

**创新实验课程报告**

专 业 计算机科学与技术

学　　 号 1150310609

小 组 成 员 黄道龙（1150310613）

王陈阳（1150310609）

# 任务简介

## 1.1课题背景及研究的目的意义

传统的问答系统是人工智能（AI：Artificial Intelligence）领域的一个应用，通常局限于一个非常狭窄专门的领域，基本上是由人工编制的知识库加上一个自然语言接口而成。由于领域狭窄，词汇总量很有限，其语言和语用的歧义问题可以得到有效的控制。问题是可以预测的，甚至是封闭的集合，合成相应的答案自然有律可循。

在信息爆炸的今天，信息获取非常困难，这就对于信息检索有了更高的要求，为了满足人们对于快速、准确地获取信息的需求，问答系统的研究应运而生问答系统作为信息检索的一种高级形式，它能用准确、简洁的自然语言回答用户用自然语言提出的问题。

能够接受用户以自然语言形式的提问，并能从大量的异构数据中查找或推断出用户问题答案的信息检索系统称为问答系统(question answering，QA)。

从某种意义上说，问答系统是集知识表示、信息检索、自然语言处理和智能推理等技术与一身的新一代搜索引擎。问答系统与传统的信息检索系统在很多方面都有所不同，主要区别可参见下表：

|  |  |  |
| --- | --- | --- |
| **比较方面** | **问答系统** | **传统信息检索系统** |
| 系统的输入 | 自然语言提问 | 关键词组合 |
| 系统的输出 | 准确的答案 | 相关文档的列表 |
| 所属的领域 | 涉及NLP和IR两个领域 | IR领域 |
| 信息确定性 | 用户信息需求相对明确 | 用户信息需求相对模糊 |

问答系统也可以划分为很多种类型，如果根据系统的应用目的和获取问题答案所依据的数据，可以将问答系统划分为基于固定语料库的问答系统、网络问答系统和单文本问答系统。

基于固定语料库的问答系统的问题答案是从预先建立的大规模真实文本语料库中进行查找。

网络问答系统是从互联网中查找问题的答案，所以可以认为它基本能够涵盖所有问题的答案。不适合评价各种问答技术的优劣。

单文本问答系统也可以称为阅读理解式的问答系统，它是一篇给定的文章中查找问题的答案，要求系统在“阅读”完一篇文章后，根据对文章的“理解”给出用户提问的答案。这种系统仅在一篇文章中查找针对提问的答案，数据冗余性不高，所以要求的技术相对复杂。

基于常见问题(frequently asked questions,FAQ)的问答系统简称为FAQ问答系统，其典型用途是对企业产品或专业知识问题的解答。FAQ数据集的内容表现形式是提问和相应答案的问答对。这种类型的数据质量很高，答案基本上都比较准确，不存在垃圾等信息，但不足之处是数据规模往往比较小。在利用FAQ数据的过程中，通常有两种方式：1）从历史FAQ数据集中检索出与用户提问想死的问题，并返回该想死问题对应的答案；2）直接从FAQ数据集中检索答案。

本次研修主要探讨的是单文本问答系统，即阅读理解式的问答系统。

## 1.2相关工作

1.2.1问答系统构成模块

在目前情况下，一个自动问答系统通常由提问处理模块、检索模块和答案抽取模块三部分组成。

提问处理模块主要负责对用户的提问进行处理，包括：生成查询关键词(提问关键词、扩展关键词等)、确定提问答案类型(人称、地点、时间、数字等)以及提问的句法、语义分析，等等。

检索模块主要根据提问处理模块生成的查询关键词，使用某种检索方式，检索与提问相关的信息。该模块返回的信息可以使段落，也可以使句群或者句子。

答案抽取模块则利用相关的分析和推理机制从检索出的相关段落、句群或句子中抽取出与提问答案类型一致的实体，根据某种原则对候选答案进行排序，把概率最大的候选答案返回给用户。

* + 1. 词袋模型的调查

2013年，Google开源了一款用于词向量计算的工具——word2vec，引起了工业界和学术界的关注。首先，word2vec可以在百万数量级的词典和上亿的数据集上进行高效地训练；其次，该工具得到的训练结果——词向量（word embedding），可以很好地度量词与词之间的相似性。随着深度学习（Deep Learning）在自然语言处理中应用的普及，很多人误以为word2vec是一种深度学习算法。其实word2vec算法的背后是一个浅层神经网络。另外需要强调的一点是，word2vec是一个计算word vector的开源工具。当我们在说word2vec算法或模型的时候，其实指的是其背后用于计算word vector的CBoW模型和Skip-gram模型。

2003年，Bengio等人发表了一篇开创性的文章：A neural probabilistic language model[3]。在这篇文章里，他们总结出了一套用神经网络建立统计语言模型的框架（Neural Network Language Model，以下简称NNLM），并首次提出了word embedding的概念（虽然没有叫这个名字），从而奠定了包括word2vec在内后续研究word representation learning的基础。

NNLM模型的基本思想可以概括如下：1、假定词表中的每一个word都对应着一个连续的特征向量；2、假定一个连续平滑的概率模型，输入一段词向量的序列，可以输出这段序列的联合概率；3、同时学习词向量的权重和概率模型里的参数。

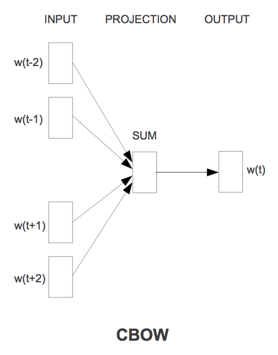
首先，我们对原始的NNLM模型做如下改造：

1、移除前向反馈神经网络中非线性的hidden layer，直接将中间层的embedding layer与输出层的softmax layer连接；

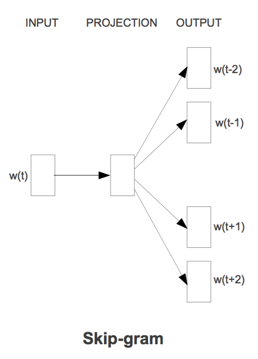
2、忽略上下文环境的序列信息：输入的所有词向量均汇总到同一个embedding layer；

3、将future words纳入上下文环境

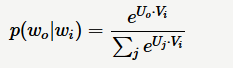
得到的模型称之为CBoW模型（Continuous Bag-of-Words Model），也是word2vec算法的第一个模型：



从数学上看，CBoW模型等价于一个词袋模型的向量乘以一个embedding矩阵，从而得到一个连续的embedding向量。这也是CBoW模型名称的由来。CBoW模型依然是从context对target word的预测中学习到词向量的表达。



这个模型被称为Skip-gram模型（名称源于该模型在训练时会对上下文环境里的word进行采样）。如果将Skip-gram模型的前向计算过程写成数学形式，我们得到：



其中，Vi是embedding层矩阵里的列向量，也被称为wi的input vector。Uj是softmax层矩阵里的行向量，也被称为wj的output vector。

因此，Skip-gram模型的本质是计算输入word的input vector与目标word的output vector之间的余弦相似度，并进行softmax归一化。

* + 1. 句子信息处理的方法

基于哈工大pyltp工具的句子信息处理（使用ltp模型中各个部分的模型进行分析

1. 分词

使用 pyltp 进行分词示例如下

# -\*- coding: utf-8 -\*-

import os

LTP\_DATA\_DIR = '/path/to/your/ltp\_data' # ltp模型目录的路径

cws\_model\_path = os.path.join(LTP\_DATA\_DIR, 'cws.model') # 分词模型路径，模型名称为`cws.model`

from pyltp import Segmentor

segmentor = Segmentor() # 初始化实例

segmentor.load(cws\_model\_path) # 加载模型

words = segmentor.segment('元芳你怎么看') # 分词

print '\t'.join(words)

segmentor.release() # 释放模型

结果如下：

元芳 你 怎么 看

1. 词性标注

使用 pyltp 进行词性标注示例如下

# -\*- coding: utf-8 -\*-

import os

LTP\_DATA\_DIR = '/path/to/your/ltp\_data' # ltp模型目录的路径

pos\_model\_path = os.path.join(LTP\_DATA\_DIR, 'pos.model') # 词性标注模型路径，模型名称为`pos.model`

from pyltp import Postagger

postagger = Postagger() # 初始化实例

postagger.load(pos\_model\_path) # 加载模型

words = ['元芳', '你', '怎么', '看'] # 分词结果

postags = postagger.postag(words) # 词性标注

print '\t'.join(postags)

postagger.release() # 释放模型

结果如下

nh r r v

1. 依存句法分析

使用 pyltp 进行依存句法分析示例如下

# -\*- coding: utf-8 -\*-

import os

LTP\_DATA\_DIR = '/path/to/your/ltp\_data' # ltp模型目录的路径

par\_model\_path = os.path.join(LTP\_DATA\_DIR, 'parser.model') # 依存句法分析模型路径，模型名称为`parser.model`

from pyltp import Parser

parser = Parser() # 初始化实例

parser.load(par\_model\_path) # 加载模型

words = ['元芳', '你', '怎么', '看']

postags = ['nh', 'r', 'r', 'v']

arcs = parser.parse(words, postags) # 句法分析

print "\t".join("%d:%s" % (arc.head, arc.relation) for arc in arcs)

parser.release() # 释放模型

其中，words 和 postags 分别为分词和词性标注的结果。同样支持Python原生的list类型。

结果如下

4:SBV 4:SBV 4:ADV 0:HED

arc.head 表示依存弧的父节点词的索引。ROOT节点的索引是0，第一个词开始的索引依次为1、2、3…

arc.relation 表示依存弧的关系。

arc.head 表示依存弧的父节点词的索引，arc.relation 表示依存弧的关系

## 1.3主要研究内容

根据问答系统在各个技术模块中所采用的不同方法，问答技术大致分为四种类型：基于检索的问答技术、基于模式匹配的问答技术、基于自然语言理解的问答技术和基于统计翻译模型的问答技术。

基于检索的问答技术就是利用检索算法直接搜索问题的答案，候选答案的排序是这类技术的核心，排序的依据通常是提问处理模块生成的查询关键词。由于不同类别的关键词对排序的贡献不同，算法一般把查询关键词分类几类：1）普通关键词：即从提问中直接抽取的关键词；2）扩展关键词：从WordNet或其他词汇知识库或Web中扩展的关键词；3）基本名词短语(base NP)；4）引用词：通常是引号中的词；5）其他关键词，等等。

对于一些固定模式的提问，可采用基于模式匹配的问题求解技术，获得尽可能多的答案表述模式是这种方法的关键技术。

基于自然语言处理的问答技术可以对提问和答案文本进行一定程度句法分析和语义分析，并实现推理。

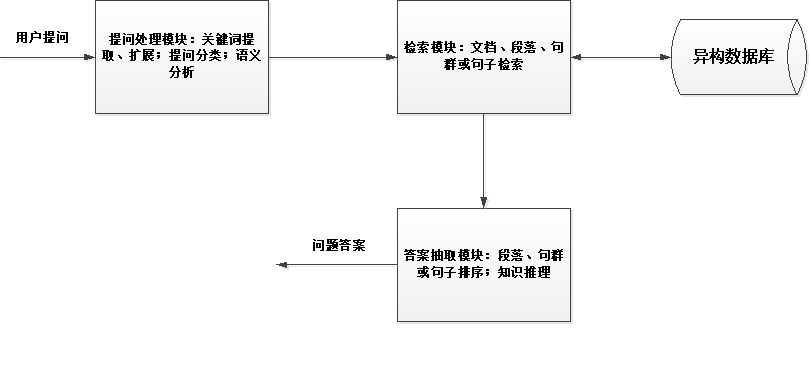
基于统计翻译模型的问答技术把提问句看作是答案句在同一语言内的一种翻译形式，答案句子中与提问句子中的疑问词对应的词即是该问句要找的答案。步骤：首先对检索句进行分析，保留句子中的提问词，然后，搜索包含答案的候选句子，对候选句子进行分析，最后，使用统计机器翻译模型(主要是对其技术)抽取提问的答案。

优缺点比较：基于检索的问答技术相对简单、容易实现，以基于关键词的检索技术为重点，只考虑离散的词，不考虑词语词之间的相互关系。无法从句法关系和语义关系的角度解释系统给出的答案，也无法回答需要推理的提问;基于模式匹配的问答技术对于某些类型的提问(如定义、出生日期等)具有良好的性能，但是不能涵盖所有提问的答案模式，也不能表达长距离和复杂关系的模式，同样也无法实现推理；基于自然语言理解的问答技术，可以对提问和答案文本进行一定程度的句法分析和语义分析，并实现推理，技术还不成熟，除一些浅层的技术(汉语分词、词性标注、明明实体识别、基本短语识别等)以外，其他技术还没有达到实用的程度；基于统计翻译模型的问答技术，很大程度上依赖于训练预料的规模和质量，而对于开放域的问答系统，这种大规模训练预料的获取非常困难，而且，目前在统计翻译中，词或短语对其的准确率本身就没有达到一个很高的水准。

# 模型

2.1基本模型

通常的问答系统模型设计如下：

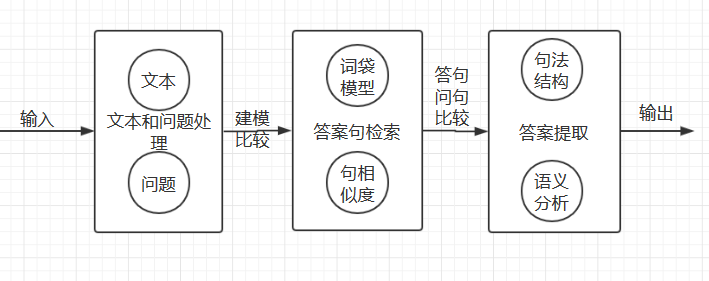


包含三个基本的模块：提问处理模块，检索模块，答案抽取模块，各个模块之间的关系密切，且各模块分别采用不同的技术加以处理，以期达到更好的效果。提问处理模块通过关键词提取、扩展，提问分类，语义分析完成对需求的分析，为下一步检索模块提供检索信息，检索模块通过上一模块的输出，选择适当的检索方式，从异构数据库中获取信息，传递给答案抽取模块，答案抽取模块通过对信息的排序和推理，选出最适合的答案输出。

2.2中文阅读理解模型

中文阅读理解是通过在一篇有限篇幅的中文文章中进行检索，回答用户提出的问题，可以使用到的只有一篇文本和问题，较之其它问答系统，它仅在一篇文章中查找针对提问的答案，数据冗余性不高，技术分析上更为复杂 ，用到的中文信息处理方法也更多。

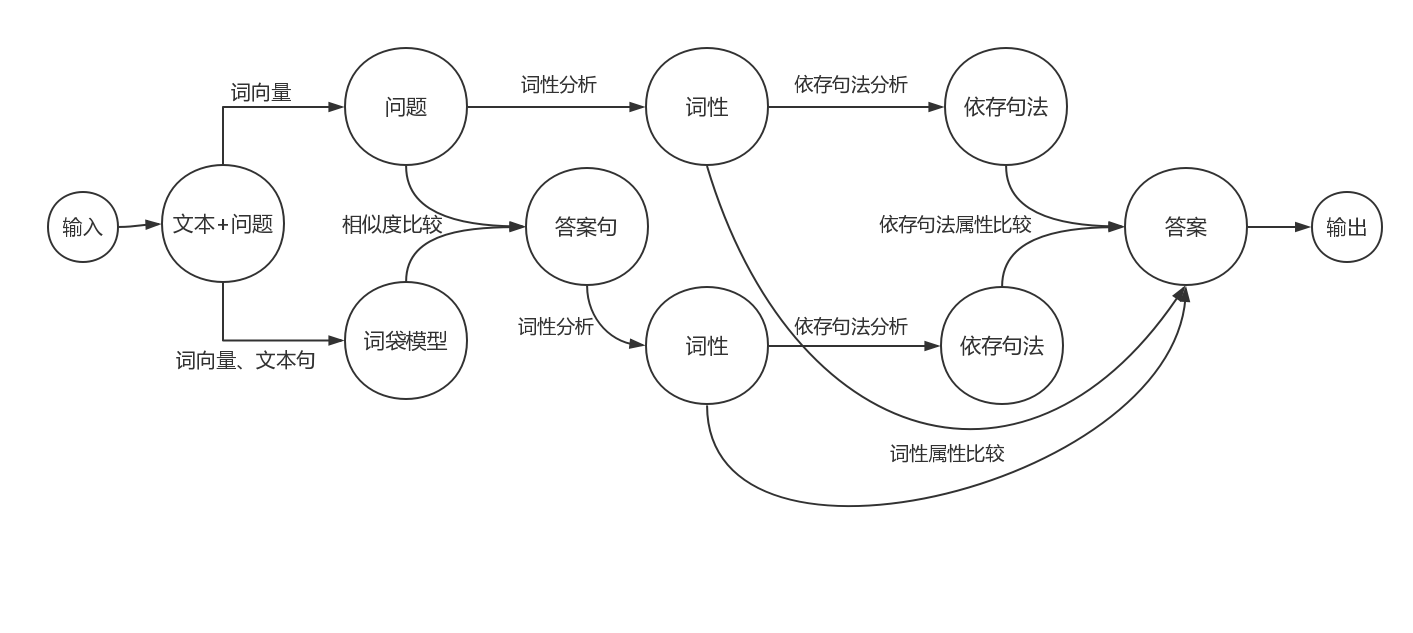
基于词袋模型的中文阅读理解系统设计如下：



通过输入文本和问题，在第一个模块中对文本和问题进行预处理，如繁简转换，中文分词，标点符号去除等一系列格式化的操作，使得文本和问题可以转化为一个个词，便于构建词向量，进而构建词袋模型，完成信息检索库的建立，在词袋模型建立完成后，第二个模块中会将问句与词袋模型中每一个句子计算句相似度，进而选出最有可能是答句的几个句子，最后，在答案提取模块，通过对上一模块中最有可能是答句的几个句子与问句做句法结构分析和语义分析，最终确定哪一个词是答案，并输出。

# 实验

## 3.1实验设计



1. 输入文本+问题：预处理（分词，去除标点符号，繁简转化等）
2. 文本的句子构建词向量，进而构建词袋模型；问题构建词向量；
3. 问题与词袋模型中句子进行相似度比较，获取答案句；
4. 对问题和答案句进行词性分析，通过词性比较获取答案（候选方式）；

⑤由上一步中词性分析结果进行依存句法分析，通过依存句法属性比较获取答案（主要方式）。

## 3.2实验结果及分析

3.2.1实验结果

举出部分例子作为参考：

Standard\_answer：

* <qid\_0> ||| 门槛
* <qid\_1> ||| 发高烧
* <qid\_2> ||| 厨师
* <qid\_3> ||| 皮斯托纳人
* <qid\_4> ||| 魔法棒
* <qid\_5> ||| 山羊
* <qid\_6> ||| 瑶瑶
* <qid\_7> ||| 灯笼
* <qid\_8> ||| 玩耍
* <qid\_9> ||| 鱼
* <qid\_10> ||| 服装店
* <qid\_11> ||| 老鼠
* <qid\_12> ||| 孙小圣
* <qid\_13> ||| 河马
* <qid\_14> ||| 小姑娘

Train\_answer：

* <qid\_0> ||| 小鸟
* <qid\_1> ||| 河马
* <qid\_2> ||| 厨师
* <qid\_3> ||| 狄
* <qid\_4> ||| 魔法
* <qid\_5> ||| 人
* <qid\_6> ||| 瑶瑶
* <qid\_7> ||| 灯笼
* <qid\_8> ||| 玩耍
* <qid\_9> ||| 鱼
* <qid\_10> ||| 一
* <qid\_11> ||| 老鼠
* <qid\_12> ||| 孙小圣
* <qid\_13> ||| 魔法
* <qid\_14> ||| 小姑娘

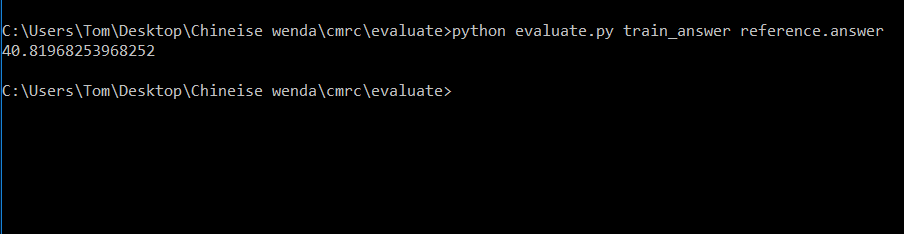
评价指标是F1值，其计算方式为：

precision = 1.0 \* num\_same / len(pred\_tokens)

recall = 1.0 \* num\_same / len(ref\_tokens)

F1 = (2 \* precision \* recall) / (precision + recall)

其中 num\_same 为预测答案和正确答案共有字符数



# 结论及收获

4.1结论

基于词袋模型的中文阅读理解系统构建是可行的，对于给定的文本和问题，能够对一些问题给出正确的回答，正确率有一定的保证。

4.2遇到的问题

1. 选出答案句

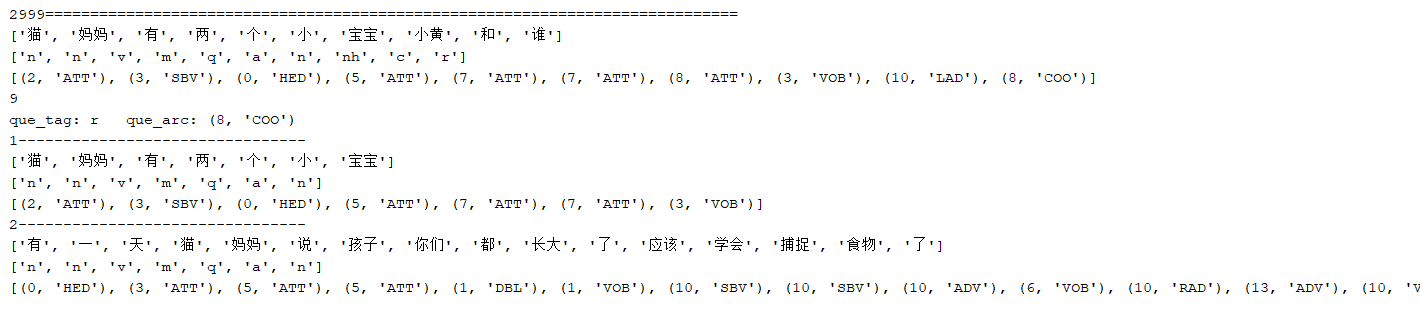
1、选不到含有正确的句子（前三句）

原因： 分词存在小部分错误；

繁简转化；

分句的存在使得相似度下降；

代词的使用等

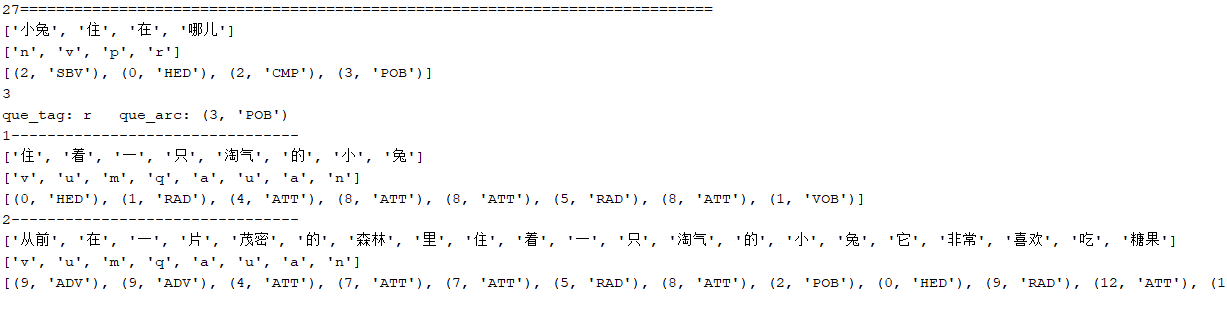


2、含有正确答案的句子相似度排不到第一位

原因： 分句的存在；

代词、组词等常见词的词频高，特殊词（与答案句关联性高的词）词频低，

计算相似度时权值没有区分等；



1. 选出答案词

1、词性分析

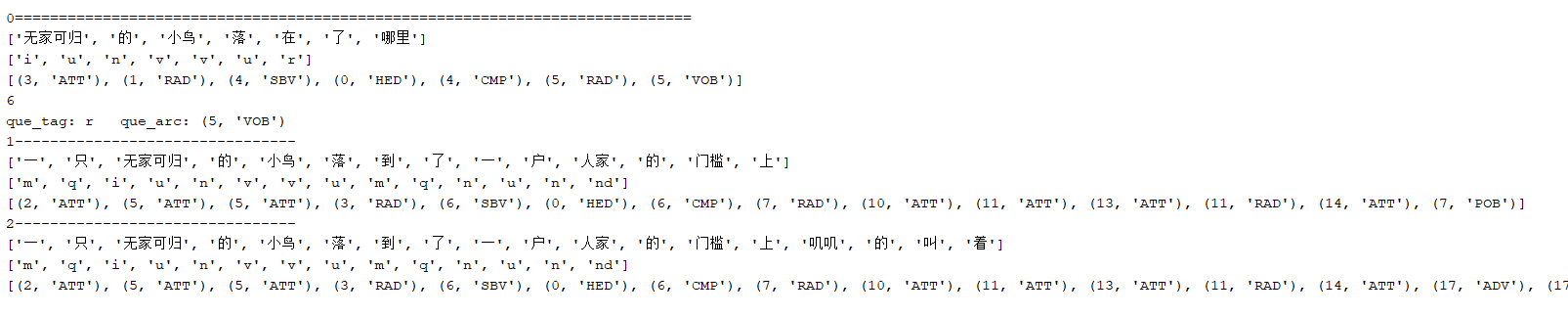
原因： 疑问词词性（问答关系）与答案句中词性（同类词性不只一个）匹配出现选择；

分词错误等也有影响

2、依存句法分析

原因： 问句与答案句的依存句法分析不同，难以匹配到相同的词；

分词错误等



4.3改进分析

1. 选出答案句

1、选不到含有正确的句子（前三句）

原因： 分词存在小部分错误----使用自定义词库进行个性化分词；

繁简转化；

分句的存在使得相似度下降----分句转化为单句；

代词的使用----代词的名词代换等

2、含有正确答案的句子相似度排不到第一位

原因： 分句的存在----修改分句为单句；

代词、组词等常见词的词频高，特殊词（与答案句关联性高的词）词频低，

计算相似度时，权值没有区分等----

对低频特殊词的识别，提高特殊词在模型中的权重；

1. 选出答案词

1、词性分析

原因： 疑问词词性（问答关系）与答案句中词性（同类词性不只一个）匹配出现选择----依存句法辅助解决；

分词错误等也有影响

2、依存句法分析

原因： 问句与答案句的依存句法分析不同，难以匹配到相同的词----利用整个句子的依存句法进行分析；

分词错误等

4．4收获

通过本次创新研修，使我对于中文阅读理解有了深刻的认识，对于典型的词袋模型的运用有了了解，也在学习和调查中对于中文信息处理有了更深的理解，对典型分词，词性标注，依存句法分析的工具工大pyltp的使用更加清楚了。在整个系统设计和实现过程中，对于一个问题完整的调查，研究，设计，以及实现解决方案的过程有了一次不可多得的体会。

# 组内分工

5. 组内分工

在本次研修课程中，我们小组成员有黄道龙，王陈阳，李清尘三位同学。我们的工作主要包括了三次课堂报告，资料查找及系统构想，系统代码实现等

①课堂报告

三次课堂报告的前两次的主要内容由黄道龙编写，王陈阳和李清尘同学进行校对和美工工作的处理。第三次报告由王陈阳编写，黄道龙、李清尘提供材料和校对。

三次课堂报告演讲均由王陈阳同学上台演讲，余下两位同学进行老师提问的答案补充。

②资料查找及系统构想

由于小组内每一位同学一开始对于该系统的实现都有不同的见解，每个人都对自己的实现构想进行了相关的资料的查找。最终，我们经过讨论得出了现有的系统模型框架，并且对相关的资料进行了汇总。

③系统代码实现

最终的测试版本的系统代码中，黄道龙主要完成了问题分析模块和答案抽取模块，并参与了文本句子处理模块的编写和调式工作；王陈阳主要完成了文本句子处理模块，并参与了答案抽取模块的算法讨论和调式工作；由于李清尘没有接触过python语言，他参与讨论了这三个模块的实现工作，并且他使用java语言实现了模块的部分功能。