班级：计科1503

学号：2015014357

北京化工大学

毕业设计（中期报告）

**题目 领域问答系统的设计与实现**

**专 业 计算机科学与技术**

**学 生 黄奕达**

**指导教师 万静**

2019**年**4**月**12**日**

|  |  |  |
| --- | --- | --- |
| **是否开题** | **指导教师签字** | **答辩小组组长签字** |
| **同意** |  |  |
| **不同意** |  |  |

目 录

[第1章 课题来源及背景 1](#_Toc448086163)

[第1.1节 课题背景和意义 1](#_Toc448086164)

[第1.2节 国内外研究现状 1](#_Toc448086165)

[第2章 主要内容 3](#_Toc448086166)

[第3章 已完成的工作 4](#_Toc448086167)

[第3.1节 文献查阅与翻译 4](#_Toc448086168)

第3.2节 知识图谱的构建 4

[3.2.1 python爬虫 5](#_Toc448086172)

[3.2.1 数据表示形式以及相应框架 6](#_Toc448086172)

[第3.3节 问题分类 6](#_Toc448086171)

[3.3.1 卡方统计量 7](#_Toc448086172)

[3.3.2 TF-IDF 7](#_Toc448086173)

[3.3.3 SVM分类器 9](#_Toc448086174)

[第4章 尚需完成的工作 9](#_Toc448086176)

[第5章 存在的问题及解决的办法 10](#_Toc448086177)

[第6章 具体时间安排 10](#_Toc448086178)

# 第1章 课题来源及背景

## 第1.1节 课题背景和意义

随着互联网技术的发展与信息资源的爆炸性增长，传统的搜索引擎技术的劣势愈发明显。根据用户的查询而产生的快速排序网页，虽然有着较高的信息检索效率，但是并不意味着能够为用户提供快捷准确的信息资源。当用户查询信息时，要先从查询内容中提取出关键词，但是大部分用户难以从自己的问题中提取出真正准确有效的关键词，因此依照用户给出的关键词，对于搜索引擎反馈的大量结果，还需要进行人工排查和筛选，从而在面对当今爆炸增长的信息资源时，难以满足人们的需求。而知识图谱技术的出现，为解决面对大量数据的信息检索问题提供了新的思路，同时，知识图谱的这些相关特性对于问答系统也提供了新的思路，使得问答系统的准确性与实用性都有了新的提升。

## 第1.2节 国内外研究现状

1. 知识图谱的构建

对于知识图谱的构建，各种信息处理的方法有所不同，主要区别在于处理的是结构化数据还是非结构化数据。结构化数据可对其直接进行处理，而非结构化数据则需要人们对其再加工，此处我们主要讨论的是非结构化数据的信息抽取。信息抽取主要由实体抽取和关系抽取这两个步骤来完成，其中实体抽取又叫做命名实体识别（named entity recognition, NER），目的是从文本中找出专有名词或者所需要的词，刘峤[1]等人将这一技术分为三类：基于规则，统计机器学习方法，面向开放域。早期研究主要采用基于规则的方法，但是该方法繁琐且有局限，难以适应数据的变化，因此随后人们将目光放在了统计机器学习方法这一方向上，主要包括RANIBER[2]所述的隐马尔可夫模型（Hidden Markov Model,HMM），HARRIES[3]提出的条件随机场（Condition Random Field，CRF）和LAPTEV[4]所提出的支持向量机（Support Vector Machine，SVM）等方法。Cheng等人[5]提出的BI-LSTM-CRF模型结合了条件随机场与LSTM神经网络，在BI-LSTM层后添加了CRF层，设定矩阵为BI-LSTM的输出，进行训练，该方法结合了上下文的特征并且考虑了句子前后的信息，提高了准确率。基于统计机器学习的方法往往与基于规则的方法结合起来使用，在基于规则的特征构建基础之下，统计机器学习方法可以发现新的规则，并且不需要研究者的干预，会产生良好的效果，这种方法直到现在也在被广泛使用。面向开放域的实体抽取不再限制于单一领域的内容，而是研究和解决面向全互联网的信息抽取。杨等人[6]在文章中指出，面向开放域的问答系统，其问题的类别划分极为重要，对结果的正确性影响最大，问题的分类体系则大多是提前由人工制定好，因此在真正进行计算的时候往往会对结果有所限制，KEHLENBECK[7]提出的bootstrapping方法可以将少量已经限定的信息，作为内容的全部，对其他未进行限定的内容进行处理，来为少量的信息作出补充，此方法常用在构建数据集时的，通过再抽样来扩充样本。

关系抽取的目的是为了得到文本中零散实体之间的对应关系，从而将实体连接起来形成知识结构。早期的关系抽取是对应基于规则的实体抽取，这种关系抽取往往需要人工干预，并且由于技术原因，早期的规则并不全面，这就导致了关系抽取的不准确。随着统计机器学习的发展，在关系抽取方面又有了新的尝试，DU[9]提出了卷积神经网络，杨[8]使用了卷积神经网络[9]，对实体进行处理，再用bootstrapping思想，依照少量的标注样本生成足够多的数据，这种方法可以大量减少人工工作量，并且在训练的时候不需要人力投入。在神经网络的基础上加上注意力机制的思想，可以对于问题不同的部分有所侧重，识别出主要的特征，来增加解决问题的效率。史[10]用双向长短时记忆网络表示问句，用注意力机制获取特征并进行分类，然而该方法仍会受到问句哨声影响。

1. 问句语义分析

根据文章[8]，问题语义分析主要分为以下三种方法：语义解析法，信息抽取法，向量建模法。语义解析法是通过将问句转化为某种逻辑形式，使得知识库可以理解问题，并推理，构建相应的查询语句找出答案。其中较为出名的是Berant等人[11]的方法，他们用语义解析法得到了很好的结果，取得了成绩，但是其独创的进行语法树节点合并的桥接方法，最终会生成大量的语法树，复杂度很高，因此需要更对的人力投入，对其进行训练，方可达到效果。信息抽取法通过提取问题中的实体，依靠在知识库的查询，构建以该实体节点为中心的知识库子图，对于这个图，其中的每一个节点都是候选答案。通过对于问题的分析得到问题和候选答案的特征，用分类器筛选从而得出结果。Yao等人[12]的文章通过对问题构建语法依存树，将树转化为问题图并进行分类，最终得到了结果，其在F1-score下相对于其他部分方法有所提升，但是该文章中构建的特征比较稀疏，并且涉及到了许多先验知识，因此如果用深度学习的方法，会对这些缺点有所弥补。向量建模法是将问题与答案向量化，构建相应的模型找出问题中的关键词，通过相应的计算方法来计算问题与答案的匹配程度，从中选出最匹配的答案作为结果。Bordes等人[13]，用Embedding空间对于知识库中的实体与关系映射与问题中的关键词进行处理，用*S*(*q*,*a*)=*f*(*q*)⊤*g*(*a*)作为计算匹配程度的函数来得出结果，该方法的效果在当时是最好的结果，而且不需要借助额外的结构或是人工操作，是很好的方法。

1. 知识图谱的存储

由于知识图谱的表示存储方式不同，其存储的使用工具也不尽相同，知识图谱的存储方式中主要的两个是RDF格式和图格式。由xia[14]等人所述，RDF格式往往与三元组相关联，存储内容为主语谓语宾语，是一个句子的主要元素。对于基于关系型数据库的存储方法，常用的有Mysql，Oracle，在支持RDF的数据库中，Jena，Virtuoso为人们所广泛使用。而对于图结构的知识图谱，康[15]做了对于Neo4j的研究，Neo4j作为世界范围内使用量最高的图数据库，其有良好的并发性能，有着高速的图形算法，依赖云计算环境的分布式存储，支持语言丰富，提供了良好的服务。

# 第2章 主要内容

本课题将设计并实现一个医疗领域的领域问答系统。

（1）首先构建数据集，形成知识库，通过爬虫获得数据，通过神经网络训练数据集。

（2）编写系统与用户交互的界面，实现信息传递。

（3）设计系统的语义分析模块，通过条件随机场算法基于问题模板对于问句进行命名实体识别，通过语义依存树对问题进行分类，并实现相应的答案模板。

（4）将以往的各种方法的实现加以比较，使用较好的方法并加以改进。

（5）使用数据库neo4j，对领域问答系统的知识，三元组进行存储，对于每一个问题，依据它的的类别，给出答案的关键信息。

# 第3章 已完成的工作

## 第3.1节 文献查阅与翻译

查阅相关的文献和参考资料，来了解领域问答系统的相关内容。

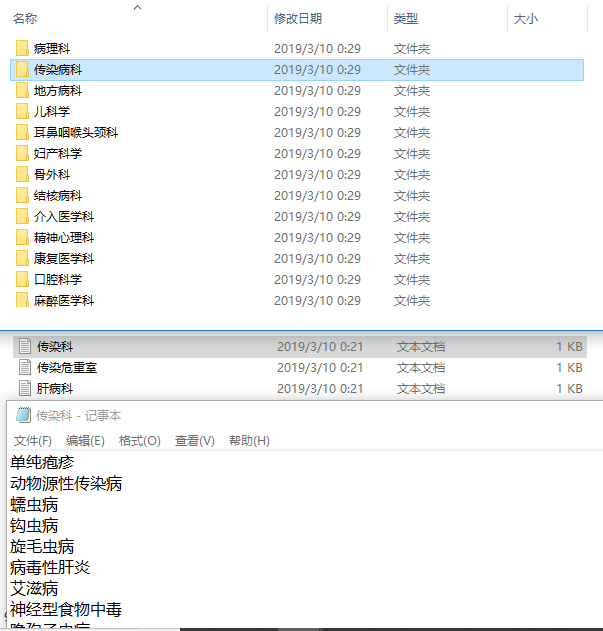
基于知识图谱的领域问答系统的实现步骤主要为数据库构建，问题分类，命名实体识别，关系抽取和答案抽取。对于每一个步骤，不同的文献介绍了不同的方法。经过阅读与对比分析，本文采用的方法如下：在数据库构建过程中，通过python爬虫爬取信息，并构建rdf数据，也就是三元组形式的数据。使用jena框架和sparql，在java中对相应的数据进行操作。在问题分类过程中，自己构建问题数据集并对相应的问题数据，用卡方的方法找到问题的关键字，对于一个问题中的所有关键字，计算tf-idf，形成问题的向量，最后将向量组成的数据输入SVM进行训练，这样就完成了问题分类模型的训练。而在用户的问题输入后，系统将先用训练好的SVM模型对问题进行分类，随后对问题进行命名实体识别，本文通过查询实体表，来获得问题中的实体，对于查询出的实体，使用深度学习的方法进行语义相似度分析，再在查询图中生成最终的答案。

## 第3.2节 知识库的构建

对于知识库的构建，首要的部分是对于数据的爬取，以及数据的表示方式。本课题使用python爬虫爬取医疗网站“好大夫”，“春雨医生”上的数据，并且将数据整理成RDF格式，再在系统中使用jena框架，对RDF数据进行操作，使用SPARQL，对三元组数据进行查询。

### 3.2.1 python爬虫

使用python爬虫爬取网站上的数据，按照，疾病，药品，科室，简介这些信息，对数据进行爬取与整理，得到的部分数据如下：



**图3-2爬虫得到的部分数据**

### 3.2.2 数据表示形式以及相应框架

本课题将数据整理成RDF格式数据，以三元组的形式进行存储，同时使用jena框架对RDF数据进行操作，使用SPARQL对RDF数据进行增删查改。Jena是一个java框架，用来支持对于，XML，RDF等类型数据的操作。SPARQL是一种专门针对RDF的查询语言和数据获取协议，其以三元组的方式对RDF数据进行相应的操作，而一个SPARQL语句就是一个RDF图。如下面的一个查询语句：

PREFIX owl:< <http://www.w3.org/2002/07/owl#>>

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

PREFIX exampleo: <http://www.example.com/ontology/#>

PREFIX example: <http://www.example.com/>

SELECT ?a ?b ?c WHERE {

example: “小柴胡颗粒”?a ?b .

?b rdf:type ?c .

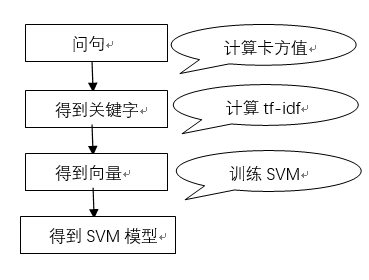
?c rdf:type owl:Class .

}

由PREFIX对应的语句是对知识库的配置，下面的带有变量a，b，c的语句则是带有变量的查询图，查询的是与实体“小柴胡颗粒”相连接的实体，也就是包含实体“小柴胡颗粒”的三元组。

## 第3.3节 问题分类

截止到中期，实现的分类方法主要是按照如下的思想：对于问题数据集，使用卡方的方法，找出这个问题之中的关键词，而对于这些关键字，更进一步的计算他们对应的tf-idf值，构成相应的向量，对于这些生成的向量，输入到SVM之中，训练相应的模型，这样，对于用户输入的新的问题，便可以用训练好的SVM模型进行分类。



**图3-1 基于SVM模型的问题分类流程**

### 卡方统计量

其思想是通过判断该词在相应类别与其他类别出现与不包含的次数，进行运算，得到卡方值，其值大小就表示这这个词对于这个类别的重要性，也就是衡量这个词能否成为这个句子的关键字的因素。

通过计算：

N:训练数据集中句子总数

A:在该类别中包含该字的句子总数

B:在其他类别中包含该字的句子总数

C:在该类别不包含该字的句子总数

D:在其他类别不包含该字的句子总数

公式如下：

（4·1）

### TF-IDF

本文采用TF-IDF方法，对选出的关键字计算权值，从而为每一个问题构成向量，其中TF是词频，IDF是逆文本频率指数。二者的乘积就是要求得到权值，TF-IDF。TF，也就是词频，其计算方法是计算这个词在相应的句子中的出现次数与这个句子中所有的词出现的次数之和的比值。也就是，对于句子中的词i和句子j，TF计算公式如下

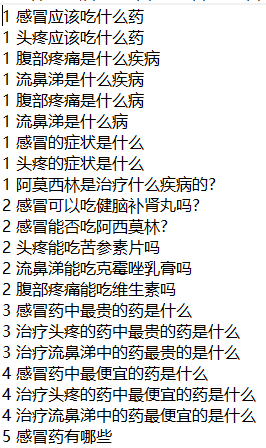
（4·2）

就是IDF是逆文本频率指数。对于词i，N为句子总数，Ni为含有词i的句子总数，IDF计算公式如下：

(4·3)

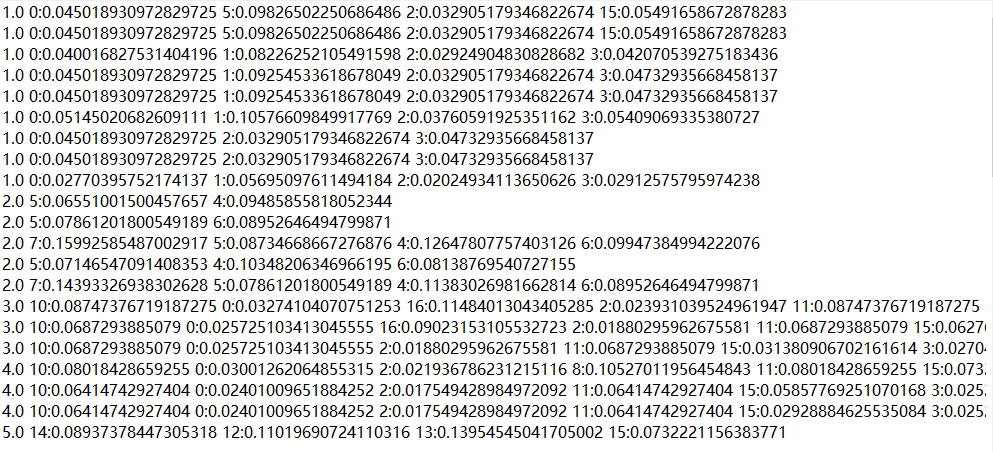
TF-IDF的值便由TF与IDF相乘而获得。

问题数据样例如下：



**图3-2 问题样例数据**

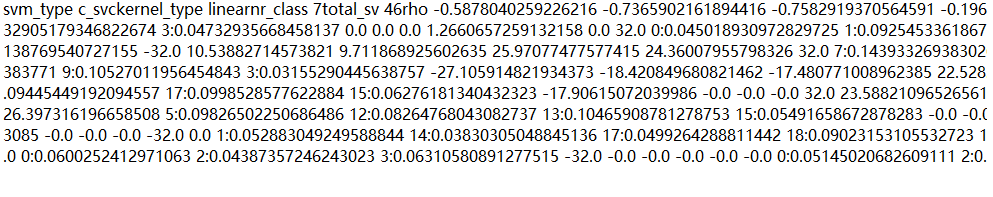
得到的问题的TF-IDF词向量如下：



**图3-3 问题样例数据的TF-IDF表示的向量**

### 3.3.3 SVM分类器

SVM分类器的主要思想是，对于已经给出的，分好类别的一系列样本点，寻找超平面，使得距离这个超平面最近的点到这个超平面的距离最大，当超平面确定完成，分类器也完成了的训练。而当给出的数据是不可分时，SVM的核函数会帮助解决这一问题，通过将数据映射到高维空间，原始空间中的线性不可分问题就得以解决。也就是说，这时候的SVM是现在低维空间下完成计算，然后在高维空间中找到超平面。常用的核函数有多项式核函数，RBF核函数等。最终训练完成的模型如下：



**图3-4 训练完成的SVM模型数据**

# 第4章 尚需完成的工作

在未来的一个月内，将完成以下工作：

（1）使用深度学习方法，对于用户的问句进行分析，对其进行，命名实体识别并且进行语义相似度分析。

（2）构建语义依存树，找到问句之间各个部分的相应关系。

（3）构建查询图并且使用最短路径算法找到问题的答案

# 第5章 存在的问题及解决的办法

问题1：数据集的数据量有限。

解决：与深度学习算法不同，采用的SVM模型，对于小规模数据有着良好的效果。

# 第6章 具体时间安排

**表6·1 时间安排表**

|  |  |  |
| --- | --- | --- |
|  | 设计（论文）各阶段名称 | 起 止 日 期 |
| 1 | 对国内外研究进展进行调研 | 2019.2.10 — 2019.3.15 |
| 2 | 学习知识图谱，python爬虫相关知识 | 2019.3.15 — 2019.3.20 |
| 3 | 学习SVM模型等方法，设计系统，实现部分功能 | 2019.3.20 — 2019.4.15 |
| 4 | 完成所有功能并完善系统 | 2019.4.15 — 2019.4.30 |
| 5 | 撰写论文和答辩 | 2019.5.1 — 2019.6.1 |