

定义 4.1 语义关系图 SRG（Semantic Relation Graph，语义关系图）定义如下：

$SRG=(V,E)$

其中 V 是节点集合，$\forall v\in V$，v 代表一个概念，E 是本体关系图中一系列边的集合，$\forall e\in E$，$e=(v_i,v_j,rw)$ 表示 v_i 和 v_j 之间的一个关系，其权重为 rw。

定义 4.2 查询 查询（Query）由一组概念的集合组成，定义如下：

$Q=(A_1,A_2,\cdots,A_m)$

其中 A_i 代表与第 i^{th} 次查询条件对应的概念。

基于以上定义，本文给出查询扩展的形式化描述如下：

已知查询 Q，实现基于语义的查询扩展，称为 *SemanticExpansion*(Q)，查询扩展主要步骤包括查询关键词映射、构造查询子树、生成扩展关键词集，如图 4.1 所示。

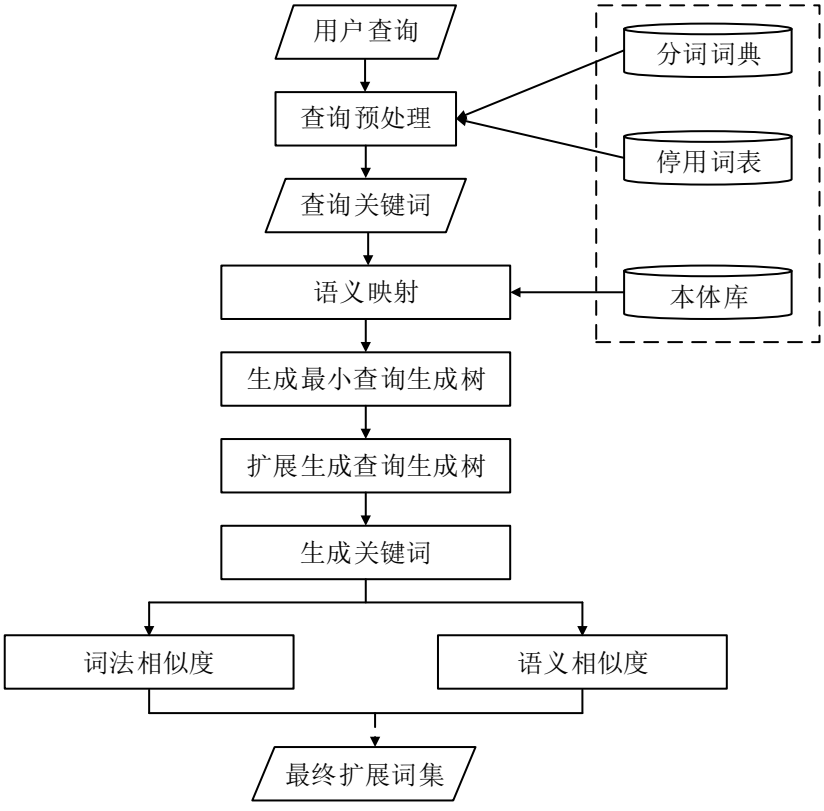

图 4.1 基于语义的查询扩展方法

4.2.1 查询关键词映射

查询关键词映射即将用户查询的关键词映射到本体关系图中，得到分散在本体图结构中的各个概念节点，将用户查询转化成图模式，其表现形式为节点和边的集合。由于首先用户通常以自然语言形式提交查询 Q，因此首先需要对已知查询 Q 利用问句预处理提取其关键词。此处主要使用 2.2.2 小节中所提文本表示、中文分词技术，将用户查询切分成一系列词汇集合，之后使用停用词表删除无意义的词汇及符号，保留对句子意义有用的实词，得到查询关键词集合 $Q=(q_1,q_2,\cdots,q_n)$。另外，为了获取最高概率匹配，本文使用哈工大同义词词林对查询关键词集合 Q 进行初步同义词扩展，扩展后得到集合 $Q_E=\{ q_{11}, q_{12}, \ldots, q_{1n}; q_{21}, q_{22}, \ldots, q_{2n}; \ldots; q_{m1}, q_{m2}, \ldots, q_{mn}\}$。

之后将查询关键词集合 Q_E 中的概念映射到本体中生成概念树，即将 Q_E 中的某个关键词 $q_{ij}(1\leqslant i\leqslant n, 1\leqslant j\leqslant m)$与 SRG 中节点集合 V 相对应的 $v_i(1\leqslant i\leqslant N)$进行匹配。

假如在 SRG 节点集合 V 中没有相对应的概念则不进行关键词 q_i 到本体树中概念的映射，若有则完成匹配并生成集合 Q_{match}。本文使用概念占有率的定义完成概念词汇映射操作，概念占有率表示了概念 c 在检索关键词中所占的比重，如定义 4.3 所示。

定义 4.3 概念占有率 O(c)　设概念词 q_i，令 α 表示为 SRG 节点集合 V 中的概念词汇数量，β 为使用同义词词林扩展后的概念词集合 Q_E 中的元素个数，使用集合思想计算 O(c)如公式（4.1）所示。

$$O(c_i)=\frac{\beta}{\alpha}=\frac{\left|C_{c_i}\cap Q_{match}\right|}{C_{c_i}} \tag{4.1}$$

在从用户查询关键词到本体概念的映射过程中，用户的查询意图可以由该公式清晰直观地反映。基于用户认知的角度，当 O(c)值越大，则概念 c 在检索关键词中所占比重也大，说明用户有极大可能性检索该概念或其子概念下的数据，因此在一定程度上 O(c)能反映用户的检索意图。

通过以上操作，可以得到概念节点在本体中的映射，各个命中概念节点分散在本体图中，删除映射中与用户查询词无关的节点和边，将用户查询 Q 可进一步表示为节点和关系的集合。例如，如图 4.2 所示的领域本体，假设用户问句经问句处理后所得关键词序列为 Q={C3,C5,C10,C12}，经概念占有率计算映射至本体中如图中阴影部分所示，Q 可进一步描述为 $Q=\{C, E\}$，其中 C 指匹配到的概念节点集合，E 指节点间相连的边关系集合。

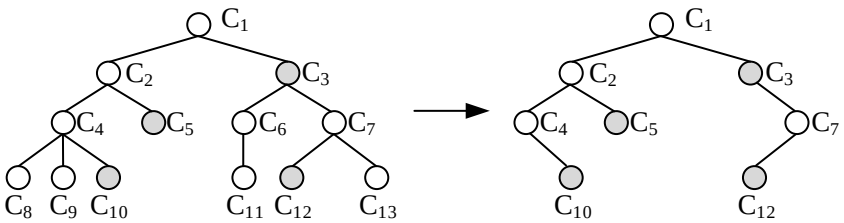

图 4.2　查询关键词映射

4.2.2 构造查询生成树

受多义词及同义词的影响，SRG 中不同概念可能表达同一语义含义。用户的查询习惯使得其查询关键词一般都指向特定领域，但在知识描述上会出现相同或相似词汇，因此需要构造一棵最能代表用户查询要求的查询生成树用于表示用户的检索意图。SRG 中概念间的语义关系的紧密程度可由节点间关系权重描述，具有更小相关权重的概念对具有更紧密的语义关系。因此，构造查询生成树问题可以转化为在 SRG 中找到一个与 Q 最相关的查询扩展子图，该子图覆盖 Q 中的所有查询词，其中不仅所有查询词都可访问，且路径的权重值之和最小，查询生成树满足定义 4.4。

定义 4.4 查询生成树（Query Spanning Tree） 给定语义关系图 SRG 和查询 Q，查询生成树 QST 满足以下条件：

①QST 包含 Q 中的每个节点；

②QST 不包含环；

③QST 是 SRG 的连接子图。

查询生成树 $QST=(V',E')$ 的权重值记为 $W(T)=\sum_{i-1}^{n} rw(e_i)$，$n$ 为 QST 中边的个数，$rw(e_i)$ 是边 e_i 的关系权重。

定义 4.5 最小查询生成树（Minimum Query Spanning Tree） 已知语义关系图 SRG 和查询 Q，最小查询生成树 MQST 可以描述为一个查询生成树，同时其满足以下条件：

$$W(MQST)=\min\{W(T)\,|\,T\in TS\}$$

其中，$TS=\{T_1,T_2,\cdots,T_n\}$ 是满足 SRG 和 Q 的所有查询生成树的集合。

根据定义可知，最小查询生成树 MQST 是 QST 的子集，可以看作是具有最小权重的查询生成树。最小生成树与最小查询生成树不同，前者的节点是语义关系图中的所有概念节点，后者的节点是用户查询中的所有概念节点；其次，前者有 $n-1$ 条边（n 为节点数），后者的边数则不确定。本文改进最小生成树算法，基于映射本体后的图形式的查询 $Q=\{C,V\}$和语义关系图 SRG，使用矩阵 M 存储查询中每个节点之间的最短路径和相关值，使用邻接表 A 保存查询中每个节点的连接性，构建最小查询生成树。

改进最小生成树算法核心思想是不断地从 SRG 中选择一条新路径并将其添加到 MQST，使得 MQST 保持最小权重值，即与 Q 最相关，算法伪代码如表 4.1 所示。首先从 Q 中随机选择一个节点 v_0 作为起始节点，并初始化 MQST 中的节点集 V 和边集 E。然后，确定 Q 中的每一对概念(u,v)的连通性，若连通则将节点存储在 A 中，并保存(u,v)之间的最短路径及对应权重。初始化权重数组 Weight 和最小堆 H，进行迭代确保数组 Weight 中的连通节点的 v 值是其到 MQST 的最小权重值。不断将路径中的节点和边添加到集合 V 和 E，直到 Q 中的节点都添加到 MQST 中，最终输出 MQST。

表 4.1　最小查询生成树构建算法

算法 4.1　最小查询生成树构建算法

输入：Q　//查询
　　RG　//语义关系图
输出：MQST　//最小查询生成树

```
1.  v0 ← Random(Q)
2.  H ← Heap(Q)
3.  V ← v0
4.  E ← Ø
5.  A[ ] ← 0
6.    FOR all (u, v) such that (u, v) ∈ Q DO
7.      IF Reachable(u, v) THEN
8.        A[u] ← A[u] U v
9.        M[u, v].p ← ShortPath(RG, u, v)
10.       M[u, v].w ← Weight(M[u, v].p)
11.     ELSE
12.       M[u, v].p ← NULL
13.       M[u, v].w ← ∞
14.     END IF
15.   END FOR
16.  FOR EACH i ∈ Q DO
17.    IF i = v0 THEN
18.      w[i] ← 0
19.    ELSE
20.      w[i] ← ∞
21.    END IF
22.  END FOR
23.  REPEAT
24.    hm ← popMinElement(H)
25.    FOR EACH u ∈ A[hm] DO
26.      IF M[hm, u].w < w[u] THEN
27.        w[u] ← M[hm, u].w
28.        adjust(H);
29.        path(u1, u2, ⋯, um) ← M[hm, u].p
30.        FOR EACH (ui, uj) ∈ path DO
31.          V ← V U ui
32.          E ← E U (ui, uj, rw(ui, uj))
33.        END FOR
34.      END IF
35.  UNTIL H = NULL
36.  RETURN MQST(V, E)
```

由 MQST 和 QST 的关系可知，MQST 是 QST 的子集，QST 中还包含满足 SRG 和 Q 的所有集合，因此需要对最小查询生成树 MQST 进一步扩展后生成查询生成树 QST。在查询 Q 的本体映射子图中，根节点到所有查询关键词节点的路径至少一条，为保证本体关系图中每个边所赋权重的有效性，本文设置权重阈值 δ，同时定义每层最大边个数 n，令有效路径为 VP，则 VP 满足：

①k≤n

②$\sum_{i=1}^{k} rw(e_i) \le \delta$

则 MQST 的有效扩展即为 QST，计算如公式（4.2）所示，式中中VPS_{v_i}代表 v_i 的有效路径集合。

$$QST = MQST \cup VPS_{v_1} \cup VPS_{v_2} \cup \cdots \cup VPS_{v_m} \quad (4.2)$$

4.2.3 生成关键扩展词集

用户查询具有盲目性，输入的查询语句有可能存在与用户查询意图语义相关但意义不明确的信息，依据用户查询的分词结果构建查询生成树时，若单纯以查询生成树作为检索依据则可能出现错误结果。因此，在将用户查询意图形式化后，需要对 QST 中关键概念进行扩展以获取更多有价值信息。扩展时首先对查询生成树中的概念进行相似度计算，依据相似度计算结果进行排序，依据 top-k 策略对排序结果进行筛选，进而将扩展结果和用户提交的查询词 A_i 合并成新的查询关键词，生成最终语义查询扩展集合 *SemanticExpansion*(*Q*)。在概念相似度计算中，本文从词法和语义两个层面考虑，分别阐述如下。

（1）词法相似度

传统词法层面的概念相似度有基于 WordNet 的相似性、基于编辑距离的相似性等，常用方法有 Path 度量、Resnik 度量、Levenshtein 相似度等。Path 度量基于 Wordnet 图，使用 WordNet 中两个概念之间的路径长度表示它们之间的相似性。Resnik 度量使用两个概念的最小公共子串计算概念间共享信息。

本文使用 Levenshtein 编辑距离计算词汇相似度，Levenshtein 编辑距离即将一个字符串转换为另一个字符串所需的单个字符的最小操作数。Levenshtein 相似度 *LevSim*(A_1,A_2)计算如公式（4.3）所示。

$$LevSim(A_1, A_2) = 1 - \frac{LevenshteinDistance(A_1, A_2)}{\max Length(A_1, A_2)} \quad (4.3)$$

（2）语义相似度

本文简化 3.2.3 小节所提语义相似度计算方法计算概念间语义相似度，使用本体中两个概念间语义距离度量其相似度。考虑概念节点深度和概念节点词序的影响，对已知本体间关系进行权重分配，计算两个概念节点 *m*, *n* 间权重 W(*m*, *n*)，在此基础上，用户查询词 A_1、A_2 的语义相似度计算计算公式如（4.4）所示。其中 W(A_1, root)代表概念节点 A_1 距本体根节点的语义距离，W(A_2, root)代表概念节点 A_2 距本体根节点的语义距离，W(com, root)表示概念节点 A_1 和 A_2 的最近公共节点距本体根节点的距离。

$$SemSim(A_1, A_2) = \frac{1}{W(A_1, root) + W(A_2, root) - 2 \times W(com, root) + 1} \quad (4.4)$$

最终本文综合词法相似度和语义相似度，计算概念相似度，如公式（4.5）所示。

$$Sim(A_1, A_2) = \alpha \times LevSim + \beta \times SemSim \quad (4.5)$$

其中，α 和 β 的值表示词法相似度和语义相似度的权重，满足 $\alpha + \beta = 1$，此处依据经验值取 $\alpha = 0.4$ 和 $\beta = 0.6$。

4.2.4 算法设计

基于以上描述，本文设计基于语义的查询扩展算法，算法输入用户查询 Q、语义关系图 SRG，最终输出查询的关键扩展词集 Q_{SE}。算法描述如表 4.2 所示。

表 4.2　查询扩展算法

算法 4.2　查询扩展算法
输入：Q　//用户查询 O　//本体 SQG　//语义关系图
输出：Q_{SE}　//查询扩展词集
1.　$Q \leftarrow Q=\{q_1,q_2,\cdots,q_n\}$　//对用户查询 Q 进行分词
2.　C ← 扩展 O 中的概念
3.　计算概念占有率 O(c)
4.　生成映射本体子树 O_Q
5.　构造最小查询生成树 MQST
6.　QST ← MQST$\cup VPS_{v1} \cup VPS_{v2} \cup \cdots \cup VPS_{vn}$
7.　FOR EACH $q_i \in O_Q$ DO
8.　　FOR EACH $v_j \in$QST DO
9.　　$Sim(q_i,v_j)$
10.　　IF $Sim(q_i,v_j) > Sim(q_{i+1},v_{j+1})$
11.　　DELETE q_{i+1},v_{j+1}
12.　　$Q_{SE} \leftarrow q_i,v_j$
13.　　END FOR
14.　END FOR

4.3 实验及分析

4.3.1 实验环境及数据

为了验证本文算法的性能，本文设计了语义检索系统进行查询扩展实验，实验环境如表 4.3 所示。

表 4.3　实验环境

项目	平台及参数
语言	Java
操作系统	Windows 8.1 专业版
开发平台	My Eclipse
数据库	MySQL
本体编辑工具	Protégé4.3.0、Jena 2.4

目前国际上常见的检索性能验证的测试集多为来自 TREC（Text Retrieval Conference，文本检索会议）的公共测试集，如 TREC、CASM、ISI 等。该类测试集包含领域较广，覆盖的知识面较多，依据此类测试集无法建立通用本体，并不适用于

本文的查询扩展方法。因此，本文以图书领域智能问答系统为背景，利用网络爬虫工具从陕西省图书馆网站内爬取 500 篇相关文档作为测试集。

实验中采用的领域本体是由 Protégé 构建的图书领域本体，该本体由 OWL 语言描述，并使用 Jena 内部自带的通用规则推理机 Generic Rule Reasoner 对本体文件进行基于一般用途的推理，确保本体概念及关系的正确性与一致性。该本体图共有实体 142 个，关系 506 对，包含图书与图书、图书与出版社、图书与作者等之间的关系对。部分本体结构如图 4.3 所示，该本体片段包含展示了图书、作者、出版社、出版时间、读者、借阅时间等实体及其之间的关系其中带原点的方框表示类，带菱形的方框表示实体，线条则代表类之间及实体之间的关系。

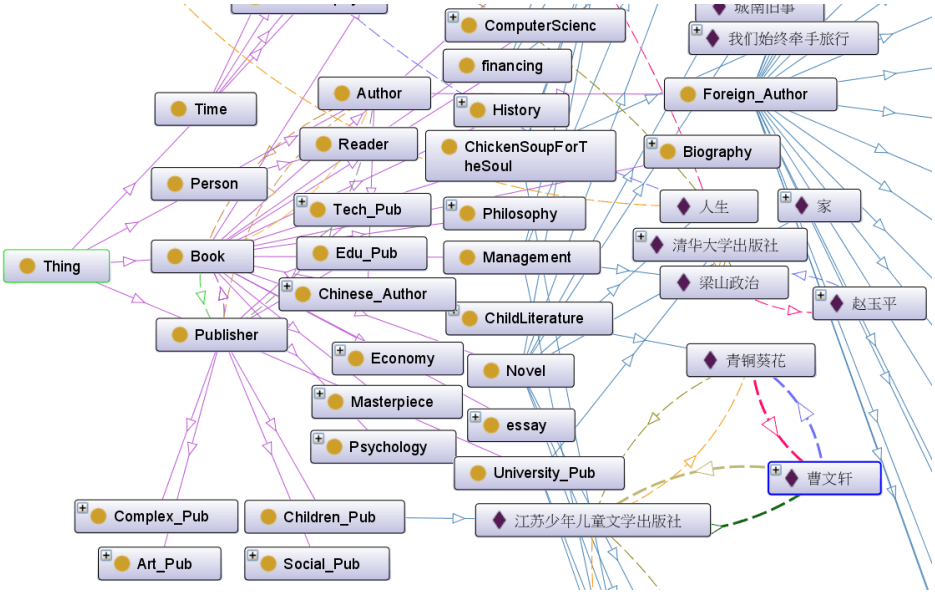

图 4.3　图书领域本体片段

4.3.2 实验设计与分析

实验时首先进行用户输入查询问句的预处理，主要目的是抽取初始查询的关键概念，主要过程为：

①使用 Jena 工具导出图书领域本体的概念及概念间关系至文本文件中，将其作为本体查询词典写入 NLPIR 分词系统的用户词典中；

②使用 NLPIR 分词系统对用户的初始自然语言查询进行分词、去停用词操作，得到概念词集合 S；

③使用哈工大同义词词林对集合 S 进行扩展得到扩展集合 S’；

④使用 4.2.1 小节所提概念占有率的定义将用户查询关键词映射到本体概念中。

在完成上述步骤之后，使用改进最小生成树算法构造查询子树，最后通过概念语义相似度的计算生成关键词扩展子集，利用查询扩展子集对相关文档进行检索。

查询扩展的效果影响着检索系统的性能，检索系统的评价指标通常有查全率、查准率、F－度量值、E－度量值、MAP（Mean Average Precision，平均精度值）、NDCG（Normalized Discounted Cumulative Gain，归一化折损累计增益）等，本文选用最为

常见的查全率、查准率和 F - 度量值进行性能评价。

为了对本文所提方法进行全面评估，实验中选取无查询扩展方法、关键词查询扩展方法与本文所提方法进行比较。比较方法描述如下：

无查询扩展方法：获得用户查询后，对其分词后直接检索相关文档集合。

关键词扩展方法：对用户查询进行分词后获得初始关键词，构造关键词的同义词序列，之后检索相关文档。

针对以上方法分别进行 30 次查询操作，统计查询结果的准确率、召回率、F - 度量值如表 4.4 所示。F - 度量值可以综合反映系统的检索性能，因此本文随机选取 10 次查询计算 F - 度量值的平均值，比较结果如图 4.4 所示。可以看出，本文所提基于本体的查询扩展性能最优，关键词查询扩展方法次之，无查询扩展方法性能最差。

表 4.4　查询性能比较

查询扩展方法	准确率	召回率	F - 度量值
无查询扩展方法	0.326	0.461	0.382
关键词查询扩展方法	0.465	0.535	0.497
本文所提方法	0.653	0.697	0.675

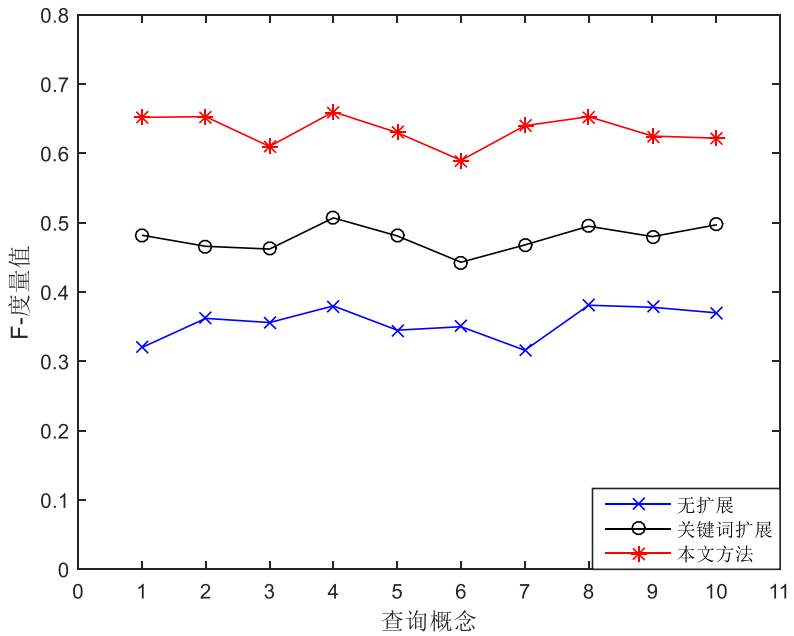

图 4.4　F - 度量值对比图

由表 4.2 可知，在查询性能上，关键词查询扩展方法和本文所提方法都优于无查询扩展方法。无查询扩展方法只单纯将用户问句进行问句预处理操作就直接进行查询，经分词得到的关键词并不能清晰完整的表明用户的查询意图，所以该方法性能表现较差。关键词查询扩展以查询词为中心进行机械式扩展，并没有考虑查询概念词间的语义关系，因此只能检索出部分相关文档，虽然其召回率表现较优，能在检索中更

多的查询到相关文档，但是由于用户查询的句子结构复杂且存在一词多义现象，导致其准确率较低。本文所提基于本体的查询扩展方法与关键词查询扩展方法相比，不仅保证了查询的召回率，也保证了查询的准确率。其原因是借助于本体结构进行了概念词的语义扩展，在扩展之后又依据相似度排序筛选，保证了扩展后的关键词的精确性。

在查询扩展中，概念词扩展集合的规模也会影响到查询的效果。若扩展词过少，则不能充分表达用户查询意图，查询的召回率必然降低，扩展词过多又会造成查询精度降低的问题，准确率随之降低。为了研究概念词扩展集合规模对查询结果的影响，本文对扩展词数量进行了实验对比，扩展词个数由 5 开始，以 5 个为间隔递增至 35，计算对应的 F - 度量值，结果如图 4.5 所示。由图可以看出，当查询关键词增多时，由于不断引入其他噪声到初始查询中，导致了 F - 度量值不断下降，产生了“查询漂移”现象，当扩展词个数在 10~15 个左右时，查询效果较好。

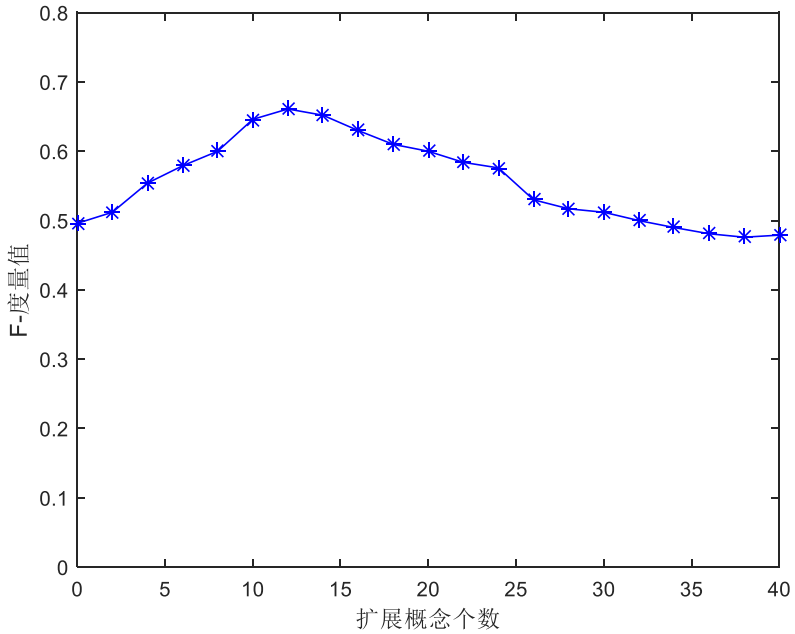

图 4.5　扩展规模对 F - 度量值的影响

4.4 本章小结

本章阐述现有查询扩展方法的优缺点，在对比分析的基础上提出基于语义的问句查询扩展方法。设计基于语义的查询扩展方法，首先对用户问句进行预处理，将处理后生成的关键词序列映射到本体中，之后使用改进的最小生成树算法生成最小查询扩展子树，根据最小查询扩展子树与查询生成树的关系构建查询生成树，最后基于相似度计算对查询扩展子集进行二次筛选，生成最终关键扩展词集。实验结果表明，本章提出的方法与无扩展方法、基于关键词扩展的方法相比性能更好，更清晰的表达用户查询意图，这也为下文原型系统的设计奠定基础。

第 5 章　智能问答原型系统

在第二章智能问答系统理论及关键技术研究的基础上，本文使用第三、四章所提的句子语义相似度计算方法及基于语义的查询扩展方法，搭建了一个基于语义理解的智能问答原型系统。系统接受用户以自然语言形式的问答，返回精确答案，同时设计句子相似度模块和查询扩展模块对系统实现进行展示。

5.1 系统设计方案

5.1.1 实验环境配置

基于以上几个章节的讨论，本文选择在 Windows 操作系统下完成基于语义理解的智能问答系统的原型系统设计与实现，系统设计的基本平台环境如表 5.1 所示。

表 5.1　平台环境一览表

硬件环境		软件环境	
		操作系统	Windows 8.1 专业版
		开发平台	My Eclipse
处理器	Intel Core i5-4570	开发语言	Java、OWL
RAM	4.00GB	数据库	My SQL
ROM	500G	本体编辑器	Protégé4.3.0
		开发工具包	Jena 2.4
		分词工具	NLPIR

此处主要阐述与本体构建和推理相关的 Protégé和 Jena。

Protégé 是一个免费的开源平台[66]，基于描述逻辑提供一系列工具支持领域本体构建，使得概念和属性的描述成为可能。Protégé支持采用各种表示格式的本体创建、本体合并、本体间公理转移、多个实体的重命名及其他操作，可视化工具允许本体关系的交互导航，其提供的自带推理机可检查本体中关于概念的描述和定义是否互相一致，同时也可以识别概念间的子类关系。在所构建本体具有多个父节点情况时，Protégé提供的推理机可以帮助检查本体的层次结构。Protégé 官方提供离线软件编辑本体和在线编辑本体的两种操作模式，本文采用离线编辑模式，使用 Protégé4.3.0 版本在本地客户端进行领域本体的构建。

Jena 是由惠普开发的用以构建语义 Web 应用的开源 Java 平台[67]，属于当前热门的本体推理系统之一，提供针对 RDF、RDFS、OWL 的编程环境，包括对 RDF 文件进行处理的 RDF API，以及对 OWL、RDF、RDFS 文件进行解析的解析器。Jena 提供基于规则的 OWL 推理引擎，分为 RETE 引擎和 tabled datalog 引擎两个部分，不同的引擎需要定义不同的规则以确定其行为，并且提供了前后向链执行模型以及混合执

行模型。Jena 中 JDBC 驱动可以支持本体的持久化存储，同时支持使用 SRARQL 语言对本体进行查询语言。在 Jena 中需要针对本体概念和属性之间的差异特征定义不同推理规则，用于检查多个概念及属性之间的联系以及属性的互逆性、传递性等。本文使用 Jena 2.4 版本进行本体的持久化存储和基于一般规则的推理。

5.1.2 系统架构

本文构建图书领域的领域本体作为知识库，设计基于 FAQ 库的问答和基于本体的问答相结合的问答策略，实现基于语义理解的智能问答系统。系统架构图如图 5.1 所示，该架构自下而上由基础设施层、数据获取层、知识资源层、核心技术层及应用层构成。

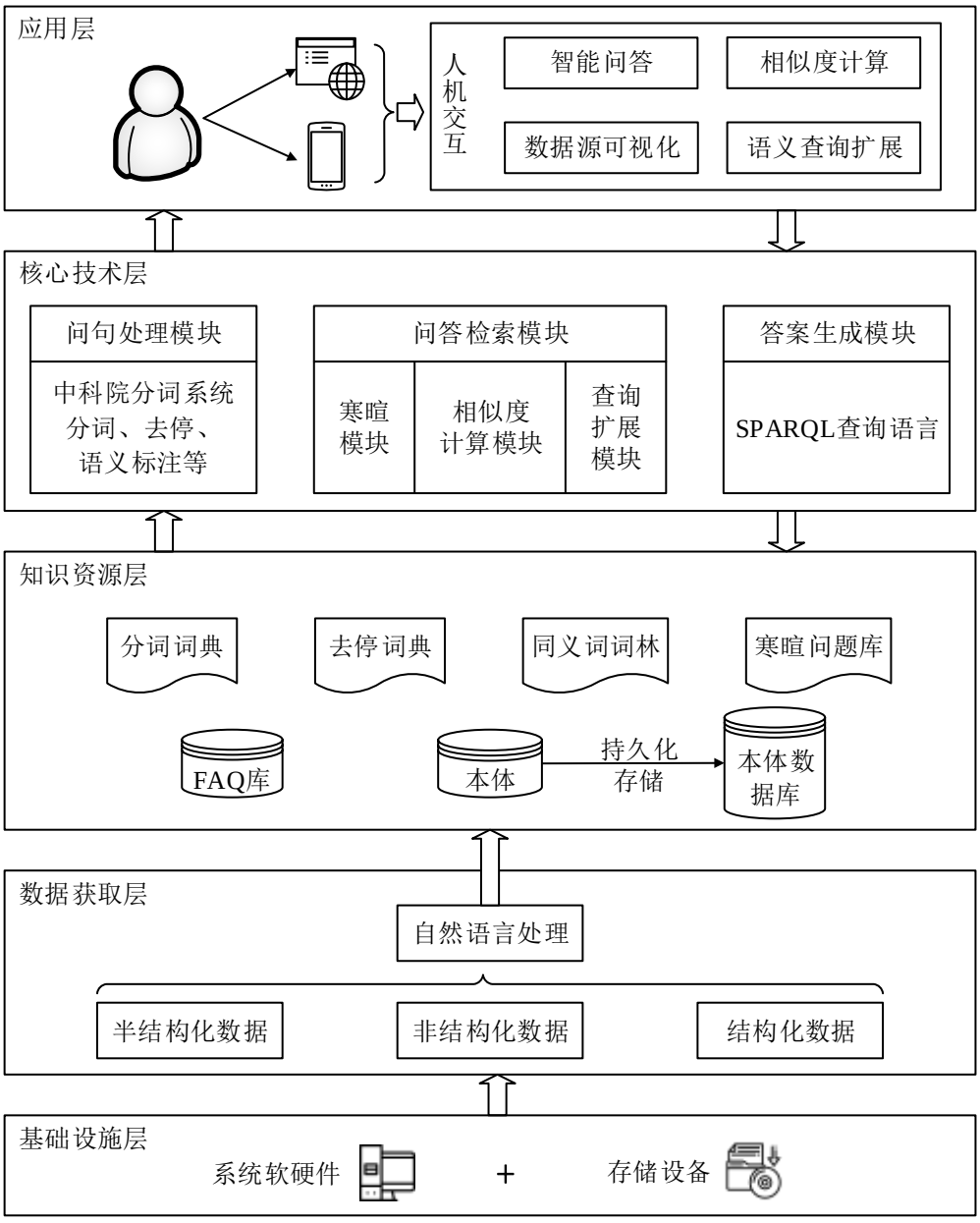

图 5.1　基于语义理解的智能问答系统架构图

基础设施层是基于语义理解的智能问答系统可靠运行的底层设备，为问答系统提

供系统硬件、软件及存储设别等一系列基础设施。

数据获取层为知识资源层提供数据来源和知识储备，采用爬虫工具在中国国家图书馆、陕西省图书馆等网页上提取读者留言数据以及一些包含图书信息的网页，对其进行页面解析、文本提取等操作后，将网页上提取的非结构化、半结构化数据转化为结构数据。

知识资源层提供整个系统的数据支撑，其最终目的是为上层各应用模块提供数据支持和保证，主要包含分词词典、去停词典、同义词词林、FAQ 库、寒暄问题库及领域本体。其中分词词典、去停词典主要配合问句预处理模块使用，完成用户问句的分词、去停和消歧；同义词词林用于查询关键词的简单扩展；FAQ 库中存储用户常用问题库，为基于 FAQ 库问答模块提供支持；寒暄问题库对用户的寒暄问题进行友好回复；图书领域本体为整个系统提供语义查询及推理能力，同时对本体进行持久化存储操作实现系统的高可用性。

核心技术层是在以知识资源层所提供的数据的基础上进行智能问答核心技术模块的实现，包含问句处理、智能问答、答案生成三个部分。问句处理调用中科院 NLPIR 分词系统对问句进行分词、去停处理；智能问答模块针对用户问句设计 FAQ 库和本体库结合的问答策略进行答案的查询和检索，寒暄问答模块用于系统和用户之间的友好交互，相似度计算模块目的是用户问句和 FAQ 库问句的匹配，查询扩展模块则利用领域本体将用户查询意图进行扩展；答案生成模块使用 SPARQL 语句对扩展后的关键词库进行语义查询并将答案提交给用户。

应用层设计友好界面提供智能问答、句子相似度计算、基于语义的查询扩展、数据源查看等可视化界面，用户可以便携访问系统，通过友好界面向后台服务器提交查询问句，接受并查看后台返回的答案信息。

5.2 系统核心技术及功能模块实现

5.2.1 数据准备

本文构建图书领域的智能问答原型系统，知识资源层中分词词典、去停词典及同义词词林都是开放资源，系统设计中需要手工构建领域本体、FAQ 库和寒暄问题库，构建过程分别阐述如下。

（1）图书领域本体构建

目前基于图书领域的智能问答研究还不够深入，没有标准化的统一研究方法，还未形成可用的、工人的标准数据集。本文结合本体描述语言 OWL，采用 Protégé 工具构建图书领域本体。本文依据七步法进行图书领域本体构建，主要步骤描述如下：

①选择词汇和概念

本文参考文献[68]~[71]，并参考《中国图书馆图书分类法》、《中国分类主题词表》及国际互联网关于图书的网站，如中国国家图书馆、中国科学院图书馆、陕西省图书

馆等网站，结合本文问答系统架构的设计，搜集大量图书领域基本词汇，包括图书类别及名称、出版社类别及名称、作者编者及名称等。

②定义类名、属性

对收集的图书领域词汇进行去停、抽象、删重等格式化操作，定义术语分类。在图书领域本体中，定义四个一级类图书（Book）、出版社（Publisher）、人员（Person）、时间（Time），之后在一级类下划分二级子类。依据《中国图书馆分类法》在图书类下定义人物传记（Biography）、儿童文学（Literature for Children）、历史（Histories）、哲学（Philosophy）、小说（Fiction）等子类；人员类包含作者（Author）和读者（Reader）两个子类，其中作者又分为中国籍作者（Chinese_Author）和外国籍作者（Foregin_Author）；出版社类中定义七个子类，时间类中定义两个子类，此处不再赘述。整个图书本体以 Thing 为大类被划分成了四层结构，分别为 Book、Person、Publisher、Time 等，此处给出 Thing 和 Book 之间的 *is-a* 关系类图如图 5.2 所示。

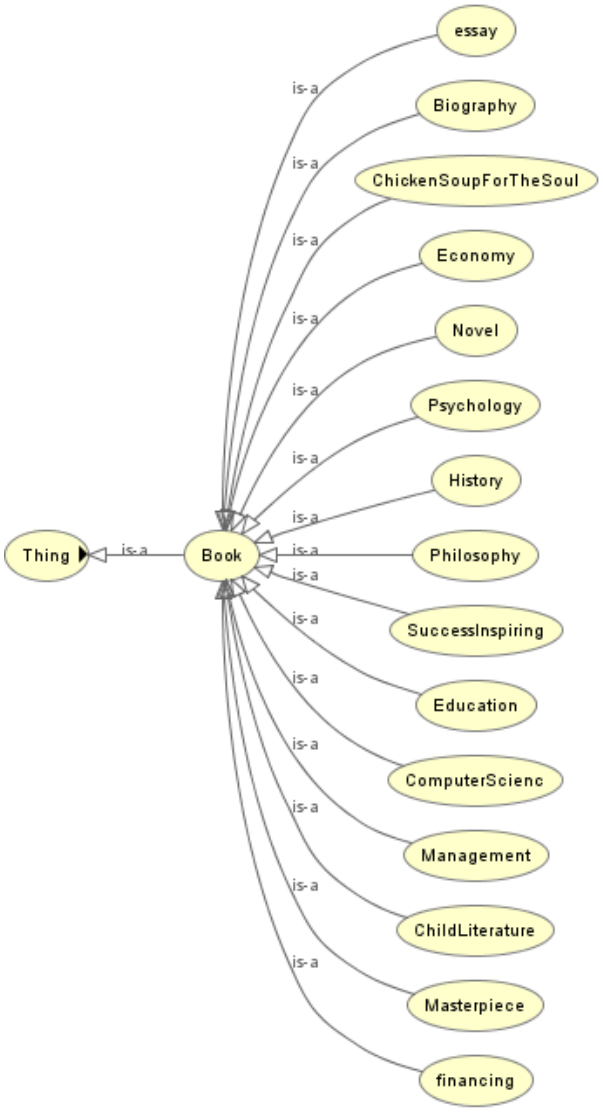

图 5.2　类间 *is-a* 关系图

在定义好图书领域类名之后需要定义各类的数据属性和对象属性。在图书领域本体中针对四个大类间互相存在的关系定义了 20 多个属性，包含图书与出版社、作者、读者、时间之间的互相关系，如出版（hasPublish）、撰写（hasWritten）、付稿费（payMoneyTo）、出版时间（isPublishedIn）、借阅时间（isBorrowedBy）、归还时间（isReturnedBy）、被出版（isPublishedBy）等，每个属性都分别定义其定义域和值域，如 hasPublish 属性的定义域是 Publisher，值域是 Book，如图 5.3 所示。

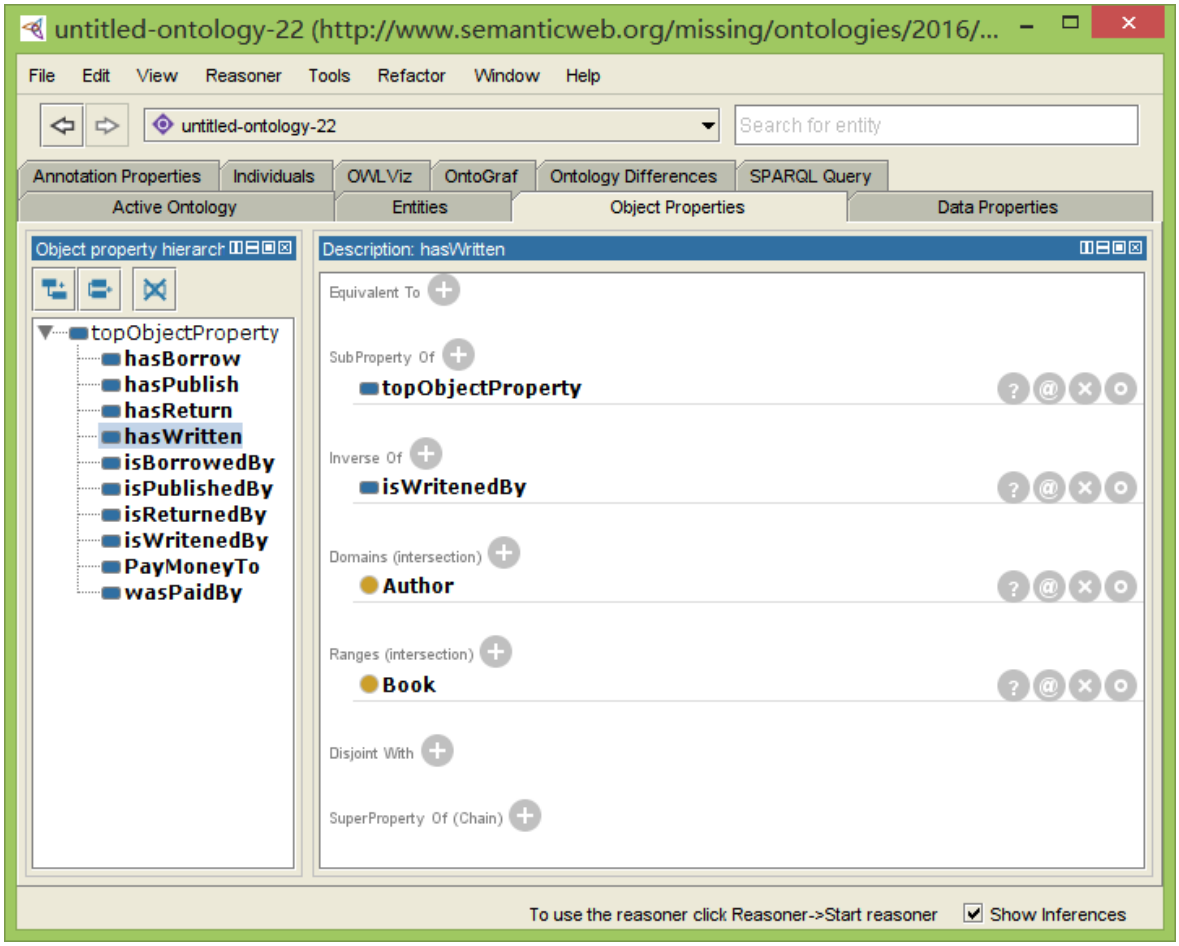

图 5.3　图书本体属性范例

③定义属性限制

属性的限制规则包含基数及关系特征，利用 OWL 属性限制对类进行描述十分必要，本文主要针对属性间关系特征进行定义。如图书本体中 Author 类的属性 hasWritten，定义其关系特征“inverse of ‘isWrittenBy’”，即意味着 hasWritten 与 isWrittenBy 为互逆属性，两个属性的值域和定义域正好相反，这样就限制了两个属性的定义域和值域，否则在使用推理机进行本体检查时会出错。

④添加实例

建立好本体库中的类和属性之后必须添加一定数目的实例以使用实际应用的需求。Protégé 提供了友好的图形化界面使得实例添加十分容易，同时也提供本体的图形化展示。如图 5.4 所示，是本文构建的图书领域本体片段。

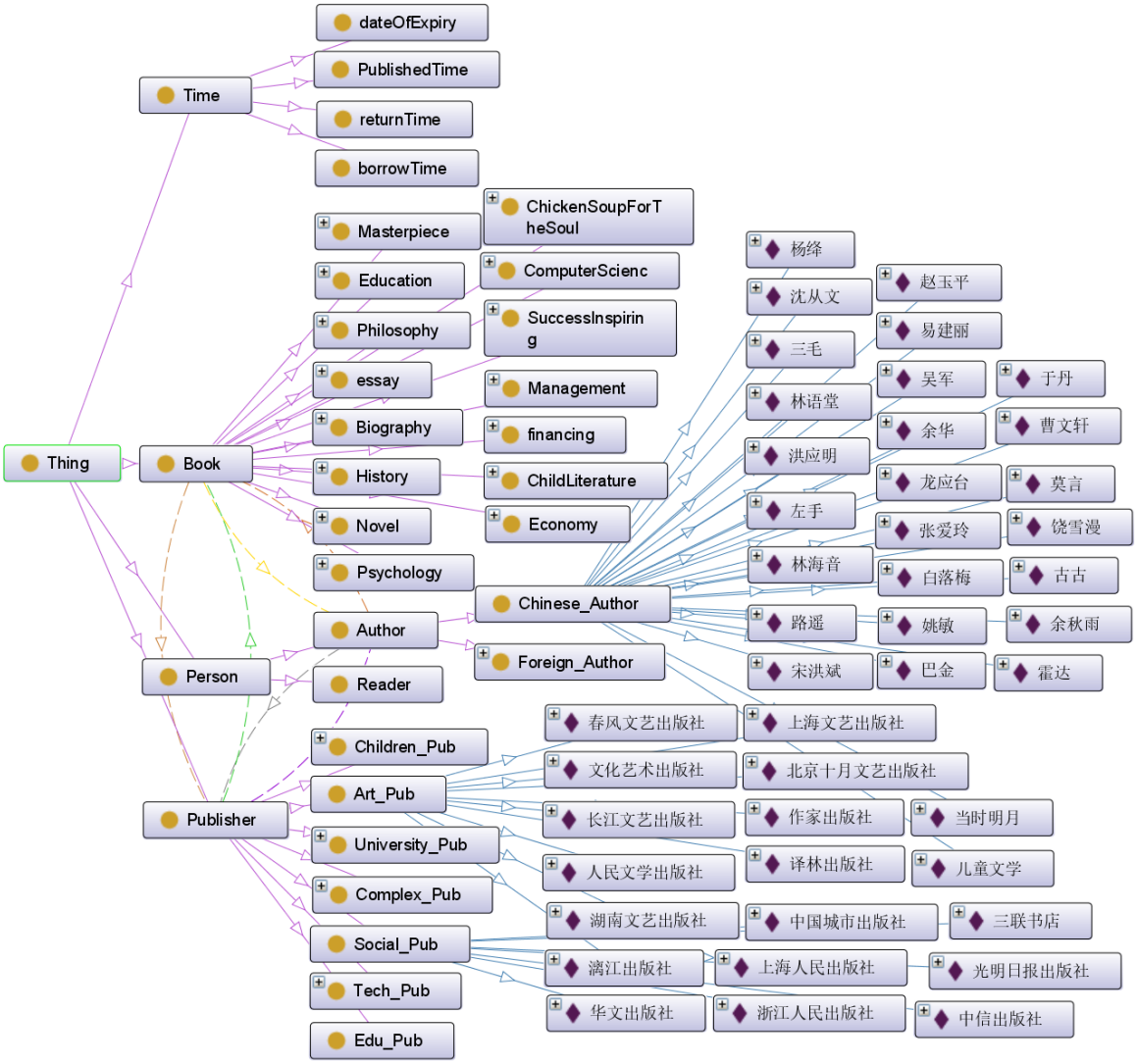

图 5.4　图书领域本体片段

（2）FAQ 库构建

在问答系统研究中问题库的质量可以直接影响问答系统的准确率和召回率，因此问题库的构建十分重要。本文主要针对图书领域进行讨论，因此其他一些已知资源库如哈尔滨工业大学的开放问题库等无法直接使用，因此本文通过以下四种来源收集问题构建 FAQ 库。

①目前有不少学者都从事特定领域的问答系统研究，如余正涛教授进行旅游领域问答系统研究、于忠清进行营养知识问答系统研究等。本文参照他们所构建的 FAQ 库，进行图书领域问题库的构建，如成熟的旅游领域问句“丽江有哪些景点？”、“云南有什么土特产？”等。

②从互联网上收集图书领域相关问句并进行合适变换，如陕西省图书馆读者留言咨询、各大高校图书馆读者留言等。

③哈尔滨工业大学信息检索研究中心所构造了开放域问题库共计 6266 句，本文参照其进行图书领域常用问题库的构建。

④参考 TREC 文本检索会议所提供的免费英文问句进行构建。

通过以上四个途径本文整理初始问句集，经相似度计算后筛选重复问句，并给出剩余问句的对应答案，最终形成图书领域问答系统的 FAQ 库。

本文构建图书领域 FAQ 库片段如表 5.2 所示。

表 5.2　图书领域 FAQ 库

Id	Question	Answer
…	…	…
7	文艺出版社都有哪些？	比较著名的文艺类出版社有《作家出版社》、《北京十月文艺出版社》、《人民文学出版社》、《文化艺术出版社》等。
8	请问杨绛的著作有哪些？	《称心如意》、《弄真成假》、《风絮》、《倒影集》、《洗澡》、《春泥集》、《将饮茶》、《干校六记》等
9	请问吴军都有什么书？	《数学之美》、《浪潮之巅》、《文明之光》
10	区块链的书都有哪些？	《区块链设计与应用》、《图说区块链》、《区块链核心算法解析》、《区块链与人工智能》、《区块链项目开发指南》等。
11	人民文学出版社出版了哪些巴金的书？	人民文学出版社出版的巴金的书有《家》、《回忆录选》、《一个家庭的戏剧》等。
12	《深入理解计算机系统》是哪个人写的？	《深入理解计算机系统》是美国的 Bryant. R. E. 等人编写的计算机经典书目。
13	路遥的《人生》是什么出版社出版的？	北京十月文艺出版社
…	…	…

（3）寒暄问题库构建

寒暄问题库的主要作用是给用户提供良好的操作体验，要求系统不仅能回答关于图书领域的专业知识，还能对日常寒暄语如“你好”、“谢谢”等进行回答。寒暄语通常十分简短，是依据谈话技巧的问答模式，如表 5.3 列出了本文设计的寒暄问题库。系统使用基于关键词匹配的方法进行寒暄语判断，将用户输入与寒暄问题库中的问句遍历匹配，如果匹配成功，则返回与之对应的回答，否则进入到 FAQ 库问答进行下一步处理。

表 5.3　寒暄问题库

ChatId	ChatQuestion	ChatAnswer
1	你好	你好，我是智能助理，请问有什么可以帮您的？
2	你好呀	你好，我是智能助理，可以帮您吗？
3	Hello	Hi，我是您的智能助理，请问有什么可以帮您的？
4	Hi	Hello，我是你的智能助理，请问有什么可以帮您的吗？
5	哈哈	哈哈，我超智能的，请问有什么可以帮您？
6	谢谢	不客气，感谢您的使用，竭诚为您服务。
7	在吗	您好，在的，有什么可以帮您的吗？
8	再见	再见，期待您的再次使用！
9	拜拜	感谢您的使用，期待再次为您服务！
10	亲	你好！欢迎提问！
…	…	…

5.2.2 问句处理

在问答系统中用户以自然语言方式提出问题，首先经由寒暄处理模块判断用户问句是否为寒暄句，若是给出相应寒暄回答，若不是则对问句进入分词处理。

问句预处理的主要步骤如下：

Step 1：获取用户问句。

Step 2：与寒暄问题库对比，若不是寒暄语，则进入Step 3；若是寒暄语，则依据寒暄问题库给出寒暄回答，等待用户下一轮输入，若有输入则进入Step 1，若无输入则退出。

Step 3：加载分词词典和去停词典，调用中科院分词系统NLPIR生成分词结果。

Step 4：对每一个分词结果，依次与停用词典中的词汇进行比对，判断当前关键词是否为停用词。

Step 5：若该关键词是停用词，则从分词结果中删除。否则继续Step 3，直至将所有分词结果遍历结束。

Step 6：输出最终分词结果并结束。

如用户问句“《呐喊》是哪个作者写的？”，分词结果为“《/呐喊/》/是/哪个/作者/写/的/？”，去停结果为“呐喊/作者”。

5. 2. 3 智能问答

（1）问答策略的设计

基于语义理解的问答系统是一种多策略问答系统，本文主要研究图书领域的问答系统的实现，对用户已自然语言理解形式提出的问句进行分析，采用不同的问答策略返回用户答案。本文设计问答策略流程如图 5.5 所示。

①FAQ 库存储常识性问题，当用户输入查询问句后，直接将用户问句与 FAQ 库中的问句进行语义相似度匹配，如果匹配成功，将对应的答案直接返回给用户。

②如果匹配失败，即 FAQ 库中不存在与用户问句相似的常识性问题，则对用户问句进行分词、去停等问句预处理操作，提取出查询关键词，进行基于本体的查询扩展，之后生成 SPARQL 查询句，在图书领域本体中进行语义查询，进而返回答案结果，同时将返回结果的过程中将问题存入到 FAQ 库中，方便下次查询。

（2）句子相似度计算的设计与实现

使用第 3 章提出的多特征融合的句子语义相似度进行句子相似度的计算。该模块的输入是两个问句，分别计算两个问句的结构相似度和语义相似度，最终进行融合加权计算句子的综合相似度，输出则是两个句子的相似度计算结果。

（3）基于语义的查询扩展的设计与实现

使用第 4 章提出的基于语义的查询扩展方案进行用户查询词的扩展。该模块的输入是用户自然语言形式的问句，经问句预处理、查询关键词映射、构造查询子树等步骤，最终输出查询关键词的扩展词集。

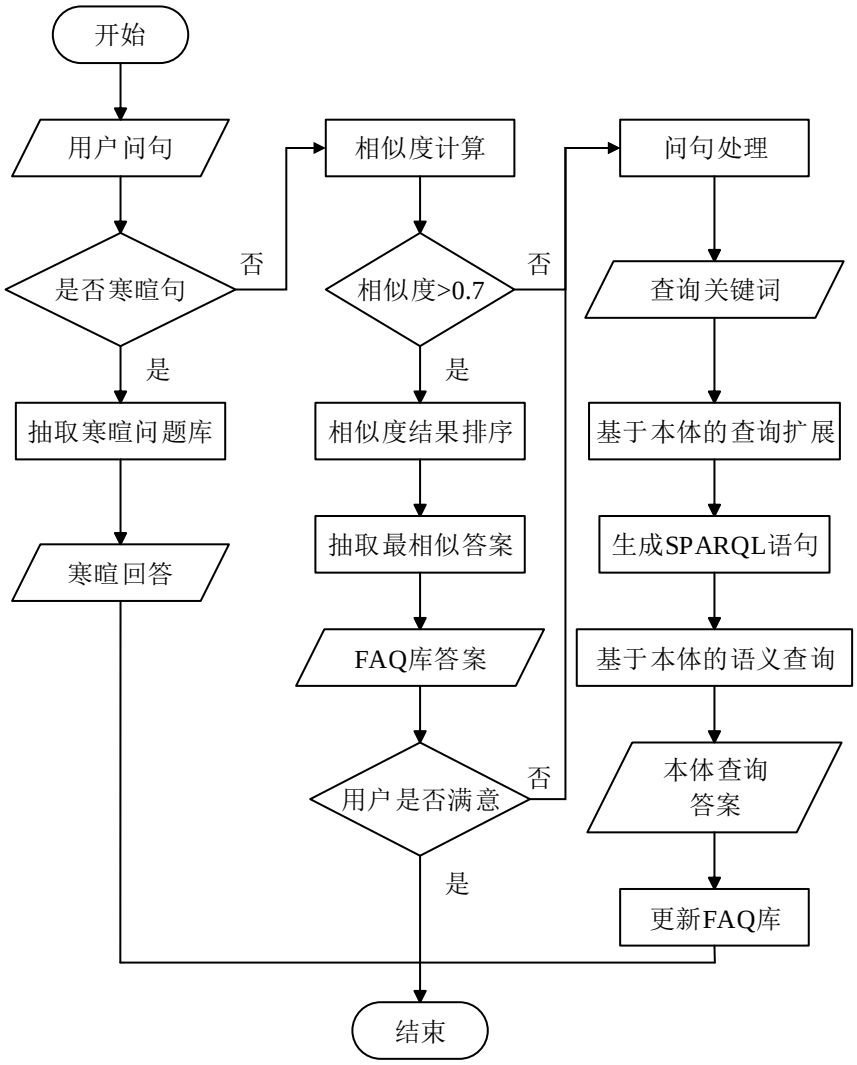

图 5.5　基于语义理解的智能问答系统问答策略流程图

5.2.4 答案生成

在 FAQ 查询中，计算用户问句与 FAQ 库问句的相似度，在 FAQ 库中达到相似度阈值的常用问题库个数必然不止一个，因此需要对所有相似的常用问题进行重排序，选择最相似的问句将答案返回给用户。在语义查询中，经自然语言处理技术对用户查询句进行处理，获取用户查询意图之后，需要针对用户查询意图生成对应知识库的 SPARQL 查询语言。使用查询扩展技术解决由于用户在问答时关键词使用不当造成的问答检索结果偏差，根据语法构建查询三元组，生成对应的 SPARQL 查询语言在领域本体中进行语义查询，最终将查询结果返回给用户。

5.3 原型系统测试

综合以上讨论，本文利用上述核心技术，在 My Eclipse 环境下设计了一个基于语义理解的智能问答原型系统，系统包括智能问答交互界面、句子相似度计算界面、基于语义的查询扩展界面和数据可视化展示界面。

（1）智能问答交互界面

智能问答交互界面主要通过用户和系统的交互实现智能问答，同时在页面下方给予用户可提问的问句样式，如图 5.6 所示。

图 5.6　智能问答交互界面

用户在该页面中输入问句请求，系统后台判断问句类型（寒暄问句、查询问句）之后对问句进行相应处理，并返回相关答案。用户在问答系统交互界面进行问答时，经常会输入一些与真正查询意图无关的寒暄语，如“你好”、“在吗”，系统针对用户输入的寒暄语进行寒暄回答，以改善用户的使用感，给用户良好的体验。用户输入问句后，首先会在 FAQ 库中进行用户问句与 FAQ 库问句的匹配工作，若匹配成功则返回答案，提示用户是否对答案满意，如果用户不满意，则进行语义查询扩展，在本体中进一步查询，将答案返回给用户。

（2）句子相似度计算效果

为了清晰地展示本文的查询策略，系统设计了相似度计算展示模块，如图 5.7 所示，该部分展示了用户问句与 FAQ 库的匹配结果，分别使用多特征融合的句子相似度、余弦相似度与编辑距离相似度计算两个句子的相似度并返回结果。

图 5.7　相似度计算结果图

相似度计算模块中，系统首先接收用户的问句，系统使用本文所提多特征融合句子相似度计算方法遍历计算用户问句与 FAQ 库问句的相似度，将 FAQ 库中与用户问句相似度最高的问句输出给用户。同时，系统还提供了与余弦相似度、基于编辑距离相似度的对比数据，更加直观的展示了本文所提算法的性能。图 5.7 中，用户输入问句是“《数学之美》的出版社是什么？”，经相似度计算后用户问句最匹配的句子是“请问《数学之美》这本书是哪个出版社的？”，多特征融合的相似度算法计算结果为 0.79，而余弦相似度算法为 0.735，基于编辑距离的相似度算法为 0.421。

（3）查询扩展效果展示

查询扩展界面提供了用户处理结果、基于语义的查询扩展结果、无查询扩展结果及关键词查询扩展结果，如图 5.8 所示。

图 5.8　查询扩展效果展示图

基于语义的查询扩展是智能问答系统的核心模块，通过引入本体作为知识源，实现对用户查询的语义扩展，达到精准理解用户意图的目的。如图 5.8 所示，用户输入查询问句后，若 FAQ 库中检索不到相匹配的问句，则对用户问句进行自然语言处理的分词、去停等操作，提取用户查询关键词。图中用户问句“《数学之美》的出版社是什么？”经分词后、标注词性后得到的最终关键词是“《数学之美》”和“出版社”。之后使用本文所提基于语义的查询扩展方法对用户查询意图进行扩展，得到扩展词集为“数学之美、专著、书、著作、出版社、出版单位、图书出版商”共 7 个关键词，而无查询扩展和关键词查询扩展方法分别得到 2 个和 3 个查询关键词。

（4）数据可视化展示

在智能问答、相似度计算、查询扩展的基础上，本文设计原型系统提供了核心数据的可视化展示，包含领域本体的图形化展示以及 FAQ 库的展示。其中本体展示模块使用 Jena 包解析本体，调用 Protégé 提供的接口读取本体数据，利用 OWLViz 查看所构建本体的层次结构。FAQ 库展示模块使用 JDBC 连接数据库，读取本地 MySQL 数据库，使用 HTML 标签的表格模板进行展示，如图 5.9 所示。

图 5.9　数据可视化展示

5.4 本章小结

本章在前文提出的多特征融合的句子语义相似度计算方法和基于语义的问句查询扩展方法的基础上，设计并实现了基于语义理解的智能问答原型系统。本章首先介绍了系统体系架构，之后针对核心技术及功能模块的实现展开详细阐述，使用七步法构建了图书领域本体，设计并构建了 FAQ 库和寒暄问题库，设计 FAQ 库和语义问答相结合的问答策略，最终设计交互界面进行核心功能的可视化展示。

第 6 章　总结与展望

6.1 总结

搜索引擎和开放域问答系统的发展给人们获取知识提供了便捷，然而存在检索信息返回太多且不精确、关键字匹配方式的检索不能表达用户意图、检索经常返回语义无关答案等问题。语义 Web 技术是互联网的下一代延伸，赋予了互联网通用信息交换的能力，其中的本体技术具有良好的层次结构和较强的语义表达能力，自提出以来就得到了密切关注。将语义 Web 技术应用于问答系统，使用本体作为知识源，可以清晰的表达领域之间的概念及概念间关系，使得问答系统对用户查询增加语义理解能力，同时其形式化存储更易被计算机识别和计算。使用本体技术对用户问句进行语义分析，可以增加问答系统的语义理解能力，提高问答系统的效率。本文基于图书领域构建了领域本体和 FAQ 库，设计基于语义理解的智能问答原型系统。设计并构建 FAQ 库存储用户常用问题，可以高效的从问答系统中抽取答案，避免问答系统多次重复回答。本文的主要工作体现在以下几个方面：

（1）分析了本文的研究背景及意义，说明基于语义理解的问答系统建设的必要性。分别综述了问答系统、问句相似度计算及查询扩展的研究现状，阐述了问答系统发展的三个阶段，介绍了语义 Web 和问答系统中一些关键技术，包括语义 Web 及本体基本概念、本体构建及查询技术、问答系统体系结构、问句预处理技术、句子相似度计算技术等。

（2）提出了多特征融合的句子语义相似度计算方法以解决 FAQ 库问句和用户问句的匹配问题。将层次分析法和语义距离分别应用到结构相似度和语义相似度计算中，使用层次分析法计算句子的词形、词序、句长特征对应权重，使用最短路径算法计算加权本体图中语义距离，进而描述语义相似度。将结果相似度和语义相似度加权融合，计算句子相似度。

（3）设计基于语义的查询扩展方法，将富有丰富语义信息的本体应用于查询扩展，将用户查询映射到领域本体中，构造最小查询生成树，利用最小查询生成树与查询生成树之间的关系扩展查询关键词集，使用语义相似度对查询扩展词集进行二次筛选，从而获得更加全面和准确的结果。

（4）采集网络非结构化和半结构化数据资源，经数据处理后生成 FAQ 库，使用七步法构建图书领域本体，作为系统的数据源。设计 FAQ 库和本体结合的问答策略，设计并实现了基于语义理解的智能问答原型系统，系统分为五个层次，分别为基础设施层、数据获取层、知识资源层、核心技术层、应用层。

6.2 展望

本文研究的问答系统还存在很多不足，接下来的工作将会针对以下两方面问题进一步研究。

（1）本文提出的句子语义相似度计算方法将句子分为结构相似度和语义相似度，在语义相似度部分需要计算概念节点与根节点的语义距离，本体库的完善程度会影响语义相似度结果，若本体库规模较小，或本体中实体间关系定义错误，则相似度计算结果就会出现偏差。同时，从用户的角度来看，问句的谓词通常可以体现提问者的提问意图，更好的反应问句的语义信息。因此，下一步工作是完善领域本体库，并针对智能问答系统提取问句的谓词信息，将谓词信息归纳入语义相似度计算中，继续提升多特征融合的句子语义相似度算法的准确率。

（2）本文所提的基于语义的查询扩展方法在一定程度上解决了术语错配问题，对用户查询意图进行扩展，改善查询效果。然而智能问答系统的使用过程中会产生大量历史数据，这些历史数据通常可以表达用户的使用习惯和问答的重点领域，更加清晰的描述了用户的查询意图。因此下一步工作是有效地重用这些历史数据，从历史数据中抽取查询意图，进一步改进基于语义的查询扩展的查询效果。

参考文献

[1] 叶育鑫, 欧阳丹彤. 语义 Web 搜索技术研究进展[J]. 计算机科学, 2010, 37(01): 1-5.

[2] Altavist. 关于搜索引擎的统计调查数据[EB/OL]. http://www.sowang.com/9238/meiri/7.htm. 2018.

[3] Höffner K, Walter S, Marx E, et al. Survey on Challenges of Question Answering in the Semantic Web[J]. Semantic Web, 2016, 8(6): 895-920.

[4] Liu Y, Yi X, Chen R, et al. A Survey on Frameworks and Methods of Question Answering[C]// International Conference on Information Science and Control Engineering. IEEE, Beijing, 2016: 115-119.

[5] 曾帅, 王帅, 袁勇, 等. 面向知识自动化的自动问答研究进展[J]. 自动化学报, 2017, 43(9): 1491-1508.

[6] Yu X, Pan L, Yang F. Research of E-government Auto Answer Consulting System Architechture[C]// International Conference on Management of e-Commerce and e-Government. IEEE, 2011: 97-100.

[7] 张宁, 朱礼军. 中文问答系统问句分析研究综述[J]. 情报工程, 2016, 2(1): 32-42.

[8] Abacha A B, Zweigenbaum P. MEANS: A medical question-answering system combining NLP techniques and semantic Web technologies[J]. Information Processing & Management, 2015, 51(5): 570-594.

[9] Lende S P, Raghuwanshi M M. Question answering system on education acts using NLP techniques[C]// Futuristic Trends in Research and Innovation for Social Welfare. IEEE, Coimbatore, 2016: 1-6.

[10] Jayalakshmi S, Sheshasaayee A. Automated question answering system using ontology and semantic role[C]// International Conference on Innovative Mechanisms for Industry Applications. IEEE, Bangalore, 2017: 528-532.

[11] Turing A M. Computing Machinery and Intelligence[J]. Mind, 1950, 59(236): 433-460.

[12] Kolomiyets O, Moens M F. A survey on question answering technology from an information retrieval perspective[J]. Information Sciences, 2011, 181(24): 5412-5434.

[13] Diefenbach D, Lopez V, Singh K, et al. Core techniques of question answering systems over knowledge bases: a survey[J]. Knowledge and Information systems, 2018, 55(3): 529-569.

[14] Unger C, Freitas A, Cimiano P. An introduction to question answering over linked data[C]// Reasoning Web International Summer School. Springer, Cham, 2014: 100-140.

[15] 谭伟. 面向网络的中文问答系统相关技术的研究与系统初步实现[D]. 清华大学, 2005.

[16] 王波. 网络环境下高校图书馆参考咨询服务[J]. 现代情报, 2008, 28(4): 152-153.

[17] 潘鹏程. 图书馆智能咨询系统模型构建[J]. 图书馆学研究, 2010, 6(12): 82-84.

[18] 李舟军, 李水华. 基于 Web 的问答系统综述[J]. 计算机科学, 2017, 44(06): 1-7+42.

[19] Ferreira R, Cabral L D S, Lins R D, et al. Assessing sentence scoring techniques for extractive text summarization[J]. Expert Systems with Applications, 2013, 40(14): 5755-5764.

[20] Lord P W, Stevens R D, Brass A, et al. Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation[J]. Bioinformatics, 2003, 19(10): 1275-1283.

[21] Resnik P. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language[J]. Journal of artificial intelligence research, 1999,

11(1): 95-130.
[22] Kusner M, Sun Y, Kolkin N, et al. From word embeddings to document distances[C]// International Conference on Machine Learning. ACM, Lille, 2015: 957-966.
[23] Ruan H, Li Y, Wang Q, et al. A Research on Sentence Similarity for Question Answering System Based on Multi-feature Fusion[C]// 2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI). IEEE, Omaha, 2016: 507-510.
[24] Gokul P P, Akhil B K, Shiva K K M. Sentence similarity detection in Malayalam language using cosine similarity[C]// International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT). IEEE, Bangalore, 2017: 221-225.
[25] Madi N S A, Khan J I. Measuring learning performance and cognitive activity during multimodal comprehension[C]// International Conference on Information and Communication Systems (ICICS). IEEE, Irbid, 2016: 50-55.
[26] Tasi C S , Huang Y M , Liu C H , et al. Applying VSM and LCS to develop an integrated text retrieval mechanism[J]. Expert Systems with Applications, 2012, 39(4): 974-3982.
[27] 陈二静, 姜恩波. 文本相似度计算方法研究综述. 现代图书情报技术[J], 2017, 1(6): 1-11.
[28] 黄洪, 陈德锐. 基于语义依存的汉语句子相似度改进算法[J]. 浙江工业大学学报, 2017, 45(01): 6-9.
[29] 李茹, 王智强, 李双红, 等. 基于框架语义分析的汉语句子相似度计算[J]. 计算机研究与发展, 2013, 50(08): 1728-1736.
[30] 陈海燕. 基于搜索引擎的词汇语义相似度计算方法[J]. 计算机科学, 2015, 42(01): 261-267.
[31] 黄姝婧, 张仰森. 基于多特征融合的句子相似度计算方法[J]. 北京信息科技大学学报(自然科学版), 2017, 32(5): 45-49.
[32] 李连, 朱爱红, 苏涛. 一种改进的基于向量空间文本相似度算法的研究与实现[J]. 计算机应用与软件, 2012, 29(02): 282-284.
[33] 赵胜辉, 李吉月, 徐碧瑢, 等. 基于TFIDF的社区问答系统问句相似度改进算法[J]. 北京理工大学学报, 2017, 37(9): 982-985.
[34] 王小林, 肖慧, 邰伟鹏. 基于 Hadoop 平台的文本相似度检测系统的研究[J]. 计算机技术与发展, 2015, 25(08): 90-93.
[35] 谷重阳, 徐浩煜, 周晗, 等. 基于词汇语义信息的文本相似度计算[J]. 计算机应用研究, 2018, 35(2): 391-395.
[36] Carpineto C, Romano G . A Survey of Automatic Query Expansion in Information Retrieval[J]. ACM Computing Surveys, 2012, 44(1): 1-50.
[37] Raza M A, Rahmah M, Noraziah A, et al. A Taxonomy and Survey of Semantic Approaches for Query Expansion[J]. IEEE Access, 2019, 25(7): 17823-17833.
[38] Xu Y, Jones G J F, Wang B. Query dependent pseudo-relevance feedback based on wikipedia[C]// Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2009: 59-66.
[39] Oliveira V, Gomes G, Belém F, et al. Automatic query expansion based on tag recommendation[C]// Proceedings of the 21st ACM international conference on Information and knowledge management. ACM, Boston, 2012: 1985-1989.
[40] Bakhtin A, Ustinovskiy Y, Serdyukov P. Predicting the impact of expansion terms using semantic and user interaction features[C]// Proceedings of the 22nd ACM international conference on Information & Knowledge Management. ACM, New York, 2013: 1825-1828.
[41] Kotov A, Zhai C X. Interactive sense feedback for difficult queries[C]// Proceedings of the 20th

ACM international conference on Information and knowledge management. ACM, New York, 2011: 163-172.

[42] Dalton J, Dietz L, Allan J. Entity query feature expansion using knowledge base links[C]// Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval. ACM, New York, 2014: 365-374.

[43] 郝志峰, 陆印章, 温雯, 等. 结合相关规则和本体加权图的查询扩展[J]. 计算机应用研究, 2014, 31(10): 3028-3032.

[44] 杨清琳, 李陶深, 农健. 基于领域本体知识库的语义查询扩展[J]. 计算机工程与设计, 2011, 32(11): 3853-3856.

[45] 李维银, 石玉龙, 陈杰, 等. 基于分类模型的查询扩展方法[J]. 计算机科学, 2015, 42(6):18-22.

[46] 欧阳柳波, 谭睿哲. 一种基于本体和用户日志的查询扩展方法[J]. 计算机工程与应用, 2015, 51(1): 151-155.

[47] 叶雷, 高盛祥, 余正涛, 等. 基于事件元素无向图的查询扩展方法[J]. 中文信息学报, 2017, 31(1): 17-22.

[48] 徐博, 林鸿飞, 林原, 等. 一种基于排序学习方法的查询扩展技术[J]. 中文信息学报, 2015, 29(03): 155-161.

[49] Feng Z, Bo L, Zhen Z, et al. A Study of Semantic Web Services Network[J]. Computer Journal, 2018, 58(6): 1293-1305.

[50] Sabou M, Ekaputra F J, Biffl S. Semantic web technologies for data integration in multi-disciplinary engineering[C]// Multi-Disciplinary Engineering for Cyber-Physical Production Systems. Springer, Cham, 2017: 301-329.

[51] 王向前, 张宝隆, 李慧宗. 本体研究综述[J]. 情报杂志, 2016, 35(6): 163-170.

[52] Fang D U, Chen Y G, Xiao-Yong D U. Survey of RDF Query Processing Techniques[J]. Journal of Software, 2013, 24(6): 1222-1242.

[53] Hacherouf M, Bahloul S N, Cruz C. Transforming XML documents to OWL ontologies: A survey[J]. Journal of Information Science, 2015, 41(2): 242-259.

[54] Feng J, Meng C, Song J, et al. SPARQL query parallel processing: a survey[C]// 2017 IEEE International Congress on Big Data. IEEE, Honolulu, 2017: 444-451.

[55] Ritter A, Clark S, Etzioni O. Named entity recognition in tweets: an experimental study[C]// Proceedings of the Conference on Empirical Methods in Natural Language Processing. ACM, Stroudsburg, 2011: 1524-1534.

[56] Ebrahimi A P, Toloui A A, Mahdavy R M. A novel AIDS/HIV intelligent medical consulting system based on expert systems[J]. Journal of Education & Health Promotion, 2013, 2(1): 54-64.

[57] 张涛, 贾真, 李天瑞, 等. 基于知识库的开放领域问答系统[J]. 智能系统学报, 2018, 13(4): 69-75.

[58] 吴夙慧, 成颖, 郑彦宁, 等. 文本聚类中文本表示和相似度计算研究综述[J]. 情报科学, 2012, 30(04): 622-627.

[59] 宗成庆. 统计自然语言处理[M]. 北京: 清华大学出版社, 2013.

[60] 奉国和, 郑伟. 国内中文自动分词技术研究综述[J]. 图书情报工作, 2011, 55(2): 41-45.

[61] 王春柳, 杨永辉, 邓霏, 等. 文本相似度计算方法研究综述[J]. 情报科学, 2019, 37(03): 158-168.

[62] Mihalcea R, Corley C, Strapparava C. Corpus-based and Knowledge-based Measures of Text Semantic Similarity[C]// National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, ACM, Boston, 2006: 775-780.

[63] Jesús Oliva, José Ignacio Serrano, María Dolores del Castillo, et al. SyMSS: A syntax-based measure for short-text semantic similarity[J]. Data & Knowledge Engineering, 2011, 70(4): 390-405.

[64] 邓雪, 李家铭, 曾浩健, 等. 层次分析法权重计算方法分析及其应用研究[J]. 数学的实践与认识, 2012, 42(7): 93-100.

[65] Dulisch N, Kempf A O, Schaer P. Query expansion for survey question retrieval in the social sciences[C]// International Conference on Theory and Practice of Digital Libraries. Springer, Cham, 2015: 28-39.

[66] Gennari J H, Musen M A, Fergerson R W, et al. The evolution of Protégé: an environment for knowledge-based systems development[J]. International Journal of Human-computer studies, 2003, 58(1): 89-123.

[67] Mcbride B. Jena: a semantic Web toolkit[J]. IEEE Internet Computing, 2002, 6(6):55-59.

[68] 高劲松, 梁艳琪, 王学东, 等. 学科知识地图的本体构建方法研究[J]. 情报科学, 2013, 31(07): 72-77.

[69] 任飞亮, 沈继坤, 孙宾宾, 等. 从文本中构建领域本体技术综述[J]. 计算机学报, 2017, 40(5): 1-26

[70] 岳丽欣, 刘文云. 国内外领域本体构建方法的比较研究[J]. 情报理论与实践, 2016, 39(8): 119-125.

[71] 韩道军, 甘甜, 叶曼曼, 等. 基于形式概念分析的本体构建方法研究[J]. 计算机工程, 2016, 42(2): 300-306.

攻读学位期间取得的研究成果

奖学金：

[1] 获得 2018 年硕士研究生国家奖学金；

[2] 获得 2017-2018 学年校级二等奖学金；

[3] 获得 2016-2017 学年校级二等奖学金。

科研论文：

[1] 翟社平, **李兆兆**, 段宏宇, 等. 多特征融合的句子语义相似度计算方法[J]. 计算机工程与设计.（已录用，2019 年正刊发表）

[2] Zhai S, **Li Z**, Duan H, et al. A Service Agents Division Method Based on Semantic Negotiation of Concepts[C]// International Conference on Intelligent Science and Big Data Engineering. Springer, Cham, 2018: 57-67. (EI: 20184806144953)

[3] 翟社平, **李兆兆**, 段宏宇, 等. 区块链关键技术中的数据一致性研究[J]. 计算机技术与发展, 2018, 28(09): 94-100.

科研项目：

[1] 大数据环境下基于语义的网络舆情分析研究，陕西省社会科学基金，项目编号：2016N008（参与者）

[2] 基于大数据的语义网络舆情分析研究，西安市社科规划基金项目，项目编号：17X63（参与者）

[3] 基于语义 Web 的智能问答系统的设计与实现，西安邮电大学研究生创新基金项目，项目编号：CXL2016-24（主持人）

[4] 基于语义 Web 的旅游服务系统关键技术研究，西安邮电大学研究生创新基金项目，项目编号：CXL2016-13（参与者）

[5] 我国信息产业驱动区块链技术应用演进研究，工业和信息化部通信软科学项目，项目编号：2017-R-22（参与者）

[6] 我国区块链技术发展及创新实施策略研究，工业和信息化部通信软科学项目，项目编号：2018-R-26（参与者）

科技竞赛：

[1] 第三届全国大学生物联网技术与应用“三创”大赛全国二等奖

[2] 第四届全国大学生物联网技术与应用“三创”大赛全国三等奖

[3] 第四届中国“互联网+”大学生创新创业大赛陕西省铜奖

[4] 第十二届中国研究生电子设计竞赛西北赛区三等奖
[5] 第十三届中国研究生电子设计竞赛西北赛区二等奖
[6] 第十三届中国研究生电子设计竞赛商业计划书专项赛二等奖
[7] 全国移动互联创新大赛陕西赛区高校组三等奖
[8] 陕西省第三届研究生创新成果展省级三等奖
[9] 第八届蓝桥杯全国软件和信息技术专业人才大赛陕西赛区三等奖
[10]第三届中国“互联网+”创新创业大赛，获校级三等奖

致谢

这已经是我在西安邮电大学度过的第七个年头了。蓦然回首，一转眼，时光匆匆如流水。从本科到研究生，从学妹到师姐，念及此，百感交集。

还记得第一次进入实验室的紧张，记得小组汇报的忐忑，记得熬夜同老师讨论的精神振奋，记得一起参加比赛的风雨同舟，记得拿到奖励的喜悦，记得任务出错的自责。记得许多，不舍忘却。有过彷徨，有过迷茫，有过抱怨，更多的，却是感谢。

感谢我最敬爱的导师——翟社平老师。从本科起，翟老师的严谨认真就给我留下了深刻的印象，研究生三年更让我深刻的了解到翟老师的学识渊博、治学严谨、谦虚平和。从论文的选题、开题答辩、中期汇报，到初稿、不断完善定稿，论文的每一个细节都是在老师的细心指导下完成的，我非常敬佩翟老师严谨务实的研究精神与诲人不倦的优良美德。在这三年的时间里，翟老师不仅为我提供了优越的科研环境，同时还在生活上给予了我很多关怀，不仅教会了我学习上的方式和方法，他的谦逊温和、团队优先的作风更是教会我做人的道理。十分有幸能做翟老师的学生，感谢他的鼓励和支持，也感谢他的批评和责备，我将带着他的期望和良苦用心继续前行。

感谢答辩组老师们给我的一些指导意见和建议，从开题到中期检查，再到毕业答辩，正因为有了他们的意见，我的毕业设计才能更加顺利地完成。

感谢各位评审专家百忙之中对论文的仔细审查，他们提出的意见和建议使得论文更加完整。

感谢实验室的大家庭，感谢我的师兄、师姐们，感谢我的同门，感谢我的师弟、师妹们。我知道，我们就像是一家人一样，我们互相鼓励，互相学习。感谢一直以来他们对我的帮助和包容，感谢他们对我的支持和鼓励，一起奋斗的日子我不会忘。

感谢所有帮助过我的老师和同学，感谢葛茂老师、杨锐老师、邢高峰老师，感谢我的英语老师杨维东老师，感谢图书馆的张老师。感谢我的舍友、一起上课的同学们。在我多次向他们伸出援手的时候，他们都无私的帮助了我，耐心帮我解决问题，真诚的感谢他们。

感谢我的男朋友李先生，在我研究生的三年里，他一直默默陪伴着我，难过时的安慰、迷茫时的指点、开心时的鼓励、失落时的加油打气。感谢他的包容和关怀，给了我前进的动力。

感谢挚爱我的父母和家人，感谢他们的支持和陪伴。父母和家人总是在背后默默付出，给予我无微不至的关怀和鼓励，在我成功时为我骄傲，在我失意时给我安慰。正是他们的支持和关心，我才能专心完成我的学业。

忆往昔，峥嵘岁月；看今朝，潮起潮落；往未来，任重而道远。愿，不忘初心，方得始终！

西安邮电大学
学位论文原创性声明

秉承学校严谨的学风和优良的科学道德，本人郑重声明：所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含本人已用于其他学位申请的论文或成果。与我一同工作的同志对本研究所做的任何贡献均已在论文中做了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切的法律责任。

本人签名：_____ 日期：2019 年 5 月 16 日

西安邮电大学
学位论文知识产权声明

本人完全了解西安邮电大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属西安邮电大学。学校有权保留并向国家有关部门或机构送交论文的复印件和电子版。本人允许论文被查阅和借阅。学校可以将本学位论文全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复印手段保存和汇编本学位论文。同时本人保证，毕业后结合学位论文研究课题再撰写的文章一律署名单位为西安邮电大学。

本人签名：_____ 日期：2019 年 5 月 16 日

导师签名：_____ 日期：2019 年 5 月 16 日

习）”来阐释“智能机器”，即从一个相对而言较为简单的机器来开始，通过让其经历一系列所谓的学习“经验”，逐步将其转化为一套更加智能的机器，而且还能够被用来处理更为广泛的各类偶发事件。此后相关专家学者对智能问答系统展开了深入研究，其发展大致可分为三个阶段。

第一阶段是始于上世纪 60 年代末的基于模式匹配的专家库，该类问答系统使用数据库检索技术在二维的结构化数据表中进行检索，需要预先构建大规模领域词典以及检索规则，最具代表性的是 1961 年的 BaseBall 和 1972 年的 LUNAR。这类系统可以通过自然语言完成问答，但其数据存储及查询受关系型数据库的限制，不适用于非结构或半结构化数据，对领域词典重度依赖，自动获取知识的能力存在瓶颈。

从上世纪 90 年代起，以信息和信息检索技术为核心的问答系统逐渐取代了基于模式匹配的专家库。该阶段的问答系统建立特定的框架并组建知识库，其底层数据支撑是一些非结构化、半结构化或无结构的原始文档、网页等，检索时首先在大量的原始信息中检索与查询相关信息，之后进一步抽取精确的答案。这类问答系统的典型代表是基于 Web 的自动问答系统 Start、基于 FAQ（Frequently Asked Questions，常见问题解答）数据集的问答系统 Finder、基于搜索引擎的 MULDER 等。该类问答系统在非结构化及半结构化的数据处理上占一定优势，然而系统以关键词作为索引进行查询，缺乏对知识的深度理解，无法保证数据准确性。

随着大数据、深度学习、知识管理等技术的飞速发展与深度应用，智能问答系统进入了以知识和知识自动化为中心的阶段。国外许多专家学者在知识表示的不同层面进行研究，由此衍生出不同类别的问答系统。基于逻辑表示的问答系统将知识看作一组逻辑公式的集合，对知识的更新则转化为对逻辑公式的增加、删除或修改[12]。基于框架表示的问答系统设计特定领域的框架表示存储特定对象的所有知识，使用框架语义分析生成答案[13]。基于语义 Web 的问答系统将语义 Web 的本体、数据推理等技术应用于智能问答系统，使用以语义网络构建的知识库进行知识表示，知识查询综合运用模式匹配、语义解析、SPARQL（SPARQL Protocol and RDF Query Language，SPARQL 协议和 RDF 查询语言）等技术[14]。基于语义 Web 的智能问答系统能够以形式化的方式表示语义，提高了异构系统之间的互操作性，进一步提高了问答系统的智能性和精确性。使用语义知识库替代传统基于关键词文本的知识表达，实现智能问答，利用知识库自动生成用户答案，提升问答系统处理复杂问题的能力，已经成为问答系统未来的发展趋势。

与国外的研究水平相比，中文问答系统不管是算法研究还是框架研究都相对滞后。不同于英文自然语言，中文自然语言拥有灵活丰富的表现形式，并且中文自身的语法结构极其复杂，在处理时缺乏基础性资源如大型通用知识库，也缺少对应的评价机制。同时，对于通用问题的解决目前并不能使用自然语言处理工具完成，在系统搭建和设计时也无法直接使用国外成熟技术。因此中文问答系统具有较高的研究及应用

价值，大批科研机构和高效研究者纷纷展开研究，取得了一定的研究成果。

谭伟[15]使用浅层语义分析方法处理用户问句，建立索引模型提高检索效率，最终利用语义框架结构间的相似性抽取答案。王波[16]针对标准咨询服务的作用及现存问题展开研究，深入分析其需求类型及自身特点，提出了咨询服务的改进意见。潘鹏程[17]主要面向图书馆智能咨询系统展开研究，探讨图书馆场景下问答的基本需求，通过分析现有智能问答系统的体系架构及特点，构建了图书馆领域的智能问答系统模型。而在垂直领域，多是社区问答或基于 FAQ 的问答，同时一些基于知识库的问答只停留在实验性质，底层数据较少，也没有应用于实际，如基于美食本体、农业本体、航空领域等的问答。另外，不少企业也逐渐推出智能问答机器人[18]，如微软精炼了千万条真实语料库开发聊天机器人“小冰”、百度依据其自身搜索技术推出“小度机器人”、京东基于自身客服问答大数据设计并研发“JIMI 智能客服”等。

中文问答系统以自然语言形式与用户进行交互，具备更高的智能性和准确性，在一定程度上弥补了传统搜索引擎的不足，同时也促进了问句检索和查询扩展等技术的相关研究。本文针对基于语义理解的智能问答系统开展研究工作，针对问答系统问句分析和信息检索过程中存在的问题，重点研究句子语义相似度的计算以及基于语义的问句查询扩展。

1.2.2 句子相似度计算

句子的语义相似度是计算语言学中的一个度量，表示依赖于它们的层次关系的两个概念的共性。句子相似度计算是计算两个句子内在含义之间的相似度，作为自然语言处理的研究重点，已经广泛应用于文本分类、智能问答、信息检索等领域。针对句子相似度计算方法，国内外学者在不同角度进行了深入研究。

国外学者如 Ferreira 等人[19]在词法和句法分析的过程中引入了语义分析，提出了一种基于内容的相似度度量方法，然而并未考虑到句子长度的影响，存在一定局限性；Lord 等人[20]提出使用树状结构知识库的父节点所共享的内容计算概念节点相似度；Resnik 等人[21]基于公共父节点概念中信息量最大的节点的信息内容进行相似度计算；Kusner 等人[22]利用词向量空间技术，将相似度描述为某文档词汇移动到另一文档中对应词汇所需要的最小移动距离；Haipeng Ruan 等人[23]利用 Word2vec 和词嵌入方法计算相似度，其结果与形态相似度结合，最终句子整体相似度使用人工神经网络计算；Gokul P.P 等人[24]针对马拉雅拉姆语的特征，使用关键词的同义词结合余弦相似度算法查找句子中相似短语；Nase S Al Madi 等人[25]将语义概念上下文、位置及信息获取时间等要素结合，提出一种基于语义理解的文本相似度计算框架；Tasi 等人[26]综合使用 VSM（Vector Space Model，向量空间模型）和 LCS（Longest Common Subsequence，最长公共子序列）方法，并对文本序列赋以权重，在一定程度上提高了准确率。

国内也有不少学者展开相似度计算的研究，陈二静等人[27]借鉴了 Gomaa 等人的分类框架，将文本相似度计算方法分为基于字符串的方法、基于语料库的方法、基于

世界知识的方法以及其他方法；黄洪等人[28]对句子中的动词、形容词等搭建依存树计算句子相似度；李茹等人[29]采用语义框架描述句子，进而计算相似度；陈海燕等人[30]将两个关键词共同出现的片段定义为语义片段，使用搜索引擎技术分析网页的语义片段数计算文本相似度；黄姝婧等人[31]在语义特征中着重考虑反义信息和否定信息，并与词频、词序等特征结合，构建句子相似度计算方法；李连等人[32]在传统 VSM 算法的基础上进行改进，引入表征文本特征词覆盖的参数优化了文本相似度计算结果；赵胜辉等[33]针对问答系统的特征，充分考虑用户的问答意图，制定问句分类标准，改进了 TF-IDF（Term Frequency–Inverse Document Frequency，逆文本频率指数）算法来计算问句相似度；王小林[34]等人在 Hadoop 平台上使用信息熵和信息增益改进 TF-IDF 算法，并且使用语义加权因子提高了相似度的准确度；谷重阳等[35]将文本向量之间的关系表示为词汇相似度，改进了基于余弦的文本相似度方法。

1.2.3 查询扩展技术

当前信息检索系统中最主要的问题是术语错配问题，即索引器和用户在描述文档的概念时所用的术语存在较大差异性[36]。有学者认为，对一个概念使用相同词汇表达的可能性小于 20%，造成这一比例较低的主要原因是同义词及多义词的使用。同义词和多义词可能导致检索错误、检索到无关文档等问题，进而导致系统的准确率和召回率降低。

查询扩展技术[37]是解决术语错配问题最有效的技术之一，是查询优化的一个分支研究方向，其基本思想是将原始查询扩展到最能反映用户实际查询意图的其他词，或简单地生成更有用的查询词，从而对相关文档展开二次检索，弥补单一关键字检索的不足，减少相关文档遗漏，提高了信息检索的完整性和准确性。查询扩展技术可简单分为四个步骤，即数据源处理、术语权重赋值和排序、术语选择、查询重表示。除信息检索领域外，查询扩展技术还在其他应用领域如智能问答、跨语言信息检索、信息过滤、文本分类等方面取得了较好的应用。近年来，不少学者针对查询扩展技术展开研究，提出了多种查询扩展方法。

Xu Y 等人[38]根据文档的共现信息发现词之间的相似性，利用词汇之间相似性实现查询扩展；Oliveira V 等人[39]通过分析用户的查询日志来扩展查询，建立查询空间与文档空间之间的关系。这些方法主要侧重于保持扩展词和查询词的同义性的同义词扩展，以及扩展查询和原始查询的同义词，而忽略了概念之间的语义关系，用户的真实意图无法从根本上表达出来。因此有学者开展语义扩展研究，Bakthtin A 等人[40]在检索前估计每个扩展项对查询性能的影响进行查询筛选，以提高查询扩展效率；Kotov A 等人[41]通过交互反馈让用户参与查询扩展消歧，在对文档进行全局分析的基础上提出一种查询词语义识别方法；Dalton J 等人[42]提出了一种实体查询特征扩展技术，通过丰富实体的特征及其与知识库的链接来实现查询扩展。

国内学者郝志峰等人[43]借助相关规则挖掘和 WordNet 知识库，构建了加权词语

关系图，根据加权图的结构和权重信息进行查询扩展；杨清琳等人[44]设计了基于领域本体的语义检索系统应用架构，使用语义距离表示的语义相似度实现关键词的查询扩展；李维银等人[45]通过用户查询词的多种特征进行简单扩展，使用朴素贝叶斯分类模型对查询扩展词进行分类筛选，进一步优化扩展词集合；欧阳柳波等人[46]使用本体扩展用户查询词，之后采用共现分析方法结合用户查询日志对扩展词集进行二次筛选；叶雷等人[47]针对新闻领域的特征，设计基于事件元素无向图的方法，利用新闻事件之间的相关性展开查询扩展；徐博等人[48]将排序学习方法引入到查询扩展中，从语料库中提取扩展词相关特征，并训练扩展词排序模型进行扩展词集的重排序。

1.3 研究内容

本文构建图书领域的领域本体作为知识库，设计基于 FAQ 库的问答和基于语义的问答相结合的问答策略，将语义 Web 关键技术与传统问答系统结合，设计并实现基于语义理解的智能问答系统。FAQ 库存储常识性问题，用户提问经问句处理首先与 FAQ 库问句进行匹配，如果匹配失败，系统转向基于本体的问答模块，基于本体查询扩展产生答案。在 FAQ 问答模块中使用所提基于多特征融合的语义相似度算法计算用户问句与 FAQ 库问句的语义相似度，从 FAQ 库中进行常识性基础问题的问答查询；基于语义查询扩展方法扩展用户查询关键词，使用 SPARQL 语言查询本体库生成答案。基于以上内容，本文的主要研究内容分为以下几点。

（1）用户问句与 FAQ 库问句的匹配方法研究

针对问句对之间的匹配问题展开研究，综合考虑句子的结构信息与语义信息，提出了一种多特征融合的句子语义相似度计算方法。该方法提取句子的词形特征、词序特征及句长特征计算结构相似度，基于本体概念的语义距离计算语义相似度，对结构相似度和语义相似度进行特征加权构建多特征融合的句子语义相似度计算方法。实验对比了传统余弦相似度算法和基于关键词的相似度算法，验证了本文算法的有效性。

（2）问句查询扩展方法研究

针对查询中的术语错配问题展开研究，设计并实现基于语义的问句查询扩展方法。首先将用户查询问句映射至领域本体中，根据领域本体对用户查询进行初始扩展，使用改进最小生成树算法构造最小查询生成树，在查询词筛选时使用语义相似度和编辑距离相似度结合的策略，最终生成查询扩展词集。实验对比了无查询扩展方法和关键词查询扩展方法，验证了方法的合理性。

（3）智能问答策略设计与实现

以句子语义相似度计算和语义查询扩展为核心，设计了 FAQ 库问答和语义问答结合的问答策略，设计并实现基于语义理解的智能问答原型系统，该原型系统主要包括基础设施层、数据获取层、知识资源层、核心技术层和应用层五个层次。在该原型系统中，设计相似度计算模块和基于语义的查询扩展模块对所提算法进行验证，同时

提供可视化界面对知识源进行展示。

1.4 论文组织结构

根据 1.3 节给出的研究内容，本文共分为 6 个章节，组织结构的规划安排如图 1.1 所示。

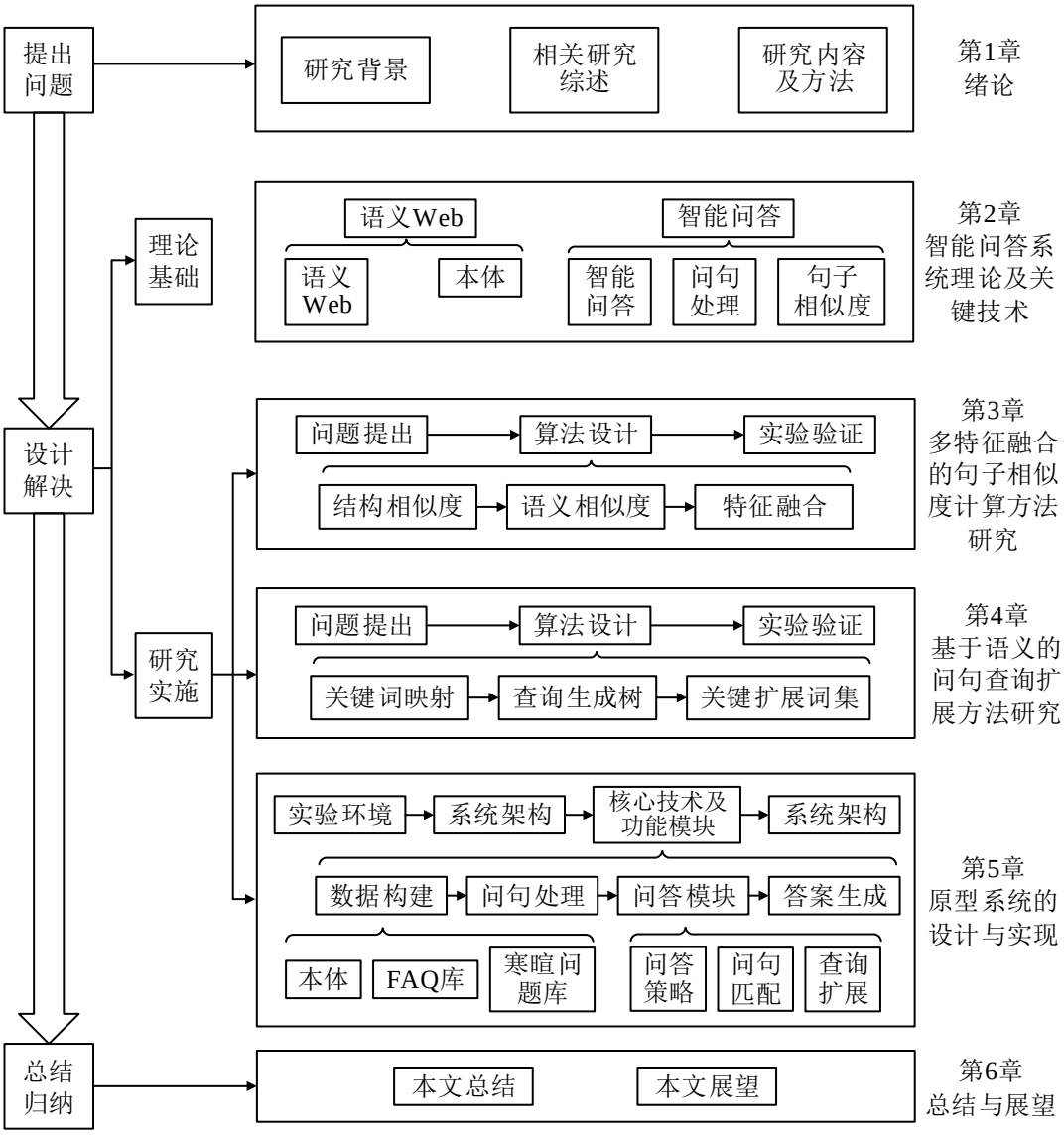

图 1.1　论文组织结构

第一章为绪论。阐述智能问答系统的研究背景及意义，分别论述了问答系统、句子相似度计算、查询扩展技术的研究现状，讨论了本文的主要研究内容，最终通过图形化方式呈现了本文的组织结构。

第二章为智能问答系统理论及关键技术。详细介绍了语义 Web 基础理论、本体基本概念、本体的构建与查询等；介绍智能问答系统关键技术，包括智能问答系统体系结构、问句预处理、句子相似度计算等技术。