



## 本科毕业设计（论文）

### 基于关系抽取的答案自动提取系统的设计及实现

Design and Implementation of the Automatic  
Answer Extraction Based on Relation Extraction

学    院： 软件学院

专    业： 软件工程

学生姓名： \*\*\*\*\*

学    号： \*\*\*\*\*

指导教师： \*\*\*\*\*

北京交通大学

2016 年 5 月

## 学士论文版权使用授权书

本学士论文作者完全了解北京交通大学有关保留、使用学士论文的规定。特授权北京交通大学可以将学士论文的全部或部分内容编入有关数据库进行检索，提供阅览服务，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。

（保密的学位论文在解密后适用本授权说明）

学位论文作者签名：

指导教师签名：

签字日期：      年    月    日

签字日期：      年    月    日

## 中文摘要

传统的搜索引擎利用查询扩展或关键词进行检索并返回大量网页，用户很难精准定位自己需要的信息。问答系统让用户以自然语言的形式提问，并返回准确简短的答案，答案自动提取是从已检索到的网页中找出答案并返回给用户，其准确性对问答系统本身的准确性起着关键作用。关系抽取是从文本中抽取出特定的实体对，形成针对某一属性或关系的结构化数据。本文是用关系抽取的方式从 Web 非结构化文本中自动提取出新的关系三元组，实现问答系统答案自动提取。

本文对比现有关系抽取的方法，根据百度知识图谱现有情况，选择了 Distant Supervision 方法；针对 Web 非结构化文本设计了基于深层语义关系的词法特征、句法特征和整句特征，并对文本分类方法进行了分析和评估；由于中文自然语言处理存在的问题，本文在命名实体识别等方面进行了优化；鉴于目前自动问答系统在非客观问题上的缺失，本文使用 LexRank 算法从 Web 数据中自动提取答案摘要，提高了答案自动提取系统的完整性和可扩展性。本文使用百度网页库实际数据实现了命名实体答案自动提取和自动文摘算法，通过众包平台评估了数据准确率，人物关系领域抽取的答案准确率的平均值为 85%。

**关键词：**问答系统；答案提取；关系抽取；命名实体识别；自动文摘

## ABSTRACT

The traditional search engines use query expansion or keywords to achieve the retrieval of information. Therefore, it is difficult for users to find the information they need quickly. QA system allow users to ask questions in natural language, and provides a precise and concise answer. Answer extraction can find out the answer from the web pages and return to the users. Its accuracy plays a key role in QA system. Relation extraction extract the specific entity pair from texts, and form the structural data for a particular attribute or relationship. The paper use Relation Extraction to extract the new tri-tuple relationship from the Web unstructured text to realize the automatic extraction of Answering System.

The paper compares the existing methods of relation extraction, and choose Distant Supervision according to the situation of Baidu Knowledge Graph; Considering the unstructured data of the web, the paper designs the syntactic features, lexical features and sentence features based on deep semantic relations, analysis and estimate the methods of text classification; Because of the problems existing in Chinese Natural Language Processing, this paper optimizes some methods like Named Entity Recognition; The current QA system lacks of processing the objective problems, the paper designs a method to extraction answer from Web using Automatic Text Summarization, which can improve the integrity and scalability of the Answer extraction system. This paper uses the Baidu web database to achieve the named entity's answer automatic extraction and automatic summarization algorithm, and assess the precision rates through the crowd-sourced platform. The average value of the answer accuracy rate in the field of relationship is 85%.

**KEYWORDS:** Automatic Question Answering; Answer Extraction; Relation Extraction; Named Entity Relation Extraction; Automatic Summarization

## 目 录

中文摘要 .....	II
ABSTRACT .....	III
目 录 .....	IV
1 绪论 .....	1
1.1 课题来源 .....	1
1.2 相关研究综述 .....	2
1.2.1 关系抽取研究现状 .....	2
1.2.2 自动文本摘要研究现状 .....	3
1.2.3 中文问答系统答案提取研究现状 .....	4
1.3 论文的主要研究内容 .....	5
1.4 论文的组织结构 .....	6
2 答案自动提取系统的技术优选 .....	8
2.1 知识图谱概要 .....	8
2.2 关系抽取技术对比和优选 .....	9
2.2.1 Bootstrapping 技术 .....	10
2.2.2 Distant Supervision 技术 .....	11
2.3 关键技术优选 .....	11
2.4 本文研究的重难点 .....	13
2.5 本章小结 .....	15
3 基于关系抽取的答案自动提取的算法设计 .....	16
3.1 WEB 语料获取和处理 .....	16
3.2 网页文本处理和优化 .....	18
3.3 句子特征 .....	21
3.3.1 词法特征 .....	22
3.3.2 句法特征 .....	22
3.3.3 整句特征 .....	23
3.3.4 特征合并 .....	24
3.4 命名实体类型答案提取 .....	24
3.5 本章小结 .....	27
4 答案自动提取系统的实现 .....	28
4.1 系统框架 .....	28
4.1.1 整体架构 .....	28

4.1.2	功能模块分解 .....	29
4.2	模块详细设计 .....	31
4.2.1	检索模块 .....	31
4.2.2	基础文本处理模块 .....	32
4.2.3	基础算法模块 .....	34
4.2.4	命名实体答案提取模块 .....	36
4.2.5	自动文本摘要模块 .....	38
4.3	本章小结 .....	40
5	实验方案及结果验证 .....	41
5.1	实验方案 .....	41
5.1.1	平台基础 .....	41
5.1.2	数据来源 .....	41
5.1.3	效果评估 .....	41
5.2	文本分类 .....	42
5.2.1	数据准备 .....	42
5.2.2	文本类别判定 .....	43
5.2.3	实验结果分析 .....	45
5.3	命名实体答案提取 .....	49
5.3.1	置信度校验 .....	49
5.3.2	实验结果分析 .....	50
5.4	自动文本摘要 .....	51
5.4.1	句子相似度判断 .....	52
5.4.2	句子权重计算 .....	53
5.4.3	文摘可读性加工 .....	53
5.4.4	实验结果分析 .....	53
5.5	本章小结 .....	54
6	总结与展望 .....	55
6.1	全文总结 .....	55
6.2	展望 .....	55
	参考文献 .....	57
	致 谢 .....	59
	附录 A 英文原文 .....	60
	附录 B 中文翻译 .....	70

## 1 绪论

本章介绍了本课题的来源和一些相关背景，详细介绍了中文自动问答系统答案提取、关系抽取和自动文本摘要的研究现状，以及本文的主要研究内容。

### 1.1 课题来源

当前互联网的普及为人们提供了丰富的信息资源，我们能通过搜索引擎获取自己想要的各种信息。传统的搜索引擎是基于文本关键词的搜索，当用户输入问题后，搜索引擎会返回它查询到的与问题关键词相关的所有网页。这种搜索方式的缺点是比较机械化，并没有深入地挖掘用户查询的真正地搜索需求，所使用的关键词的逻辑组合也并不能一定能够表述用户真正的检索请求，因此返回的网页或文档往往比较多，用户很难快速找到自己需要的信息。

与基于文本关键字的传统搜索引擎并不同，问答系统让用户以自然语言的形式提问，并返回准确简短的答案，而不是将大量的相关文本或网页返回给用户<sup>[1]</sup>。为了满足用户真正的检索需求，方便用户快速地找到所需的答案，国内外很多公司在进行精准自动问答方面的尝试，如微软小冰、百度度秘等，它们综合运用了信息检索、自然语言处理、语义分析、机器学习等技术，系统将会自动理解和分析用户的提问，直接返回用户想要的答案<sup>[2]</sup>。比如：当用户提问“刘德华的妻子是谁？”，问答系统可以直接给出答案“朱丽倩”，而不是大量的网页文本。

问答系统一般主要包括 3 个模块：问题分析、信息检索和答案提取<sup>[3]</sup>。问题分析模块主要是分析用户提出的问题，理解用户真正的搜索请求，一般主要包括问题领域划分、问题中的关键词提取和关键词的扩展，通过对问题的归一化（多表达归一，最短关键词描述，简化问题）、形式化（转化为结构化查询语言，让机器可以理解和计算）和改写技术（根据上下文语境改写句子，输出替换片段，更好的满足检索要求和扩大相关资源召回），全面理解用户意图；问题分析后将得到的关键词查询集合提交到信息检索模块，通过检索获取与问题相关的网页及文本；答案提取模块是从已检索到的网页及文本中找出答案（一句话或者是一个简单的实体），并返回给用户<sup>[4]</sup>。问答系统答案提取的准确性对问答系统本身的准确性起着关键作用，本文使用关系抽取的方法解决答案为简单实体类型的领域问题，对于客观答案（或者答案很长，需要整理信息）使用摘要算法提取简要答案提供给用户。

## 1.2 相关研究综述

本文主要涉及关系抽取、自动文本摘要和中文问答系统答案提取三个方面的工作，下面将介绍这三个方面的研究现状。

### 1.2.1 关系抽取研究现状

关系抽取研究在 MUC 评测会议和 ACE 评测会议的引导和推动下，许多先进的信息抽取技术被提出来，并在会议提供的平台上测试。总的来说，这些方法主要分为两类：基于模式匹配的方法和基于机器学习的方法<sup>[5]</sup>。基于模式匹配的方法需要融合各个领域知识和语言学的知识，通过人工或使用统计编写不同领域的规则集合，构造出特定句子模式，利用模式匹配的方式找到新的关系实例<sup>[5]</sup>。基于机器学习的方法将关系抽取的属性或关系识别问题转化为分类问题，通过选取有句子代表性的特征，利用不同的机器学习算法训练出不同领域分类器，最终通过训练出的分类器识别实体对之间的属性或关系<sup>[5]</sup>。

总体来说，基于模式匹配的关系抽取方法同使用编写规则集合，再进行匹配来提取答案的方法一样，存在泛化难度大，在实际使用过程中容易因为覆盖范围不够，导致召回率降低的问题。而基于机器学习的方法主要的解决方案是使用半监督或无监督技术，目前的代表性技术是 Bootstrapping 技术和 Distant Supervision 技术。

Bootstrapping 技术从少量的种子实例出发自动抽取新的实例<sup>[6]</sup>，而 Distant Supervision 技术则充分利用现有的大规模知识库（如 Freebase，谷歌或百度的知识图谱等等），使用非直接监督实例来构建大规模信息抽取系统<sup>[7]</sup>。

Bootstrapping 方法在迭代过程中容易引入噪音实例和噪音模板出现语义漂移现象，见图 1 语义漂移现象。根据 [Krause et al., 2012] 的研究，人物之间四种关系模板的交叉程度很大<sup>[8]</sup>，虽然之后很多学者，如 [McIntosh et al., 09] 提出同时扩展多个互斥类别，一个实体对只能属于一个类别<sup>[9]</sup>、[Carlson et al., 10] 提出建模不同抽取关系之间的约束，寻找最大化满足这些约束的抽取结果<sup>[10]</sup>、[Shi et al. 14] 提出引入负实例来限制语义漂移，从不同角度提出了相应的解决<sup>[11]</sup>，但这些方法在一定程度上缓解了语义漂移现

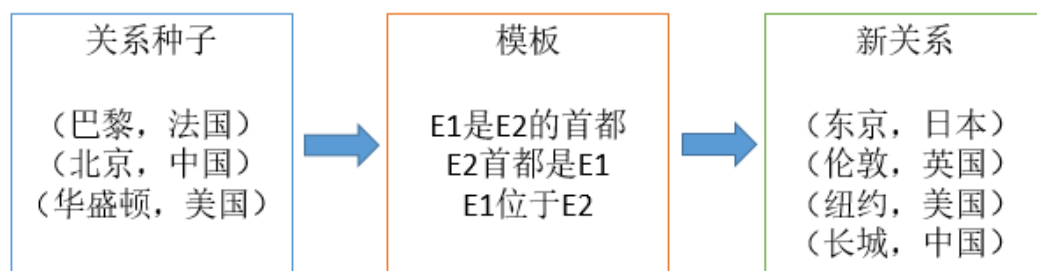


图 1 语义漂移现象



象，其结果还是准确率仍不能达到工业项目上线目标。

**Distant Supervision** 由于其使用未标注数据容易大量噪音训练实例，严重影响抽取性能，见图 2 噪音训练实例。[Takamatsu et al. ACL 12] 假设一个正确的训练实例会位于语义一致的区域，也就是其周边的实例应当都有相同一致的 Label，提出基于生成式模型的方法去除噪音数据[12]、[Surdeanu et al. EMNLP 12] 假设只要实体对的一个句子具有特定关系，那么该实体对也就具有该关系，提出关系实例 label 被建模为 hidden variable，使用 Factor Graph 来表示多个变量之间的关系[13]、[Riedel et al. NAACL 13] 提出基于协同过滤推荐的方法，将关系抽取任务建模为矩阵填空问题。

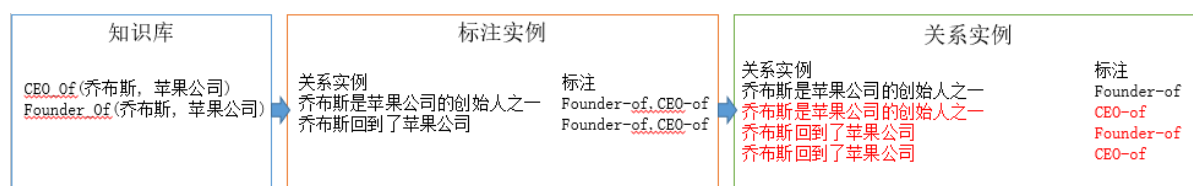


图 2 噪音训练实例

## 1.2.2 自动文本摘要研究现状

自动文本摘要主要分为单文档文摘和多文档文摘，现今单文档文摘主要方法包括：基于文本特征方法、基于词汇链的方法和基于图排序方法[14]。其根本原理是将文本视为句子的线性序列，将句子视为词的线性序列，通过计算词语权重来计算句子权重，再对按照权重对句子进行排序，选择权重高的句子作为文摘句[15]。

### 1. 基于文本特征方法

基于文本特征方法主要使用的特征包括词频、句子位置、文档标题等。文章中频繁出现的单词与文章主题有比较大的关联，因此可以根据各单词出现的频率给文中的句子打分，以得分最高的几个句子组成文章的摘要[14]。通过从句子位置特征入手，通过计算文章中段落首末句出现主题句的概率，选取得分最高的若干句子生成摘要。

### 2. 基于词汇链的方法

在文章中描述某个主题的文本块内，使用的词语应该是相关的，这些相关词语构成一条词汇链。所以，词汇链可以视作一个语言片段的标志性主题词语链，不同的词汇链对应了不同的语言片段。其计算句子权重的主要思路是先选取集合中的一个词语，形成的一个词汇链的第一个词语，计算该词语与集合中其他词语的相似度，当相似度大于一定阈值的时候，认为两个词语是在同一个词汇链中；如果相似度并未大于阈值，则将另一个词语作为新一条词汇链的第一个词语；重复上述过程，直到所有的词语都在一个词汇链中为止。再以每个词汇的权重、位置等信息为依据计算每个词汇链的权重，最后对

词汇链进行排序，选择大于某个特定阈值的词汇链，包含词汇链上词语的语句作为备选语句集合。

### 3. 基于图排序方法

把文章分解为若干单元（句子或段落等），每个单元对应一个图的顶点，单元间的关系作为边，最后通过图排序的算法（如 PageRank、Manifold Ranking 等）得出各顶点的得分，并在此基础上生成文本摘要<sup>[14]</sup>。

## 1.2.3 中文问答系统答案提取研究现状

在问答系统的答案提取方面，目前主要有基于句子相似度的方法、基于潜在语义的方法和基于模式匹配的方法。基于句子相似度的方法是通过计算句子与问题相似度来计算句子权重，从而得到候选答案集合<sup>[16]</sup>；基于潜在语义的方法是通过统计方法提取并量化同义词和多义词的潜在语义结构，从而提供准确性<sup>[16]</sup>；基于模式匹配的方法是分析同类问题的答案形式，人工或通过统计来定义不同领域问题的答案句子模式，再通过模式匹配的方式从海量文本中提取新的实体关系对作为答案<sup>[4]</sup>。

### 1. 基于句子相似度的答案提取

基于句子相似度的答案提取是通过计算用户的问题和网页文档中的各个句子的相似度，排序后选取相似度最大的句子作为答案。而两个句子的相似度计算可以使用关键词匹配方法，即通过计算目标问题和文档句子中拥有相同的关键词总数来确定二者的相似程度，使用这样的方法需要考虑共指消解、关系推断等问题。主要包括两个方面，一是实体重名，如“李晨的妻子是谁”，搜索结果为“李晨的老婆是他高中的女友叫赵琼，两人在一起十年了”，但此处李晨指的是主持人李晨而不是演员李晨，再比如“我刚买了两袋苹果”与“我刚买了一台苹果”，两个句子中的“苹果”表述的是不同意义，通过关键词匹配很难处理词的多义性问题；二是人们为了避免重复，习惯用代词、缩略语、简称来指代一些实体全称<sup>[17]</sup>，如“ICBC”，使用关键词匹配过程中可能会丢失与“ICBC”同义的“中国工商银行”或者“工商银行”等等。

### 2. 基于潜在语义分析的答案提取

基于潜在语义分析的答案提取的基本理论是文档中的词与词之间存在一些关系，通过构建同义词和多义词的词汇表来解决使用词匹配方式不可避免的共指消解问题，从而提高答案提取时的准确率。然而这种方式并没有分析句子真正的语义，如搜索“王菲的丈夫是谁”，有搜索结果有“王菲和李亚鹏离婚后，谁将成其第三任丈夫？”，此处表述含义为“李亚鹏”已经离婚，但单纯使用关键词或者同义词、多义词的词汇表并不能准确理解这一点，有可能得到错解。

### 3. 基于模式匹配的答案提取

基于模式匹配的答案提取是先通过人工处理得到包含答案的句子，并标注问题分类及答案，形成不同问题分类的问答训练语料。通过人工或统计的方法学习，提取候选答案句子的答案模式，再计算候选答案句子模式置信度和权重，并根据置信度和权重获得相应问题分类的答案句子模式。但是这种方式有一些明显问题，一方面这种方法对不同领域、不同实体属性或关系很难泛化，当需要移植到其他领域时，需要进行大量的人工标注工作，另一方面由于自然语言的多样性，同一个含义的表达方式并可以不同，在不同语境中，对同一实体属性或关系表述差异很大，人工标注过程很难覆盖全面，在实际使用过程中召回率不高。

## 1.3 论文的主要研究内容

目前中文自动问答系统答案提取部分已经在很多方面取得了一些研究成果，但在其中仍有一部分有待解决的问题，主要包括以下几个方面：

### 1. Web 非结构化文本的正文语料获取

目前自动问答系统答案提取处理的文本数据主要是经过处理后的有效文本，在信息时代，互联网作为知识贡献的渠道，存在大量可用于问答系统答案提取的文本资源。但由于互联网中大部分数据属于非结构化文本数据，增大了答案提取的难度，因此需要先对 Web 数据进行预处理，提取出正文部分。

### 2. 文本语义特征的选取

文本语义特征是答案自动提取系统中对文本处理的关键问题，目前大部分自动问答系统主要使用了一些浅层文本特征，例如分词结果、词语出现的频率、词性等。这些浅层的特征不依赖于复杂的语法分析，比较容易取得，但常常不能完整的表示文本的语义关系。

### 3. 中文自然语言处理中的局限性

由于汉语词语之间没有空格，有些词汇也不可分割，很多领域用语并不像其他语言一样存在可用标识来进行分词，词性标注、句子依存分析和命名实体识别也存在一些可改进的部分。

### 4. Web 中大量的无用数据影响

互联网中虽有大量可用的文本资源，但其中也包含了许多无用的数据，这些数据对答案提取的结果有着很大的影响，因而需要计算网页数据的置信度，在答案提取前去除这些无用的文本。

### 5. 非客观的问题的答案摘要提取

目前自动问答系统主要针对的问题属于简短的客观问题，对于答案属于长答案的领域问题缺少设计和实现。

针对以上问题，本文在学习和分析了现有的相关研究的基础上，提出了基于关系抽取的答案自动提取系统的解决方案，具体研究内容如下：

### **1. 基于标签密度法的 Web 文本正文语料提取**

由于从 Web 中获取的网页文本大多是半结构化或非结构化的，数据也往往杂乱无章，尤其是从一些论坛或置信度不高的网页中获取的数据，常常出现广告或其他无使用价值的文本（宣传站点、推荐其他内容），本文针对 Web 文本特点进行了调研，提出了相应的语料获取思路。

### **2. 基于深层文本语义结构的文本特征选取**

由于需要在答案提取前去除 Web 文本中大量无用的文本数据，本文在现有关系抽取的距离监督方法的基础上，通过实际数据验证和多次性能度量选取了新的深层文本语义结构作为文本特征。

### **3. 对中文人名和时间领域的自然语言处理进行优化**

由于汉语自身特点，通过自然语言处理后的数据中包含一些被错误处理的数据，本文就中文人名和时间两个领域提出了分词和命名实体识别方面的优化策略，提高了文本处理的质量。

### **4. 针对不同领域对 Web 文本分类**

由于 Web 文本质量参差不齐，在实际答案提取过程中，需要先处理获取到的 Web 文本。针对本文实际情况，使用新的文本特征，本文对多种机器学习算法进行评估，选择了最适宜本文的机器学习算法，并对实验结果进行了进一步的分析和验证。

### **5. 命名实体类答案自动提取**

在使用分类器对文本的类别进行判定之后，下一步便是从文本中抽取相应的答案。虽然使用分类器过滤掉了一部分明显不表述母亲关系的文本，但剩下的文本中仍然存在错误的可能。本文针对这种情况和实际网页数据，提出了备选答案可能性评分的方案和置信度校验方法。

### **6. 答案摘要提取的构建**

鉴于目前自动问答系统在非客观问题上的缺失，本文设计了从 Web 数据中自动提取答案摘要的方法，提高了答案自动提取系统的完整性和可扩展性。

## **1.4 论文的组织结构**

本文主要研究使用关系抽取的思路，在问答系统中实现答案自动提取的方法。全文

一共分为六个章节，各个章节的结构安排及主要内容如下：

第一章为绪论部分。本章介绍了本课题的来源和一些相关背景，然后详细介绍了问答系统答案提取、关系抽取和自动文本摘要的研究现状。

第二章为关系抽取与答案提取的关联分析部分。本章详细介绍了知识图谱的研究现状、其面临的瓶颈，探讨了关系抽取在知识图谱知识库构建和数据更新上运用的可能性，并提出了本文的主要思路和关键技术。

第三章为基于关系抽取的答案自动提取的关键技术部分。本章详细介绍使用关系抽取在问答系统答案提取中的关键方法和技术。

第四章为系统设计与实现部分。给出了本文实验系统的总体设计思路和系统的总体流程图结构，然后从自然文本处理、句子特征的提取、训练分类器、答案提取这四个模块详细分析系统的实现。

第五章为实验方案及结果验证部分。本章详细介绍了如何抽取对于命名实体类型的答案以及如何从可能的答案文档中抽取摘要作为答案以及最终的研究成果。

第六章总结了本文的研究工作，并提供了下一步研究的改进方法。

## 2 答案自动提取系统的技术优选

本章详细介绍了知识图谱的研究现状、其面临的瓶颈，探讨了关系抽取在知识图片知识库构建和数据更新上运用的可能性，并提出了本文研究的重难点。

### 2.1 知识图谱概要

知识图谱将杂乱的网页数据构建成一个结构化的实体，能够为用户提供更加有条理的实体及其属性或关系信息，顺着知识图谱甚至可以探索到更深入、完整和广泛的知识。要实现自动问答，搜索引擎不仅需要理解查询的问题中涉及到的命名实体及其属性或关系，还需要理解查询语句的语义信息。搜索引擎可以通过使用高效的图搜索，在知识图谱中查找与这些实体及属性或关系连接的子图，图搜索结果被进一步提交给图数据库并返回相应的答案给用户。

在构建问答系统知识图谱中基本表达元素为三元组（SPO-S 表示主语 (subject)-P 代表谓语 (predicate)-O 代表宾语 (object)），如“刘德华的妻子是朱丽倩”表达的实体关系三元组，S 为“刘德华”、P 为“妻子”、O 为“朱丽倩”；“刘德华的生日是 1967 年 9 月 27 日”表达的实体属性三元组，S 为“刘德华”、P 为“生日”、O 为“1967 年 9 月 27 日”。在问答系统答案提取的过程中是通过已知的 S 和 P 来抽取未知答案 O，如用户搜索“电视剧半路夫妻的演员有哪些？”，S 为“半路夫妻”、P 为“演员”，返回的结果 O 应为此电视剧的演员列表。

现今知识图谱主要运用于以下两个方面：自动问答，传统搜索引擎使用的关键词匹配技术没有理解查询词的语义，现在百度搜索已不再返回大量匹配的网页，而是直接显示实体卡片，此类卡片覆盖了人物、诗歌字词、股票等多个领域，见图 2.1 自动问答：



图 2.1 自动问答



图 2.2 关系推理

关系推理，基于知识图谱的关系推理，是现在搜索引擎的新特点，通过关系之间的同现现象来发现推理规则，见图 2. 2 关系推理。

知识图谱以实体为粒度理解用户意图和展现搜索结果，让人和搜索引擎的交互更加自然；基于实体的计算，直接给出答案，使搜索引擎更加智能；利用实体的属性和关系，天然的纽带连接人和服务；开放的平台，面向全网，数据来源多样，提供更全面，更有价值的结果，让优质资源更容易被用户发现。

## 2.2 关系抽取技术对比和优选

维基百科、Freebase、谷歌或百度百科知识图谱等知识库与互联网数据相比只能算沧海一粟，因此需要从海量互联网网页中直接抽取实体关系，关系抽取是一种典型的信息抽取任务，其主要思路包括三种：

### 1. 基于模式匹配的方法不断迭代抽取关系实体对

以“人物”和“地点”两个实体抽取“出生”关系为例，最初可以使用人工设定的模板“PEOPLE 出生于 LOCATION”抽取出 <PEOPLE, LOCATION> 的关系实例 < 张国荣, 香港 >，再使用此关系实例发现新的模板进行进一步提取。模式匹配的思路在扩展过程中容易产生大量噪音，降低抽取的准确率，需要大量人工标注工作。

## 2. 基于语义关系的短语来抽取关系实体对

仍以“人物”和“地点”两个实体抽取“出生”关系为例，我们可以通过句法分析，从文本中发现 < 张国荣，出生于，香港 >、< 张国荣，的故乡是，香港 > 这类关系，通过这种方法得到一个以动词为核心的短语来抽取实体对。基于语义关系的思路会定义大量关系种类，而关系语义表述不一，需要对这些关系进一步聚类。

### 3. 基于关系分类来抽取关系实体对

使用知识图谱已有的关系三元组实例，将实体对看作一个分类样例，实体对关系看



作分类标签，利用机器学习分类模型来构建分类器，对于新的实体对即可直接利用此分类器判断其关系。基于关系抽取的思路如果只是匹配出现实体对的句子，也会引入大量噪音训练样例。

关系抽取通过已有实体关系对（也就是 S 和 O）进行模式匹配或训练分类器来识别关系 P 并抽取新的实体关系对，如已知“出生地”这一关系的模式，能从海量数据中找到表述此关系的新的句子，再抽取新的实体关系对（命名实体 PEOPLE 为 S，LOCATION 为 O）；构建知识图谱的目标是发现已知命名实体的新的属性或关系，如果能够保证文本一定在表述此命名实体的某种属性或关系，可直接使用关系抽取的抽取方法提取答案。当前关系抽取主流的两种方法是 Bootstrapping 方法和 Distant Supervision 方法。

### 2.2.1 Bootstrapping 技术

Bootstrapping 技术首先人工选取少量的关系实例作为种子集合，然后利用模式匹配或者训练模型分类器的方法，通过多次迭代，不断获取新的关系实例并添加到此关系实例集合，最终得到足够的关系实例[5]。

其主要思路（如图 2.3 Bootstrapping 所示）是：

1. 事先定义关系训练集
2. 训练关系分类器
3. 通过关系分类器去挖掘新的实体关系
4. 人工标注或直接加入到原先训练集中
5. 重复 2~4

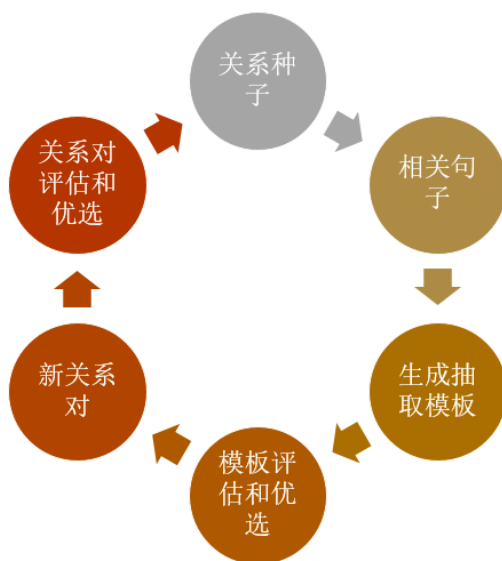


图 2.3 Bootstrapping



以“人物”和“地点”两个实体抽取“出生”关系为例，首先构建少量  $\langle \text{PEOPLE}, \text{LOCATION} \rangle$  的关系实例，如  $\langle \text{张国荣}, \text{香港} \rangle$  等，在已有的语料库中得到与关键词“张国荣”和“香港”有关并表述关系“出生”的句子，通过提取句子模式使用聚类算法得到不同关系的模式，再从互联网或其他语料库中得到关于“出生”关系的句子，使用“出生”这一关系的模式提取新的  $\langle \text{PEOPLE}, \text{LOCATION} \rangle$  的实体二元组，再将其加入到原来的关系实例集合中，重复迭代提取新的关系模式和二元组。

### 2.2.2 Distant Supervision 技术

距离监督方法基本假设为如果两个实体有一个关系，那么包含这两个实体的一句话就可能表达这个关系。利用已有知识库蕴含的事实信息作为基础，训练得到实体关系的训练集，在未标注的其他大规模的语料上，利用分类器从自由文本中挖掘新的实体关系，从而补充现有的知识库<sup>[5]</sup>。

其主要思路是：

1. 将句子关系作为类别，组成一个分类训练样本。如果两个句子表达的  $\langle \text{关系}, \text{实体 1}, \text{实体 2} \rangle$  完全一致，则抽取两个句子特征，并将它们合并在一起，组成一个更大的特征向量。
2. 训练不同关系的逻辑回归分类器
3. 在测试阶段，先对句子中的命名实体进行标注，抽取其中的命名实体对和特征。如果多个句子的命名实体对一样，则将它们的特征合并到同一个特征向量中。然后利用逻辑回归分类器，对关系名称进行识别。

仍以“人物”和“地点”两个实体抽取“出生”关系为例，首先需要基于百度知识图谱或者谷歌知识图谱得到大量  $\langle \text{PEOPLE}, \text{LOCATION} \rangle$  的关系实例，如  $\langle \text{张国荣}, \text{香港} \rangle$  等，在已有的语料库中得到与关键词“张国荣”和“香港”有关的句子，通过提取句子特征使用逻辑回归分类器训练不同关系的分类器，再从互联网或其他语料库中得到关于“出生”关系的句子，使用“出生”这一关系的模式提取新的  $\langle \text{PEOPLE}, \text{LOCATION} \rangle$  的实体二元组。

## 2.3 关键技术优选

Bootstrapping 技术和 Distant Supervision 技术各有各的优劣，见表 2.1。在问题抽象过程中，需要了解实际数据，对数据了解得越充分，越容易建立符合实际需求的应用。经过对现有问答系统的详细分析，问题的领域列表举例见表 2.2。调研和评估问题领域主要包括两个方面，一是属性和关系确定，属性和关系确定的方式很多，可以通

过百度百科提取相应领域的属性和关系，也可以从 Web 中提取表述此领域的句子，进行统计确定属性和关系，人物领域内检索量比较高的属性或关系举例见表 2.3；二是种子选取，选取结果主要用在两个方面，一是对抽取系统的结果准确率进行评估，二是作为属性和关系的训练集。

表 2.1 技术优劣比较

	优势	劣势
Bootstrapping	无需标注语，只需大文档集	引入噪音实例和料噪音模板出现语义漂移现象
Distant Supervision	无需标注语料	容易大量噪音训练实例，需要知识库

表 2.2 问题的领域列表举例

食材	人物	字词	关系推理
影视	学术	诗词	旅游景点
生活服务	星座	植物	健康
节日	音乐	作品	人物作品集

表 2.3 人物领域属性或关系举例

女儿	儿子	妻子	丈夫
前妻	前夫	父亲	母亲
女友	男友	身高	体重
别名	生日	逝世日期	作品集

表 2.4 优缺点

模型	缺点	优点
单模型	数据量大	一次解决问题
	预估难度大	能更好解决类别互斥的问题
多模型	可能产生累积误差	单个子模型更容易实现比较准确地预估
	训练和抽取成本高	可以调整各个子模型的权重，来达到最佳效果

预估句子的表述含义有两种思路，一种是建立单模型直接预估表述含义，二是对每种属性或关系建立模型。不同的方式有不同的优缺点，具体见表 2.4。

本文使用机器学习的目的在于判定一个句子是否表述指定属性或关系，目标变量属于离散型，问答系统覆盖的领域广，单模型很难覆盖全面，而且往往训练集数据量比较大，特征维度比较大，事实上在处理大数据集时常存在内存占用的问题，因此本文对不同领域的每种属性或关系建立二分类模型。使用第三章将介绍的方法提取文本特征后，将文本特征转化为特征向量，使用不同机器学习算法进行训练和测试。

由于百度知识图谱在不同领域已有大量的 SPO 数据可用于训练，且对本文问题的

抽象后确定对不同领域的每种属性或关系建立二分类模型，可使用知识库中的关系启发式的标注训练语料，因此本文使用的关系抽取方法为 Distant Supervision。

## 2.4 本文研究的重难点

当前百度知识图谱的知识库主要来源于百科和垂直领域，需要通过全网挖掘来补充和更新百度知识图谱，本文针对互联网上大量非结构化的文本数据，使用关系抽取的思路，利用已有知识图谱的实体和其属性或关系信息，训练属性或关系分类器，在大规模未标注的语料上，根据需要抽取的三元组中的 S 和 P，提取未知答案 O，如图 2.4 单实体问题答案自动提取思路。

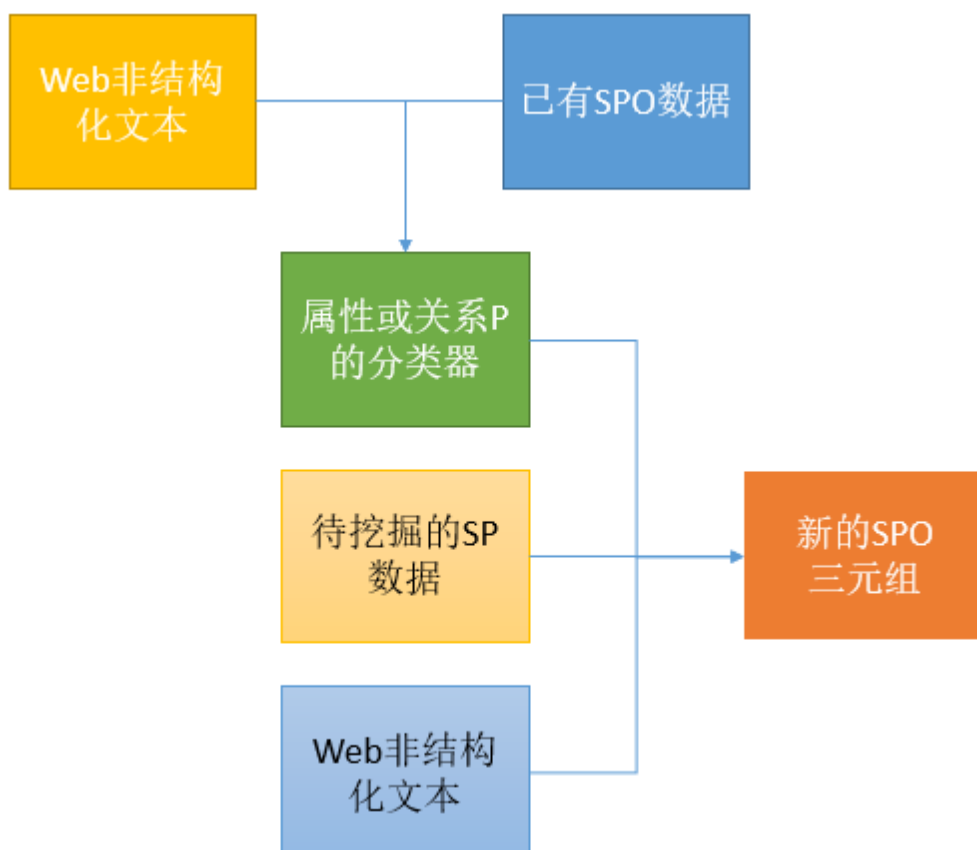


图 2.4 单实体问题答案自动提取思路

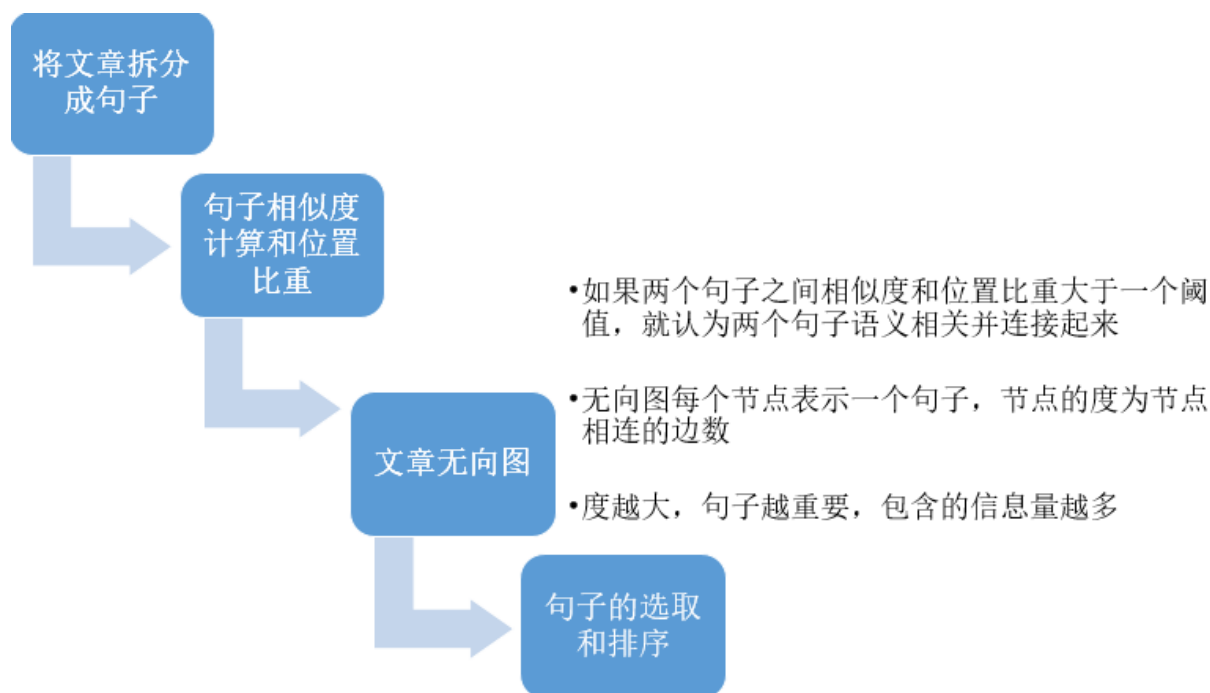
本文主要针对答案是命名实体的领域，使用构建知识图谱三元组的思想 and 关系抽取的方法提取答案，另对答案属于长答案的领域问题，本文通过提取相关文档的摘要来作为最终答案。自动文本摘要主要解决部分领域问题答案属于长句子，因此需要从可能包含正确答案的文本中提取有效摘要，其主要思路如图 2.5 客观问题答案自动提取思路。

因此，本文主要需要解决的重难点有：

### 1. Web 非结构化的文本数据处理

本文数据来源于百度网页库全量数据，数据结构混乱，需要对网页文本进行预处理。

## 2. 中文自然语言处理



鉴于实际运用中发现了大量中文自然语言处理的结果存在问题，本文对部分领域的数据进行优化。

图 2.5 客观问题答案自动提取思路

## 3. 文本特征的选择

目前许多文本特征方法使用的只是一些浅层文本特征，基于距离监督使用的文本特征并不能完全适用于中文和当前互联网杂乱的数据，因此本文通过实践选取了一些新的文本特征。

## 4. 利用机器学习分类模型构建关系分类器

基于距离监督使用的机器学习方法是逻辑回归，本文使用了多种算法进行评估，选择解决当前问题的算法。

## 5. 对于命名实体类型问题的答案提取

关系抽取距离监督的方法可能因为多种原因，仍有一些错误文本，本文对抽取出的备选答案进行了评分和置信度计算，提高最终准确率。

## 6. 文章中句子权重计算

句子权重计算是自动文本摘要的基础，本文句子权重计算在句子相似度计算的基础上加上了句子的位置特征。

## 7. 文摘的可读性加工

为了使文摘具有可读性，本文对文摘进行了去冗余和润色加工。

## 2.5 本章小结

本章主要介绍了知识图谱的研究现状，探讨了关系抽取在知识图谱知识库构建和数  
据更新上运用的方法，并提出了本文的主要思路和关键技术。

下一章将主要介绍使用关系抽取在问答系统答案提取中的关键方法和技术。

### 3 基于关系抽取的答案自动提取的算法设计

本章介绍了关系抽取在问答系统答案提取中用到了句子特征提取方法：语料的获取和处理，提取句子特征所需要文本处理。通过对每个句子特征项的提取可以得到表现实体对及其属性或关系的文本，使用提取到的句子特征训练分类器，判断文本分类后进行答案提取的详细过程。通过对先有问题抽象，选择合适的分类器模型进行训练，提供可用的文本分类判断方法，并对结果进一步验证。针对答案是命名实体类型的问题，本文提供了对备选答案的打分策略，根据 Web 数据特点提出了人物关系领域的权重计算方法。

#### 3.1 Web 语料获取和处理

本文在训练阶段需要使用百度知识图谱提供的关系实体对进行检索得到包含实体对和关系的网页；在测试阶段使用需获取的实体及其属性或关系进行检索得到包含实体及其属性或关系的网页。由于从 Web 中获取的网页文本大多是半结构化或非结构化的，并且网页文本往往比较长，也并不是全是有用的数据，尤其是从一些论坛或置信度不高的网页中获取的数据，常常出现广告或其他无使用价值的文本（宣传站点、推荐其他内容），这些文本并没有实际语义，但其中又分别提到了检索的实体和关系。本文处理的数据是基于检索后得到的包含命名实体及其属性或关系的句子或段落，如果正文提取效果不好，不能准确获取真正有语义的句子，对句子特征提取的准确度影响很大，自然也会影响到后面的答案提取，因此语料的获取是答案提取的基础。对于得到的网页，需要进行正文提取，判断正文中是否有表述实体及其属性或关系的句子，再进一步提取得到最终的语料。由于论文主要目的是解决问答系统答案提取部分，因此对于 Web 数据挖掘获取语料部分只做简要的介绍。

对于不同的网站，正文所在的位置、正文部分的结构并不相同，使用爬虫爬取的页面往往也是各式各样的，不可能对所有页面编写不同的正则表达式来提取正文内容，目前常常使用的通用正文提取方法主要有：

##### 1. 基于标签用途的正文提取

在页面设计时，常常将 TITLE、H1 或 H2 等标签作为标题，P 标签一般表示正文段落，根据标签的含义来提取正文是一种比较常见的思路，在处理部分相对标准的网站比较有效。

##### 2. 基于标签密度法的正文提取

分析网页文件的布局可知，正文部分分布集中，出现的字符比较多，标签对比较少；



而在非正文部分出现的字符较少，标签对标记多。根据这个特征可以使用标签稀疏密度来提取网页正文部分。

范冰冰 muqin 张传美 [http://news.xinhuanet.com/fashion/2014-12/13/c\\_127299721.htm](http://news.xinhuanet.com/fashion/2014-12/13/c_127299721.htm) ,范冰冰的妈妈叫张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长,算是一个事业型的人,这点范冰冰很像妈妈。范冰冰的父母后来都成为山东烟台港务局的文化干部,可以说范冰冰也算是出身在文艺世家。

范冰冰 muqin 张传美 <http://news.fznews.com.cn/fuzhou/20150512/5551bcaf7160e.shtml> 范冰冰的妈妈叫张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长,算是一个事业型的人,这点范冰冰很像妈妈。范冰冰的父母后来都成为山东烟台港务局的文化干部,可以说范冰冰也算是出身在文艺世家。

范冰冰 muqin 张传美 <http://gx.people.com.cn/n/2015/0426/c229131-24638551.html> 原标题:星妈来头大:曲婉婷母亲涉贪,黄晓明妈妈是企业干部,范冰冰的妈妈叫张传美,曾是舞蹈演员,范冰冰的妈妈叫张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长,算是一个事业型的人,这点范冰冰很像妈妈。

范冰冰 muqin 张传美 [http://news.cqnews.net/html/2014-08/15/content\\_31668903.htm](http://news.cqnews.net/html/2014-08/15/content_31668903.htm) ,范冰冰母亲:张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长。张传美为范冰冰付出很多,而且一直很支持她的演艺事业,在《还珠格格》播出之后,张传美当起了女儿的经纪人,帮她打理大小事务,直到王京花签下范冰冰,这种状况才结束。

范冰冰 muqin 张传美 <http://gx.people.com.cn/n/2015/0211/c367607-23867715.html> 原标题:揭明星母亲的超牛背景:王菲妈妈是女高音歌唱家,范冰冰的妈妈叫张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长,算是一个事业型的人,这点范冰冰很像妈妈。范冰冰的父母后来都成为山东烟台港务局的文化干部,可以说范冰冰也算是出身在文艺世家。

范冰冰 muqin 张传美 <http://www.northnews.cn/2014/0806/1687000.shtml> 范冰冰妈妈强势范冰冰的妈妈叫张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长,算是一个事业型的人,这点范冰冰很像妈妈。范冰冰的父母后来都成为山东烟台港务局的文化干部,可以说范冰冰也算是出身在文艺世家。

范冰冰 muqin 张传美 <http://finance.sina.com.cn/money/roll/20140820/151420068597.shtml> 但对于为什么有那么多人会喜欢她,答案却简单得多:她美丽、优雅、高贵。1981年9月16日,范冰冰出生于山东青岛。父亲范涛是海军航空兵文工团的歌手,母亲张传美是舞蹈演员。从小,范冰冰就受到艺术的熏陶,学习长笛、钢琴等乐器。

范冰冰 muqin 张传美 [http://news.cqnews.net/html/2014-08/15/content\\_31668903.htm](http://news.cqnews.net/html/2014-08/15/content_31668903.htm) ,范冰冰母亲:张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长。张传美为范冰冰付出很多,而且一直很支持她的演艺事业,在《还珠格格》播出之后,张传美当起了女儿的经纪人,帮她打理大小事务,直到王京花签下范冰冰,这种状况才结束。

范冰冰 muqin 张传美 <http://www.northnews.cn/2014/0806/1687000.shtml> 范冰冰妈妈强势范冰冰的妈妈叫张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长,算是一个事业型的人,这点范冰冰很像妈妈。范冰冰的父母后来都成为山东烟台港务局的文化干部,可以说范冰冰也算是出身在文艺世家。张传美为范冰冰付出很多,而且一直很支持她的演艺事业,在《还珠格格》播出之后,张传美当起了女儿的经纪人,帮她打理大小事务,直到王京花签下范冰冰,这种状况才结束。

图 3. 1 重复句子举例

本文使用了标签密度法来提取网页正文，在正文提取后，可能仍然会提取到一些错误的内容，比如网站的导航、相关信息推荐链接等，对于此类数据需要根据具体情况去除。如对于相关信息推荐链接，正文提取如果不能去掉，保留下来的就是连续的没有标点符号的长文本，可以以此为特点将其去除。

在实际检索中发现，在不同网页中常常出现描述相同内容的信息，尤其是在娱乐和新闻两个领域，句子重复出现的概率可能比较大（如图 3.1 重复句子举例），这种句子

,1981年9月16日,范冰冰出生于山东青岛。父亲范涛是海军航空兵文工团的歌手,母亲张传美是舞蹈演员。从小,范冰冰就受到艺术的熏陶,学习长笛、钢琴等乐器。虽然家教严格,但范冰冰还是养成了豁达、爱打抱不平的“爷”的性格。

但对于为什么有那么多人会喜欢她,答案却简单得多:她美丽、优雅、高贵。1981年9月16日,范冰冰出生于山东青岛。父亲范涛是海军航空兵文工团的歌手,母亲张传美是舞蹈演员。从小,范冰冰就受到艺术的熏陶,学习长笛、钢琴等乐器。

ratio : 0.684713375796

,范冰冰的妈妈叫张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长,算是一个事业型的人,这点范冰冰很像妈妈。范冰冰的父母后来都成为山东烟台港务局的文化干部,可以说范冰冰也算是出身在文艺世家。

张传美为范冰冰付出很多,在《还珠格格》播出之后,张传美当起了女儿的经纪人,帮她打理大小事务,直到王京花签下范冰冰,这种状况才结束。据说,现在张妈妈还在帮忙打理范冰冰工作室的事务,包括公司财政大权都是由她妈妈张传美掌握,可见范冰冰的妈妈还是很强势的。

ratio : 0.364741641337

,范冰冰的妈妈叫张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长,算是一个事业型的人,这点范冰冰很像妈妈。范冰冰的父母后来都成为山东烟台港务局的文化干部,可以说范冰冰也算是出身在文艺世家。

范冰冰的妈妈叫张传美,曾是舞蹈演员,现任范冰冰影视艺术培训学校校长,算是一个事业型的人,这点范冰冰很像妈妈。范冰冰的父母后来都成为山东烟台港务局的文化干部,可以说范冰冰也算是出身在文艺世家。

ratio : 0.998248686515

图 3. 2 句子去重举例

需要进行去重处理，但这样的句子又不能完全删掉，因为不同网站尤其是权威网站转载了同一篇文章，这篇文章中表述的关系很可能是准确的，因此句子相似度判断，只能用在训练集数据提取过程，减少训练集重复数据，可以使模型更为准确，也降低了数据量，缓解训练时的占用内存过大问题。网页去重的常用的方法是 Google 提出的 SimHash 算法，将高维的特征向量映射成低维的特征向量，在提取文档时计算每篇文档的 SimHash 值，通过比较不同文档 SimHash 值的 Hamming Distance 来确定文章是否重复或者高度近似，通过去掉表述相同的句子（阈值 0.6）见图 3.2 句子去重举例。

### 3.2 网页文本处理和优化

从 Web 获取的网页文本中提取到可能包含需要的实体及其属性或关系的语料集合后，查找需要的实体和关系，提取对应位置前后预定义的 K 窗口的数据，然后对得到的语料进行文本处理，主要包括分词、词性标注、句子依存分析、命名实体识别，其处理结果是本文提取句子词法特征、句法特征和整句特征的前提。

分词是中文自然语言处理的基础，如果在分词时不能正确的切分，对最终命名实体识别的结果影响很大。目前中文切词工具比较多，很多工具的效果也很好，但大多数的切词工具对于很多中国人名、影视作品名以及时间等领域的词语切分效果不好，为了词性标注及之后工作的准确性，针对不同领域需要进行一些优化。词性标注、句子依存分析和命名实体识别可以使用 Stanford NLP Group 提供的句法分析工具，其提供了对一个句子的词汇之间的句法关系。句法的研究是句子的内部结构，以词语作为基本单位；词法研究的是词的内部结构，以语素作为基本单位。句法分析主要包括两个方面，一是分析句子的句法结构：包括句法成分（主语、谓语、宾语、定语、状语、补语）和句法关系（主系、动宾关系、定中关系……）；二是主流分析方法：短语结构分析和依存分析。句子依存分析是用句子中的词之间的依存关系表示句法结构，Stanford NLP Group 中文依存关系类型举例见。命名实体识别是从语料库中，找出包括地名、人名、机构名和时间等词语或语句，对于时间或者一些数值类的实体，识别相对简单，并且可以获得较好的精度，但对于人名、地名和组织机构名等了，由于其多样性，识别过程比较困难，再加上在现在互联网环境中，新词、缩写词或代称不断出现，正确识别命名实体成为提供自动问答质量和效果的难点。现有命名实体识别方法主要包括两类：基于领域规则的方法和基于数据统计的方法 [18]。本文在后面将以中国人名和时间为例介绍这两个领域基于不同规则的命名实体识别方法。对于已分词后的句子“陈赫的老婆出生在福建省福州市长乐市，毕业于浙江传媒”，使用 Stanford NLP Group 其词性标注、句子依存分析见图 3.3 词性标注、句子依存分析，命名实体识别结果见图 3.4 命名实体识别。本文自然文本处理使用了百度 NLPC 提供的切词、词性标注、句子依存分析和命名实体识



别工具，因此论文后面提到的自然文本处理结果使用的是百度 NLP 的表达方法。

表 3. 1 Stanford NLP Group 中文依存关系类型举例

abbrev	缩写
acompl	形容词的补充
advcl	状语从句修饰词
advmod	状语
csubjpass	主从被动关系

### Tagging

陈赫/NN 的/DEG 老婆/NN 出生/VV 在/P 福建省/NR 福州市/NR 长乐市/NN ，/PU 毕业/VV 于/P 浙江/NR 传媒/NN 。/PU

### Parse

```

(ROOT
  (IP
    (IP
      (NP
        (QNP
          (NP (NN 陈赫))
          (DEG 的))
          (NP (NN 老婆)))
        (VP (VV 出生)
          (PP (P 在)
            (NP
              (NP (NR 福建省) (NR 福州市))
              (NP (NN 长乐市))))))
      (PU , )
    )
    (IP
      (VP (VV 毕业)
        (PP (P 于)
          (NP
            (NP (NR 浙江))
            (NP (NN 传媒))))))
      (PU 。 )))

```

### Universal dependencies

```

assmod(老婆-3, 陈赫-1)
case(陈赫-1, 的-2)
nsubj(出生-4, 老婆-3)
root(ROOT-0, 出生-4)
case(长乐市-8, 在-5)
nn(福州市-7, 福建省-6)
nn(长乐市-8, 福州市-7)
prep(出生-4, 长乐市-8)
conj(出生-4, 毕业-10)
case(传媒-13, 于-11)
nn(传媒-13, 浙江-12)
prep(毕业-10, 传媒-13)

```

图 3. 3 词性标注、句子依存分析

陈赫/PERSON 的/O 老婆/O 出生/O 在/O 福建省/GPE 福州市/GPE 长乐市/GPE ， /O 毕业/O 于/O 浙江/GPE 传媒/O 。 /O

图 3. 4 命名实体识别

## 1. 中国人名自动标记

现有中文人名识别方法大多基于规则匹配，在分析句子过程中，当扫描到提前标定

好的中文姓氏时，采集包含中文姓氏及其之后的成分，通过对后位置长度的限制来确定中文人名。在实际使用操作过程中可以发现，这种方式常常存在两个问题，一方面一些中文姓氏用字（“于”、“曾”、“张”等）用途比较广泛，这种词被认为是中文姓氏时，常会出现识别错误，例如“陈赫的老婆出生在福建省福州市长乐市，毕业于浙江传媒”，通过这样的方式，分词后的结果为“陈赫 的 老婆 出生 在 福建省 福州市 长乐市 ， 毕业 于浙江 传媒”，“于浙江”即为错误结果；另一方面由于中文姓氏用字多样，随意扩大或缩小中文姓氏集合对最终结果都会产生影响，扩大中文姓氏集合可能导致大量非人名词语被标注成人名，缩小中文姓氏集合可能导致部分人名词语不能被识别出来。

在真实语料中测试发现，中文切分人名普遍存在两种错误，一是将姓氏和人名切分开，如“张嘉译 的 前妻 叫 杜 珺 ， 也 是 一 名 演 员”，二是人名被切分开，如“张嘉译 在 2010 年 生 下 小 女 儿 张 译 心”。为了避免上述不足，本文在实际操作过程中对百度检索中高 PV 的人名设置了人名词典，以提高最终命名实体识别的准确率。另外本文提出在分词的基础上使用一些合并策略来正确识别中文人名，其主要优势是不需要学习和人工直接干预，在大规模真实语料库上进行测试实验表明，该方法在保证准确率的情况下能有效提高召回率。其主要步骤如下：

- 1) 首先对从网页库拿到的语料进行分词和词性标注；
- 2) 扫描切分后的结果，如果一个 Token 为中文且长度为 1 并属于百家姓集合，再判定如果其后的 Token 为中文且具有名词属性（NR，NZ，NRT 等），则合并两个词；
- 3) 如果一个 Token 为中文且长度为 2，其第一个字属于百家姓集合，再判定其后的 Token 词性确定是否合并；
- 4) 将长度为 1 或 2 的 Token 和其之后的一个字合并成新的人名，并对所有文本进行统计，如果新的人名统计数目和原 Token 数目差异很小则认为新的人名为合理人名。

## 2. 时间自动标记

时间和中国人名标记不同，虽然在切词过程中时间数据的切分通常不准确，但因为时间信息一般都有规律，可以通过正则表达式制定规则来标记时间。本文以餐厅营业时间为例，从大众点评网抽取了五十多万条餐厅营业数据，经过统计可以发现其部分规则举例见表 3.2 时间自动标记规则举例。使用正则表达式制定规则的方法还可以运用在人物生日、身高、体重和年龄等属性或关系的答案提取，因为其限制条件大，所以一般准确率比较高，虽然也会影响命中率，但在语料库数据量大的情况下，特征命中率的问题并不严重。

表 3.2 时间自动标记规则举例

---

(每天|每日|早|上午|平日)?[0-9]+:[0-9]+.{1,6}(次日|凌晨)?[0-9]+:[0-9]+

---

一.{1,6}日|一.{1,6}天

---

\d{1,2}点\d{1,2}分|\d{1,2}:\d{1,2}|\d{1,2}点\d{1,2}分

---

\d{1,2}:\d{1,2}

---

.\*(全天[\u4e00-\u9fa5]\*\$)|(24 小时).\*

---

上午\d{1,2}到晚上\d{1,2}点

---

((早|上)[\u4e00-\u9fa5]?).\*((下|晚)[\u4e00-\u9fa5]?)

---

### 3.3 句子特征

特征提取是一个文本降低维度的过程，一般来说特征提取有两种方法，一是特征选择，从原有的特征中提取出少量的具有代表性的特征，但特征的类型没有变化；二是特征抽取，从原有的特征中重构出新的特征，新的特征具有更强的代表性，并耗费更少的计算资源。其中特征选择的方式又分为 4 种：(1) 用映射或变换的方法把原始特征变换为较少的新特征；(2) 从原始特征中挑选出一些最具代表性的特征；(3) 根据专家的知识挑选最有影响的特征；(4) 用数学的方法进行选取，找出最具分类信息的特征，这种方法是一种比较精确的方法，人为因素的干扰较少，尤其适合于文本自动分类挖掘系统的应用<sup>[19]</sup>。

本文在选择句子特征时，采用了交叉验证的方法来对特征进行调优，对于相同的训练数据和测试数据样本，使用不同句子特征进行评测，由于本文提取特征的句子属于短文本，所以没有必要使用大规模文档处理的方式，最终本文句子特征包括词法特征（Lexical）、句法（Syntactic）特征、命名实体标注结果和整句特征，每个特征描述一个句子间的实体对关系。如果两个命名实体对之间存在一定的关系，则二者应该满足如下条件<sup>[20]</sup>：

1. 命名实体对之间的相对距离不会超过某个阈值  $K$ 。检索返回的结果是文本段，长度是滑动窗口的长度。检索过程中，包含这些关键词文本段就会被返回。一篇文档的检索过程中以滑动窗口的步长滑动，产生许多文本段。
2. 命名实体对在一个句子结构内（如：句号、省略号、感叹号等）。实际上从 Web 获取的很多数据，其文本格式并不标准，很难通过简单的句子结构特征（如：句号、省略号、感叹号等）对文本进行断句。经过统计，对于不能通过句子结构特征进行断句的句子，通常包括文本标题段落或正文中一些解释性文本，这样的文本具有其特殊的 HTML 标记可以用来分句。
3. 命名实体不能是代词、介词等。

4. 如果出现分词错误，分词后的命名实体与新的合成词在统计意义上应该数据差异不大。

### 3.3.1 词法特征

词法特征描述了出现在实体对之间和其周围的特征词汇。描述句子词汇的特征方法很多，主要包括词语的  $TF*IDF$  值、词语位置、是否属于句子关键词以及句法和语义信息等。本文使用的特征包括：

1. 命名实体或属性关系词的 POS (Part-Of-Speech) 标记
2. 查询词 1 左边和命名实体或属性关系词有依存关系的词汇以及其 POS 标记
3. 查询词 2 右边和命名实体或属性关系词有依存关系的词汇以及其 POS 标记
4. 查询词的先后顺序 (S 和 P 的位置关系)
5. 查询词的绝对距离 (S 和 P 之间的距离)
6. 前向标点数目, P 和 S 相隔的标点数
7. 前向代词数目, P 和 S 相隔的代词数
8. 前向动词数目, P 和 S 相隔的动词数
9. 前向名词数目, P 和 S 相隔的名词数
10. 前向助词数目, P 和 S 相隔的助词数

使用 < 范冰冰, 母亲 | 妈妈 > 进行检索, 得到的句子进行切词、词性标注, 并提取的其词法特征举例见表 3.3 词法特征举例。

表 3.3 词法特征举例

范冰冰的妈妈叫张传美, 曾是舞蹈演员
Seq true/Dis 2/Tag 0/Spd 0/Spn 0/Spv 0/范冰冰 nr/的 u/妈妈 n/叫 v/张传美 nr/, w/曾 d/是 v/舞蹈演员 n
母亲张传美为范冰冰付出很多, 而且一直很支持她的演艺事业
Seq false/Dis 3/Tag 0/Spd 0/Spn 0/Spv 0/母亲 n/张传美 nr/为 p/范冰冰 nr/付出 v/很多 m/, w/而且 c/一直 d/很 d/支持 v/她 r/的 u/演艺事业 nz
报道中称范丞丞是范冰冰母亲张传美在范涛的老家青岛所生
Seq true/Dis 1/Tag 0/Spd 0/Spn 0/Spv 0/报道 n/中 f/称 v/范丞丞 nr/是 v/范冰冰 nr/母亲 n/张传美 nr/在 p/范涛 nr/的 u/老家 nz/青岛 ns/所 u/生 v

### 3.3.2 句法特征

根据语言学知识，句子是由主谓宾等构成的主干以及定状补等构成的修饰部分组成。句法特征利用百度 NLPC 句法分析器对句子进行解析，然后从得到的解析树中提取出实体的依存路径。其结果是对句子中的词或词组，利用有向的依存关系边进行连接。具体特征包括：

1. 命名实体或属性、关系的依存路径
2. 连接到命名实体或属性、关系的词和依存路径
3. 对每个实体，增加一个不在依存路径中但与其连接的 Head 结点
4. 连接到 Head 结点的词和依存路径
5. 标注命名实体 S 和其属性、关系 P

仍然使用<范冰冰，母亲>进行检索，得到句法特征（未去除停用词和无关联的特征值）举例见表 3.4 句法特征举例。

表 3.4 句法特征举例

范冰冰的妈妈叫张传美，曾是舞蹈演员
范冰冰 DE_S_2/的 DE_3/妈妈 SBV_P_4/叫 HED_0/张传美 VOB_4/， WP_4/曾 ADV_8/是 IC_4/舞蹈演员 VOB_8
母亲张传美为范冰冰付出很多，而且一直很支持她的演艺事业
母亲 APP_P_2/张传美 SBV_5/为 ADV_5/范冰冰 POB_S_3/付出 CS_11/很多 VOB_5/， WP_5/而且 ADV_11/一直 ADV_11/很 ADV_11/支持 HED_0/她 DE_13/的 DE_14/演艺事业 VOB_11
报道中称范丞丞是范冰冰母亲张传美在范涛的老家青岛所生
报道 ATT_2/中 ADV_3/称 HED_0/范丞丞 SBV_5/是 VOB_3/范冰冰 ATT_S_7/母亲 VOB_P_5/张传美 SBV_15/在 LOC_15/范涛 DE_11/的 DE_12/老家 APP_13/青岛 POB_9/所 SUO_15/生 IC_5

### 3.3.3 整句特征

整句特征是统计整个句子中的部分特征词数目，从而进一步更为准确的描述句子的特征。特征词来源于程序迭代过程中统计不同特征词对文本分类的准确率和召回率的影响，最终确定的具体特征包括：

1. 句子中含有的命名实体个数，包括影视类词、人名、地名等
2. 句子中含有的关系引导词数目，包括人物属性或关系、作品集属性或关系词等
3. 句子中含有的名词个数
4. 句子中含有的动词个数
5. 句子中含有的代词个数

使用 < 范冰冰，母亲 > 进行检索，标注其整句特征见表 3.5 整句特征举例。

表 3. 5 整句特征举例

范冰冰的妈妈叫张传美，曾是舞蹈演员
Ner 2/Rel 1/Nun 4/Nuv 2/Nud 0/范冰冰/的/妈妈/叫/张传美/，/曾/是/舞蹈演员
母亲张传美为范冰冰付出很多，而且一直很支持她的演艺事业
Ner 2/Rel 2/Nun 4/Nuv 2/Nud 1/母亲/张传美/为/范冰冰/付出/很多/，/而且/一直/很/支持/她/的/演艺事业
报道中称范丞丞是范冰冰母亲张传美在范涛的老家青岛所生
Ner 4/Rel 1/Nun 8/Nuv 3/Nud 0/报道/中/称/范丞丞/是/范冰冰/母亲/张传美/在/范涛/的/老家/青岛/所/生

### 3.3.4 特征合并

在进行了词法特征、句法特征和整句特征提取后，还需要进行命名实体标注，由于数据量大，本文使用联合特征来提供精度，虽然联合特征会降低数据的召回率，但每个关系的训练集数据量大，对召回率的影响并不严重。句子中的命名实体将以其命名实体标签代替。前面检索例句合并特征（未去除停用词）如表 3.6 整句合并举例。

表 3. 6 整句合并举例

范冰冰的妈妈叫张传美，曾是舞蹈演员
Ner 2/Rel 1/Nun 4/Nuv 2/Nud 0/Seq true/Dis 2/Tag 0/Spd 0/Spn 0/Spv 0/PER_S nr DE_2/的 u DE_3/RQST_PER_P n SBV_4/叫 v HED_0/PER nr VOB_4/， w _WP_4 /曾 d /是 v _IC_4/舞蹈演员 n
母亲张传美为范冰冰付出很多，而且一直很支持她的演艺事业
Ner 2/Rel 2/Nun 4/Nuv 2/Nud 1/Seq false/Dis 3/Tag 0/ Spd 0/Spn 0/Spv 0/RQST_PER_P n APP_2/ PER nr SBV_5/为 p/ PER_S nr POB_3/付出 v CS_11/很多 m /， w /而且 c ADV_11/一直 d ADV_11/很 d ADV_11/支持 v HED_0/她 r/的 u/ VDO nz VOB_11
报道中称范丞丞是范冰冰母亲张传美在范涛的老家青岛所生
Ner 4/Rel 1/Nun 8/Nuv 3/Nud 0/Seq true/Dis 1/Tag 0/ Spd 0/Spn 0/Spv 0/报道 n/中 f/称 v HED_0/ PER nr SBV_5/是 v VOB_3/ PER_S nr ATT_7/ RQST_PER_P n VOB_5/ PER nr SBV_15/在 p NOR/ PER nr DE_11/的 u/ SNG nz APP_13/ LOC ns POB_9/所 u/生 v

## 3.4 命名实体类型答案提取

机器学习可以分为无监督式学习（Unsupervised Learning）和监督式学习（Supervised

Learning), 在工业实际运用中, 监督式学习更为常见。在建立分类器模型时, 监督式学习先建立一个学习过程, 将预测结果与实际结果进行比较, 通过不断调整模型使模型预测结果达到一个预期的准确率。虽然本文将从 Web 中获取包含 S 和 P 的句子, 并在文本处理过程中对句子进行了窗口限制等, 但在实际进行抽取时却发现仍然存在大量并没有表述指定 P 的句子。以人物关系 < 范冰冰, 母亲 > 为例, 从网页库得到了大量无关句子, 如“他叫董子健, 母亲正是曾一手带出李冰冰、范冰冰等一线大牌的国内著名娱乐经纪人王京花”、“范冰冰王菲赵薇, 揭明星母亲超牛家底背景”, 这些句子虽然包含了搜索关键词, 但并没有正确表述抽取时需要的人物关系, 在进行统计抽取时, 会出现大量错误的 O, 并且很难进行校验。扩展人物 S 集合后, 如果出现比较冷门的人物, 这种问题会更加明显。因此需要先训练分类器, 判断句子实际表述的内容是否属于检索的关系。分类器除了可以在答案提取可以使用外, 在得到备选 O 后, 还可以用备选的 SPO 进行检索, 使用分类器再一次校验, 为备选 O 的置信度计算提供基础。本文采取有监督的机器学习方法主要包括两个流程, 本文机器学习调研流程如图 3.5 机器学习调研流程, 绿色箭头是数据训练过程, 包括数据筛选和清理、抽取特征、训练模型等环节, 每组训练数据有一个明确的分类标识; 蓝色箭头是预估和提取过程, 对需要预估的数据, 抽取特征, 使用训练好的模型进行预估, 获得预估结果后进行答案提取。

本文使用大量包含指定 SPO 和包含指定 SP 但不包含 O 的数据作为训练集, 并使用只包含 SP 的数据作为测试集。为了选择最优算法提高最终的准确率, 本文测试了不同的算法, 并对结果进行交叉验证。在使用分类器对文本的类别进行判定之后, 下一步便是从文本中抽取相应的答案。答案提取主要分为三步, 首先是根据命名实体识别的结果从文本中提出对应的命名实体, 然后计算对每个命名实体可能是正确的 O 的概率, 最后是用提取的 S 和 O 进行进一步检索, 判断得到的句子属于 P 的概率。以 < 范

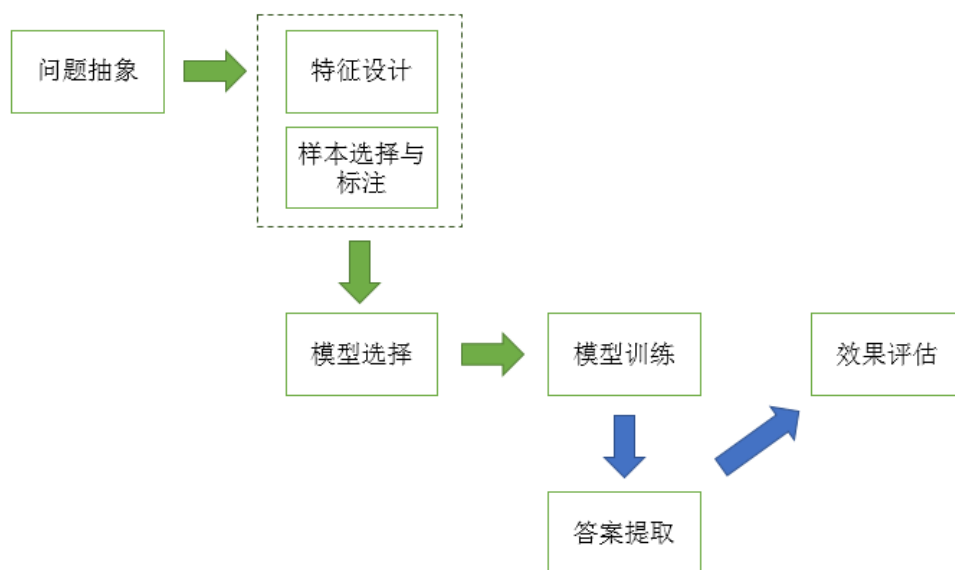
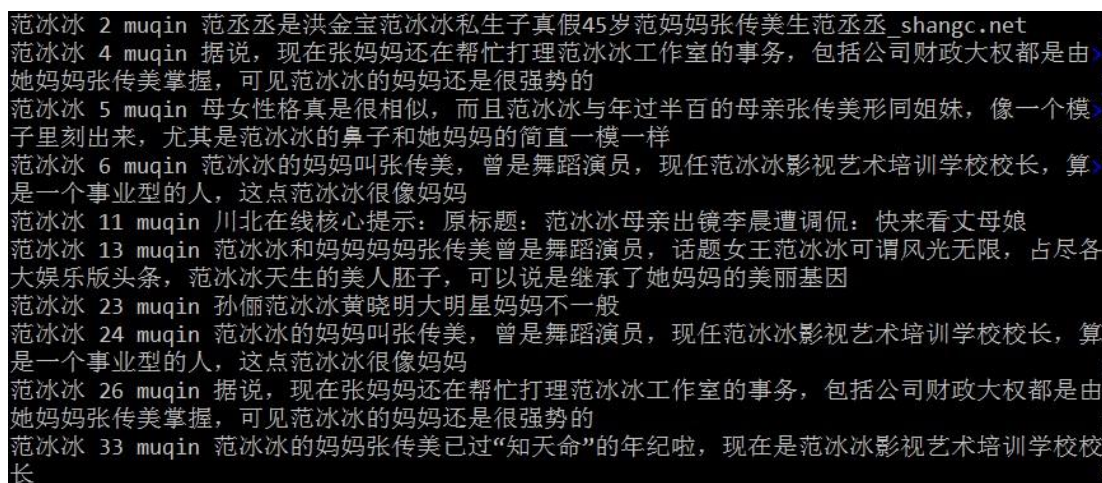


图 3.5 机器学习调研流程



冰冰，母亲 | 妈妈 > 为例，其得到的结果样例如图 3.6 样例，虽然使用分类器过滤掉了一部分明显不表述母亲关系的文本，但剩下的文本中仍然存在错误的可能，虽然数量相对较少。

对这些文本中的命名实体  $x_1, x_2, \dots, x_n$  进行统计，并计算每一种答案的权重  $Weight$ 。比较准确答案 SPO 的相对距离一般比较短，因此 O 到 S 和 P 的相对距离和越小，其属于准确答案的可能性越大，在计算句子权重时可以使用此特点，权重计算公式如公式 3.1。



范冰冰 2 muqin 范丞丞是洪金宝范冰冰私生子真假45岁范妈妈张传美生范丞丞 shangc.net  
 范冰冰 4 muqin 据说，现在张妈妈还在帮忙打理范冰冰工作室的事务，包括公司财政大权都是由她妈妈张传美掌握，可见范冰冰的妈妈还是很强势的  
 范冰冰 5 muqin 母女性格真是很相似，而且范冰冰与年过半百的母亲张传美形同姐妹，像一个模子里刻出来，尤其是范冰冰的鼻子和她妈妈的简直一模一样  
 范冰冰 6 muqin 范冰冰的妈妈叫张传美，曾是舞蹈演员，现任范冰冰影视艺术培训学校校长，算是一个事业型的人，这点范冰冰很像妈妈  
 范冰冰 11 muqin 川北在线核心提示：原标题：范冰冰母亲出镜李晨遭调侃：快来看丈母娘  
 范冰冰 13 muqin 范冰冰和妈妈妈妈张传美曾是舞蹈演员，话题女王范冰冰可谓风光无限，占尽各大娱乐版头条，范冰冰天生的美人胚子，可以说是继承了她妈妈的美丽基因  
 范冰冰 23 muqin 孙俪范冰冰黄晓明大明星妈妈不一般  
 范冰冰 24 muqin 范冰冰的妈妈叫张传美，曾是舞蹈演员，现任范冰冰影视艺术培训学校校长，算是一个事业型的人，这点范冰冰很像妈妈  
 范冰冰 26 muqin 据说，现在张妈妈还在帮忙打理范冰冰工作室的事务，包括公司财政大权都是由她妈妈张传美掌握，可见范冰冰的妈妈还是很强势的  
 范冰冰 33 muqin 范冰冰的妈妈张传美已过“知天命”的年纪啦，现在是范冰冰影视艺术培训学校校长

图 3. 6 样例

$$weight = \sum_{i=1}^n (distance(x_i) + 0.15 + pattern(x_i)) \quad (\text{公式 3.1})$$

其中  $distance(x_i)$  为抽取到的命名实体到已知的 S 和 P 的相对距离和的倒数。由于 O 到 S 和 P 的最短距离为 3，因此  $distance(x_i)1/3$ ，在实际情况中，综合考虑距离因素和统计因素，平衡二者对最终权重的影响，每个命名实体出现一次将会加上其统计意义的数值，此数据取  $distance(x_i)$  的均值 0.15。 $pattern(x_i)$  是根据不同领域抽取的常用表达方式。不同的领域可通过统计或使用语言学专家的研究得到不同的  $pattern(x_i)$  关系。针对 Web 文本中大量的错误表述，对于人物关系领域本文还使用以下策略来去除错误答案：

1. 文本去除标点和关联词后，如果取得了多个连续的备选 O，则此次计算并不累加此类备选 O 的  $weight$ ;
2. 取得的文本数据中去除包含标点符号和关联词的 Token，最终的文本中，当 S、P 和预测的 O 相邻（距离都为 1），预测的
3. 为正确答案的可能性很高， $pattern(x_i)+ = 1$ ;
4. 取得的文本数据中去除包含标点符号和关联词的 Token，当 P 和预测的 O 相邻（距离都为 1），预测的 O 为正确答案的可能性高  $pattern(x_i)+ = 0.5$ ;



5. 如果 S、P 和备选 O 之间不存在标点符号，则  $pattern(x_i)+=0.1$ ;
6. 根据 3.2 介绍的中国人名识别的统计方法，对于可能人名加入到备选 O 集合并已相同方式计算其权重，统计合并的可能人名与原人名出现的次数，如果次数相近则使用合并的可能人名作为备选 O，否则则舍弃可能人名。

### 3.5 本章小结

本章介绍了关系抽取在问答系统答案提取中用到了句子特征提取方法：语料的获取和处理，提取句子特征所需要文本处理。通过对每个句子特征项的提取可以得到表现实体对及其属性或关系的“缩影”，为之后使用句子特征进行分类训练和抽取答案提供了基础。使用提取到的句子特征训练分类器，判断文本分类后进行答案提取的详细过程。通过对先有问题抽象，选择合适的分类器模型进行训练，提供可用的文本分类判断方法，并对结果进一步验证。

下一章将主要介绍本章详细介绍了本文实验系统的总体设计思路和系统的总体流程图结构，然后从自然文本处理、句子特征的提取、训练分类器、答案提取这四个模块详细分析系统的实现。

## 4 答案自动提取系统的实现

本章介绍了本文实验系统算法的总体设计思路，系统的总体流程图、类图和系统的设计思想。

### 4.1 系统框架

本文设计与实现了问答系统答案提取部分，系统能够从网页库中提取相关 S 和 P 的数据，并从中抽取答案 O；对于需进行信息整合的数据，系统还实现了单文档自动文本摘要提取。

#### 4.1.1 整体架构

对于命名实体类型的答案，系统主要分为如下几个模块：

1. 对于某个领域 P，根据已有的 SPO 样本，从网页库中采集相关语料；
2. 文本预处理，进行分词、词性标注、句子依存分析、命名实体识别，提取句子特征；
3. 训练领域 P 的分类器；
4. 使用待抽取的 SP 数据，从网页库中采集相关语料；
5. 使用领域 P 的分类器对语料分类，取得待抽取 SPO 文本；
6. 从待抽取 SPO 文本抽取可能的答案 O；
7. 对可能的答案进行置信度校验，得到最终的答案 O。

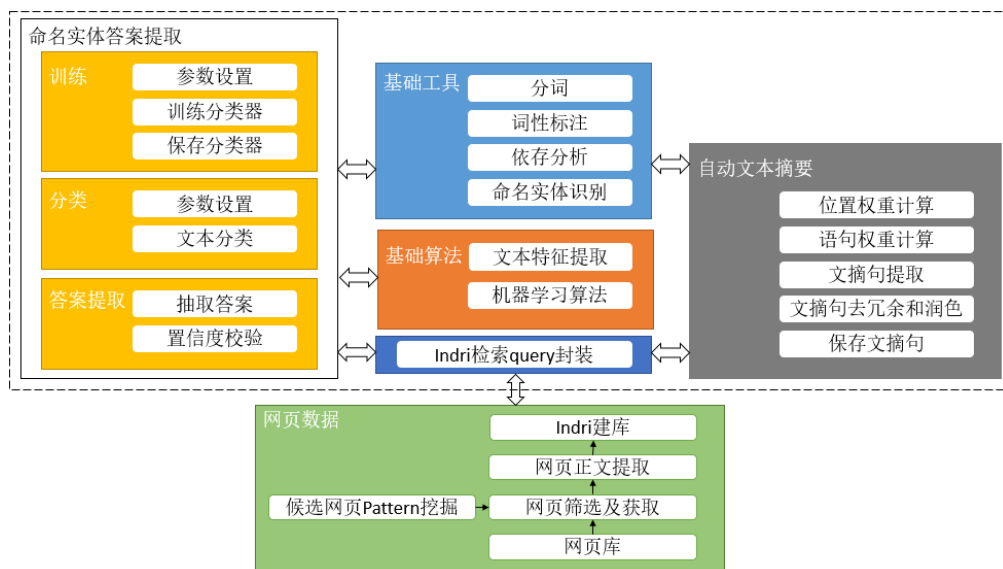


图 4.1 整体架构

对于需进行摘要提取的答案，系统主要分为如下几个模块：

1. 根据待抽取的 SP 数据，从互联网中采集相关的数据；
2. 文本预处理，分段、分句、分词并去除停用词；
3. 计算句子权重，并提取文摘句；
4. 对文摘句进行冗余和润色处理。

本系统的整体架构如图 4.1 整体架构所示。

#### 4.1.2 功能模块分解

如图 4.1 整体架构所示，系统主要分为命名实体答案提取、自动文本摘要、基础文本处理工具、基础算法工具和 query 检索封装五个模块，各个模型详细的功能分解如图 4.2、图 4.3、图 4.6、图 4.5、图 4.4 所示。

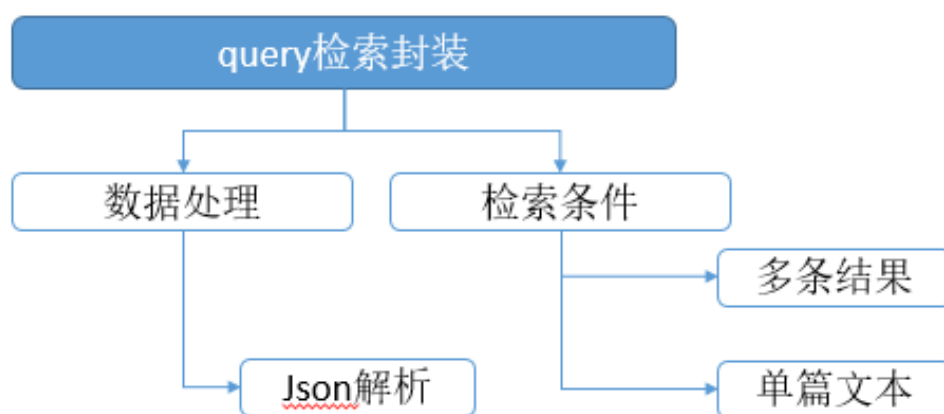


图 4. 3 功能分解

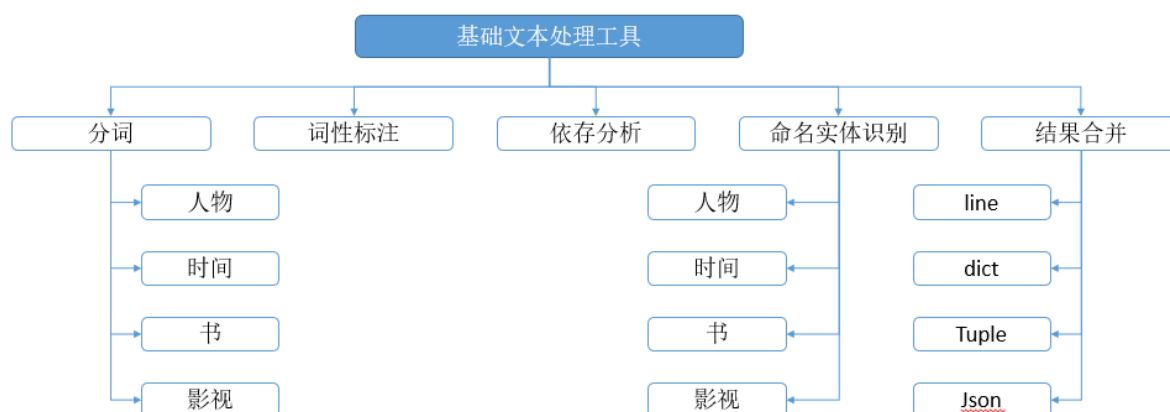


图 4. 2 功能分解

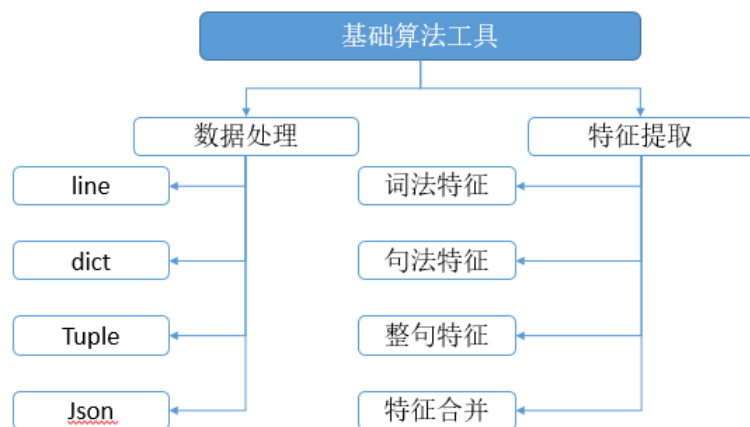


图 4. 6 功能分解

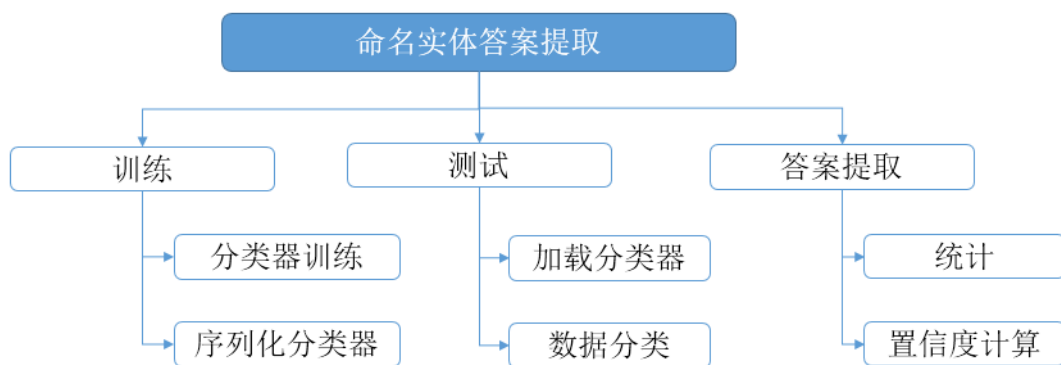


图 4. 5 功能分解

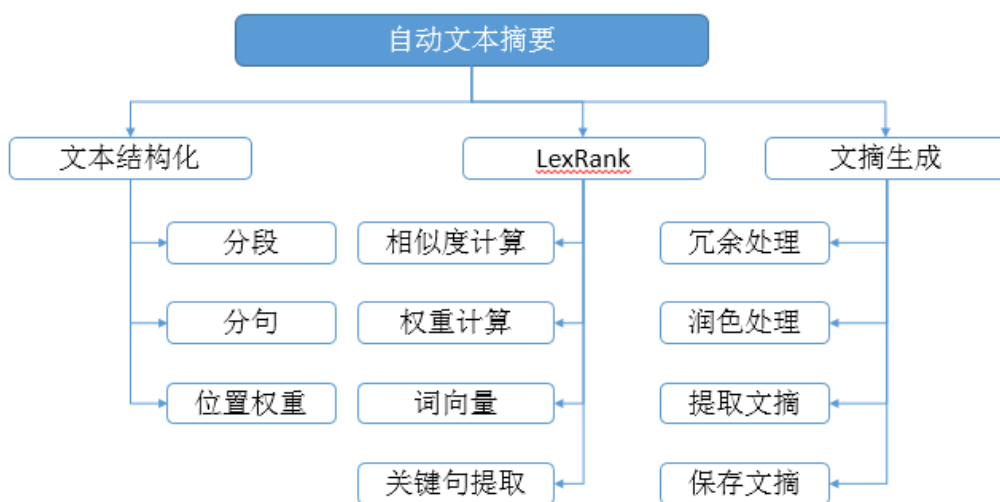


图 4. 4 功能分解

## 4.2 模块详细设计

本文主要包括五个模块：检索模块、基础文本处理模块、基础算法模块、命名实体答案提取模块和自动文本摘要模块，下面将分述各个模块的设计。

### 4.2.1 检索模块

#### 1. 流程图

检索模块预留调用 indri 获取网页库数据的接口并处理得到的 Json 字段，返回不同模块所需要的文本检索结果，其详细流程图见图 4.7。

如图 4.7 所示，检索模块的主要步骤如下：

- 1) 根据其他调用模块的不同检索请求，调用 Indri 的不同接口进行检索；
- 2) 解析 Indri 返回的数据，如果解析失败且重试次数少于 5 次，将再次请求；
- 3) 根据其他调用模块不同的检索请求封装不同的检索结果，返回给其他调用模块；
- 4) 封装检索失败的错误信息并写入日志文件中。

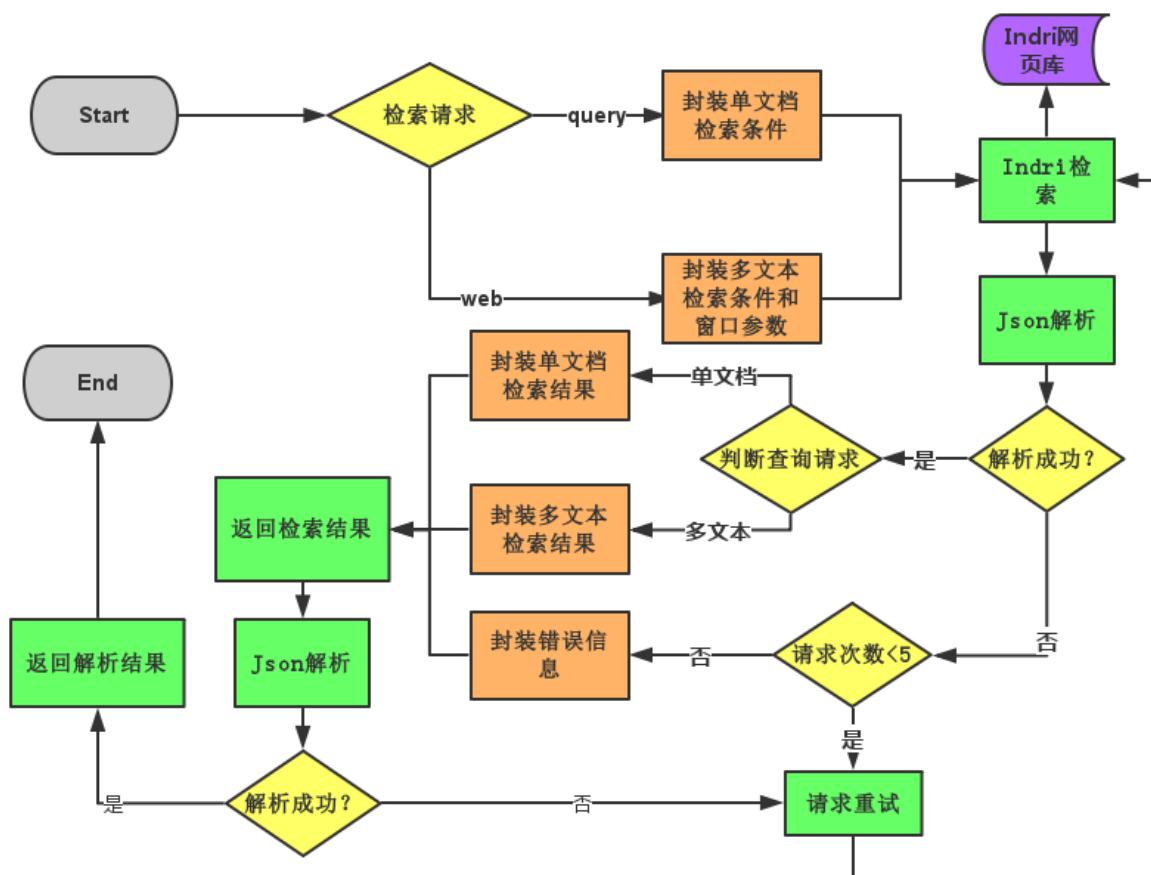


图 4.7 检索模块流程图

#### 3. 类设计

检索模块需要考虑可能存在的新需求，两种不同的请求方式对应的实现类都必须实现统一接口，方便其他模块调用时能使用一致接口。

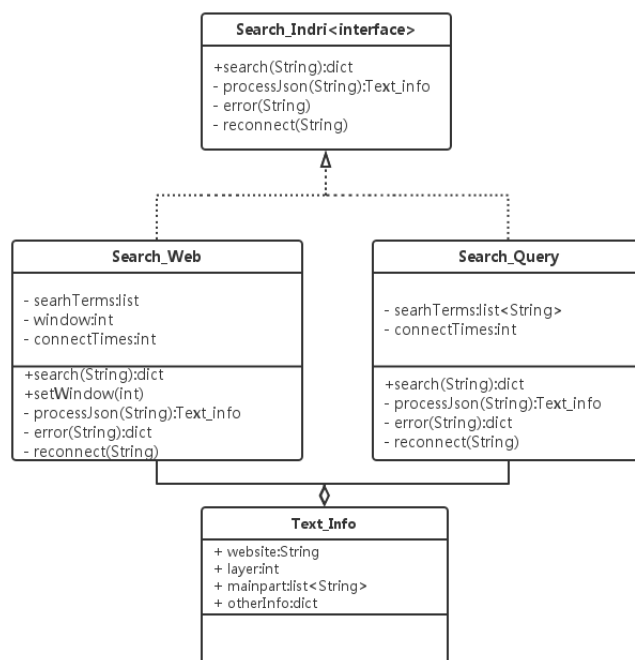


图 4. 8 检索模块类设计

如图 4. 8 检索模块类设计所示检索模块设计使用工厂模式，对外提供统一接口 Search\_Indri，外部可调用方法只有 Search，内部封装了详细的检索请求、处理流程和错误处理，查询多文本段数据时，可修改 S 和 P 的滑动窗口大小。

## 4.2.2 基础文本处理模块

### 1. 流程图

基础文本处理模块预留其他模块调用文本基本处理工具的接口，并预留添加新词的接口，返回不同模块所需要的文本处理结果，其详细流程图见图 4. 9。

如图 4. 9 所示，基础文本处理模块的主要步骤如下：

- 1) 其他模块调用此模块接口，可添加新词；
- 2) 对文本进行分词；
- 3) 并行进行词性标注、依存分析和命名实体识别，如果有错误将封装错误解析并返回结果退出；
- 4) 如果 3 没有发生错误，将针对不同领域（人名、书籍名、影视名、时间）的词汇进行扫描合并和标注；
- 5) 合并 4 处理的结果，如果有错误将封装错误解析并返回结果退出；

- 6) 如果 5 没有发生错误，将根据返回格式的要求（JSON、Tuple、Dict、Line）返回文本处理结果。

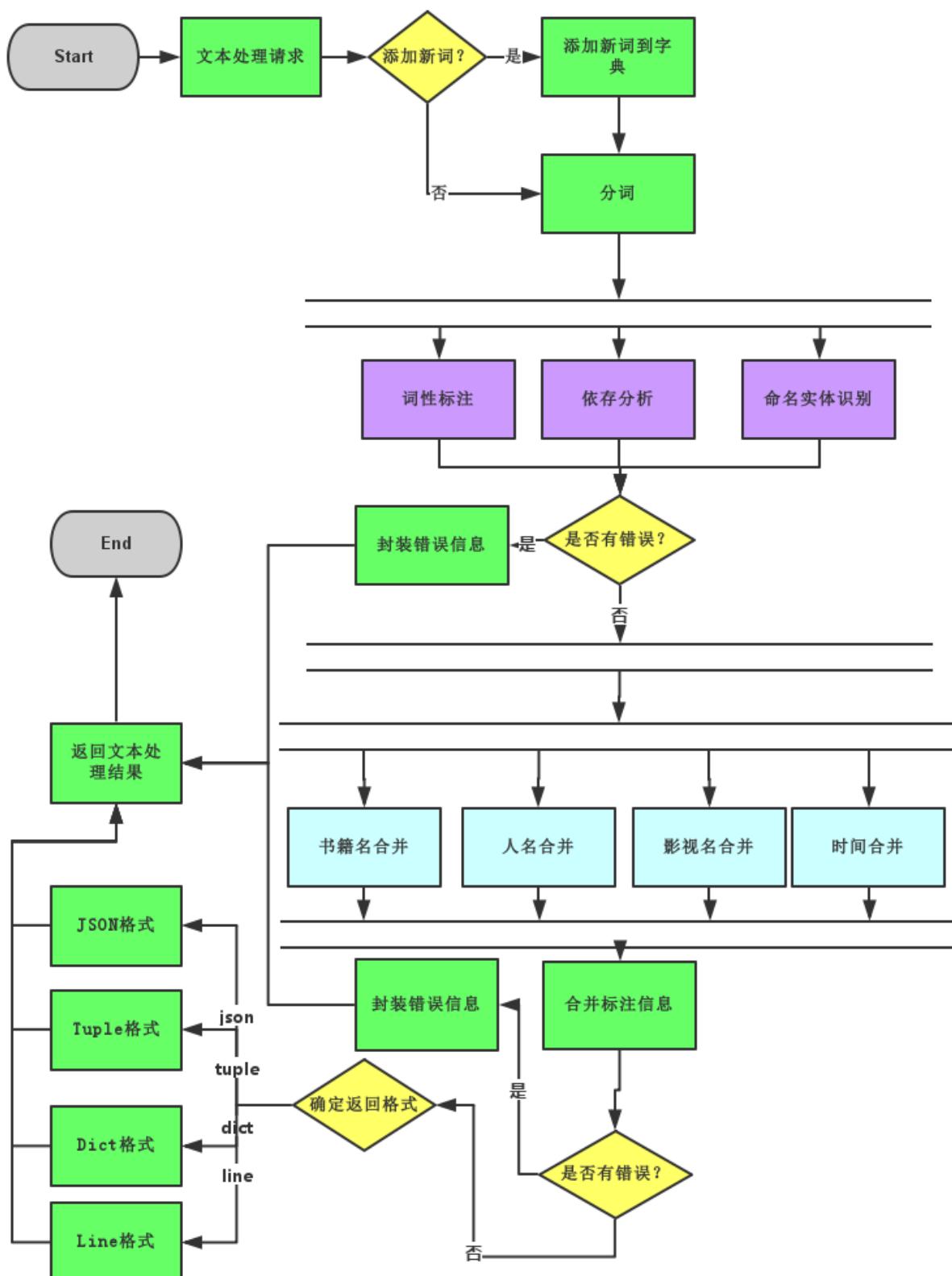


图 4. 9 基础文本处理模块流程图

#### 4. 类设计

基础文本处理模块提供了不同返回格式指定的统一接口，每一个实现类都必须实现指定方法，方便外部模块调用；其对文本相同的详细处理方法由通用类实现，并预留接口调用；考虑其他领域可能需要扩展对文本处理的改进功能，所有领域更新功能需实现统一接口。

如图 4.10 所示，基础文本处理模块使用工厂模式，其他模块只需使用统一接口 Tool，实例化需要的返回结果的实现类，调用统一方法即可；对每一行文本的处理有 Process 实现，其使用观察者模式，调用 repair\_domain()方法即可实现对文本进行所有领域的文本处理；所有领域文本处理实现类都必须实现 Repair 的接口，方便统一调用。

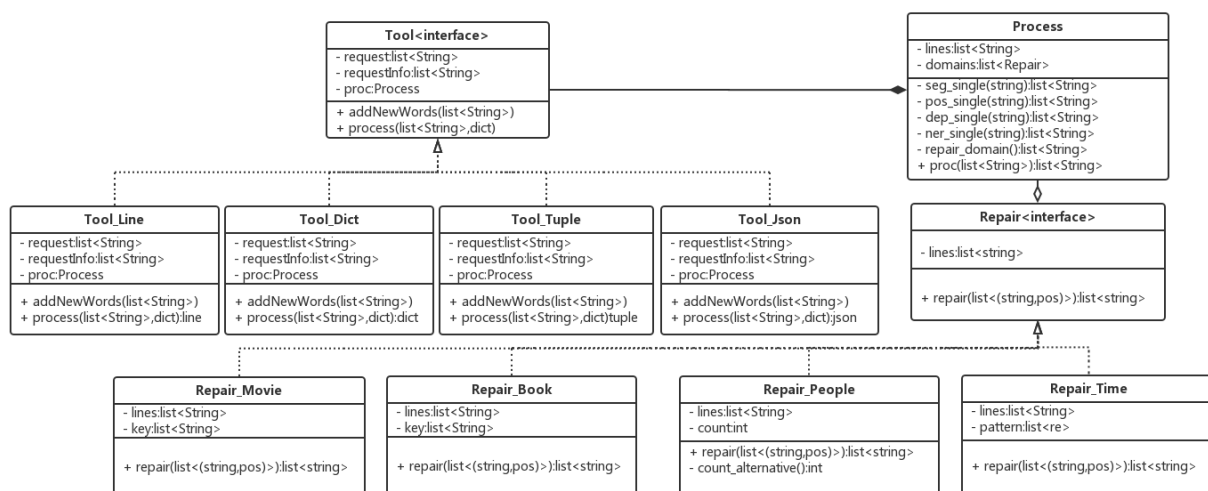


图 4.10 基础文本处理模块类设计

## 4.2.3 基础算法模块

### 1. 流程图

基础算法模块主要解决文本特征提取问题，从文本处理结果的不同格式中提取词法特征、句法特征、整句特征并对特征合并，返回统一结果，见图 4.11。如图 4.11 所示，基础算法模块的主要步骤如下：

- 1) 解析不同的输入格式；
- 2) 并行提取词法特征、句法特征、整句特征；
- 3) 合并所有特征；
- 4) 返回处理结果。



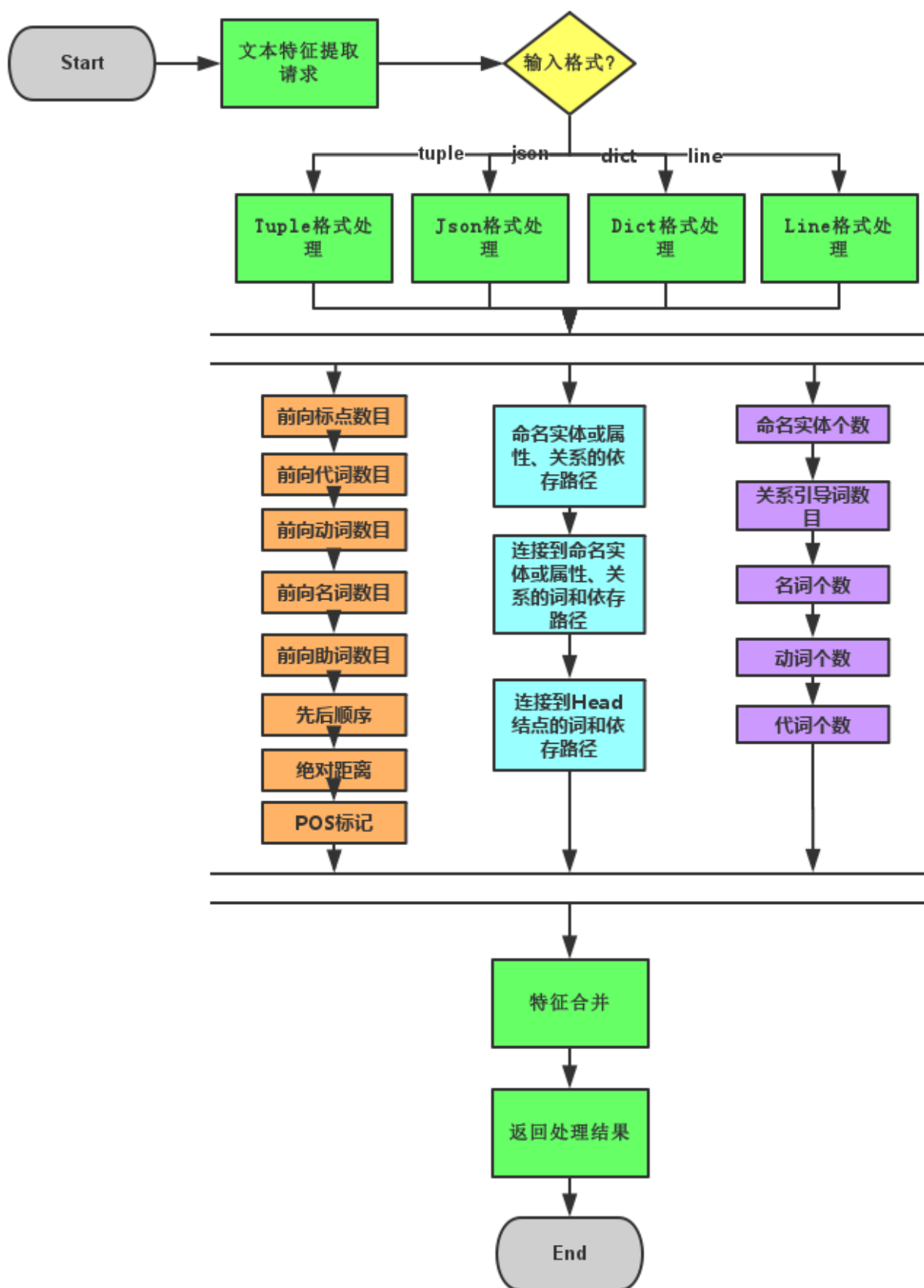


图 4. 11 基础算法模块流程图

## 5. 类设计

基础算法模块提供了不同输入格式的统一接口，每一个实现类都必须实现指定方法，方便外部模块调用；其对文本相同的详细处理方法由通用类实现，并预留接口调用。

如图 4.12 所示，基础算法模块使用了桥模式，把抽象部分和它的实现部分分离开来，让两者可独立变化。抽象部分指的是一个概念层次上的东西，也就是外部调用的接口 `Extract_Feature`，包括输入格式处理和特征抽取；它的实现部分指的是实现词法特征、句法特征、整句特征和特征合并这些特征抽取的详细方法，分离就把实现部分从它要实现的抽象部分独立出来，自我封装成对象。

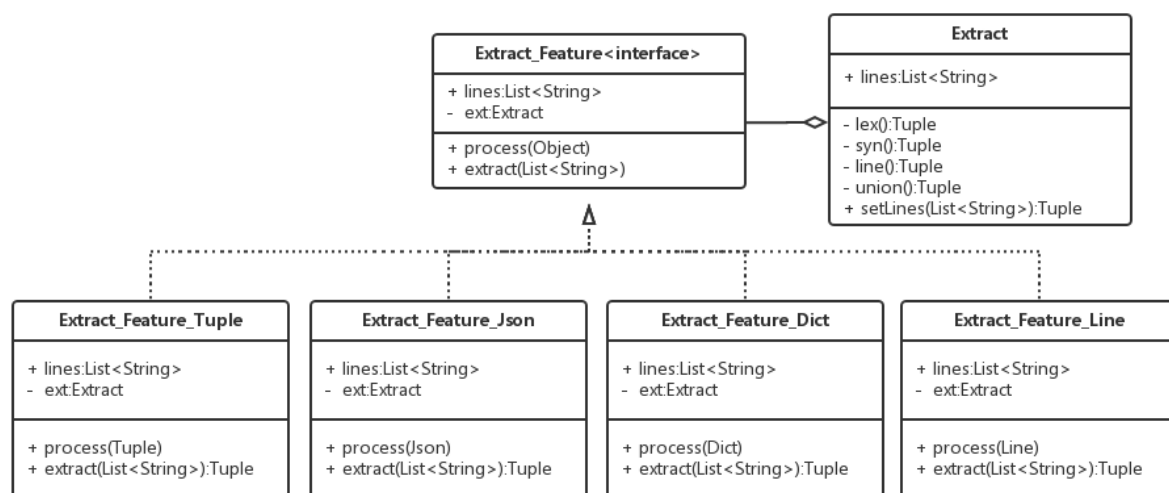


图 4.12 基础算法模块类设计

#### 4.2.4 命名实体答案提取模块

##### 1. 流程图

命名实体答案提取模块提取命名实体类型的答案，使用检索模块提供的文本数据，基础文本处理模块和基础算法模块提取的文本特征，进行训练分类和抽取，最终提取出答案保存到本地，见图 4.13。

如图 4.13 所示，命名实体答案提取模块的训练部分主要步骤如下：

- 1) 选择机器学习算法，设置不同的参数；
- 2) 将文本特征提取结果转化为向量；
- 3) 使用分类器进行训练；
- 4) 保存分类器对象。

命名实体答案提取模块的测试部分主要步骤如下：

- 1) 加载领域的分类器；
- 2) 将文本特征提取结果转化为向量；

- 3) 使用分类器进行文本分类;
- 4) 是否是校验过程, 是则进行步骤 8), 否则进行步骤 5);
- 5) 计算 SPO 绝对距离、Pattern 得分, 统计答案 O 的得分;
- 6) 对待选答案排序, 抽取最终待选答案;
- 7) 使用 SO 进行校验, 提取文本特征, 进入步骤 2);
- 8) 统计 SO 校验得分, 去除得分比较低的答案;
- 9) 保存最终答案并返回。

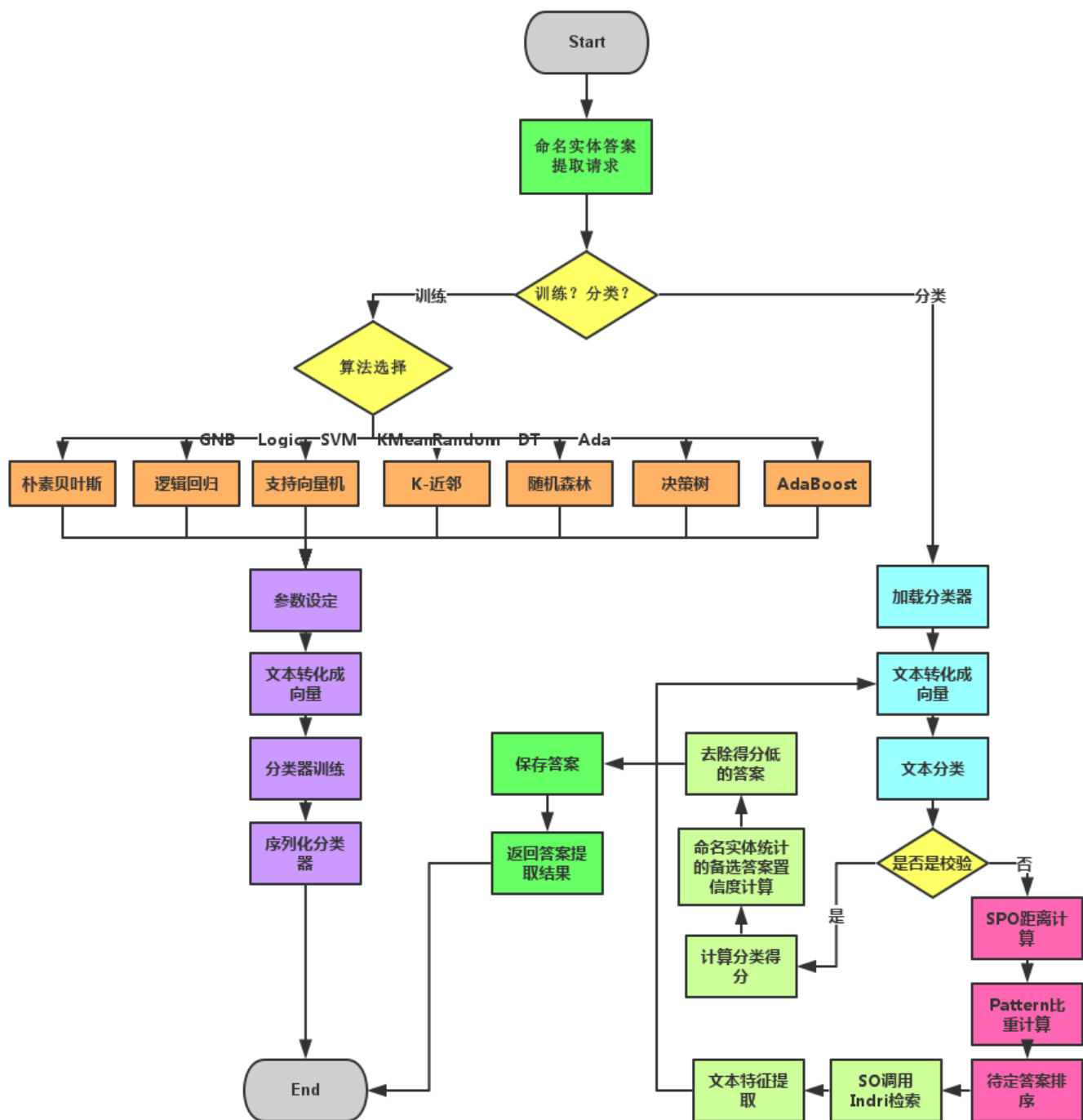


图 4. 13 命名实体答案提取模块流程图

## 2. 类设计

命名实体答案提取模块提供了两个可供外部模块调用的功能，一是训练分类器，二是使用分类器进行答案提取。

如图 4.14 所示，命名实体答案提取模块将分类器单独抽离，训练模块 **Train** 提供修改分类器、设置参数和训练的接口；抽取模块 **Extract** 提供加载分类器和抽取的接口，使用桥模式，将答案抽取得分统计功能和置信度校验功能单独抽离出来。

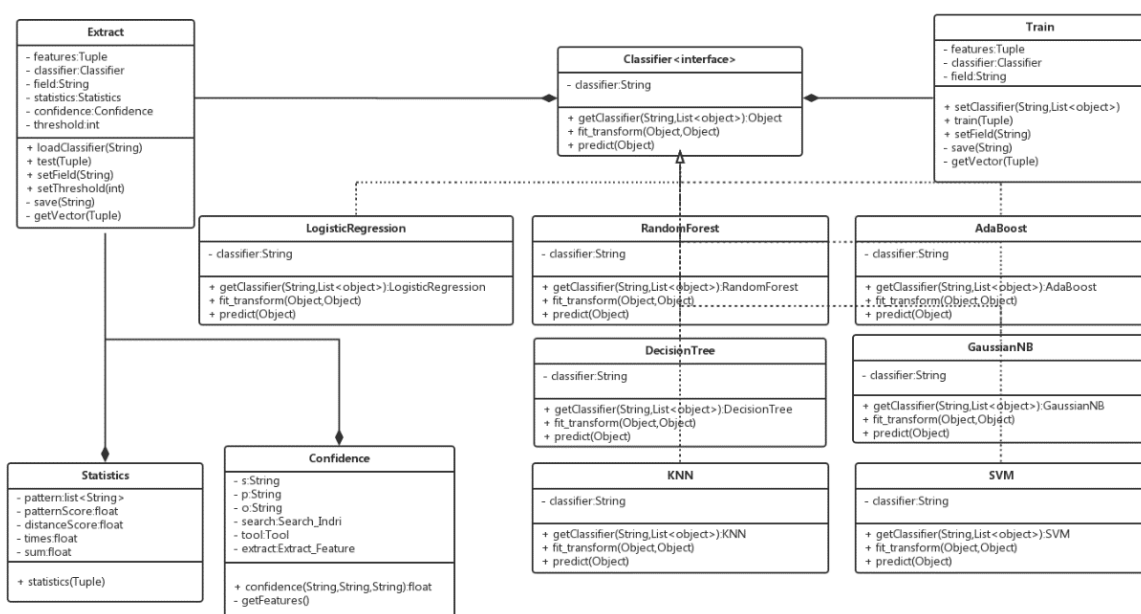


图 4.14 命名实体答案提取模块类设计

### 4.2.5 自动文本摘要模块

#### 1. 流程图

自动文本摘要模块提取单文档类型的答案，使用检索模块提供的文本数据，基础文本处理模块提供的文本预处理功能，计算句子权重，提取出文摘句进行润色将最终的文摘句保存到本地，见图 4.15。

如图 4.15 所示，自动文本摘要模块的主要步骤如下：

- 1) 检索得到可能包含答案的文档。
- 2) 对文档分段并计算每个句子所在的段落权重；
- 3) 进行分句并计算在段落中的位置权重；
- 4) 对文档分词并去除停用词；
- 5) 计算各个句子的相似度，得到文章无向图；

- 6) 使用 LexRank 和句子其他特征计算句子的总权重；
- 7) 根据权重排序并抽取句子，去除冗余句子；
- 8) 根据语言表达特点对文摘进行可读性加工。

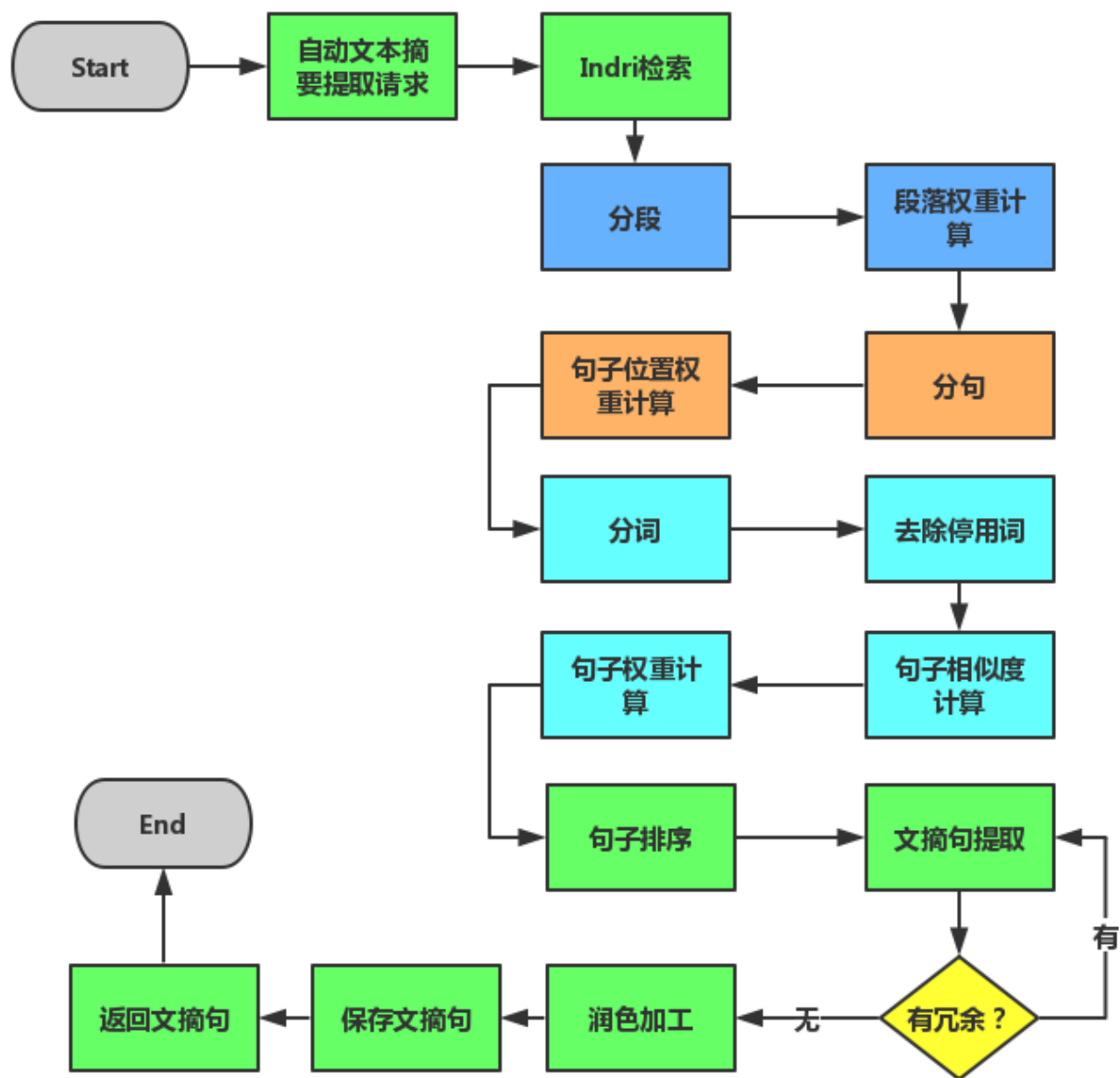


图 4. 15 自动文本摘要模块流程图

## 2. 类设计

自动文本摘要模块提供了一个可供外部模块调用的接口来得到最终的文摘句，其内部封装了句子相似度计算、句子位置权重、LexRank 算法得到的句子权重和对文摘句可读性处理的详细算法。

如图 4.16 所示，自动文本摘要模块提供 **StructuralDocs** 将文档转化为结构化的对象并计算句子的位置权重，**LexRank** 计算句子相似度以及句子最终的权重，并提供排序和取得关键句子的接口，**AutoTextSummarization** 提供了句子去冗余和润色的接口以及最终文摘句生成的接口。

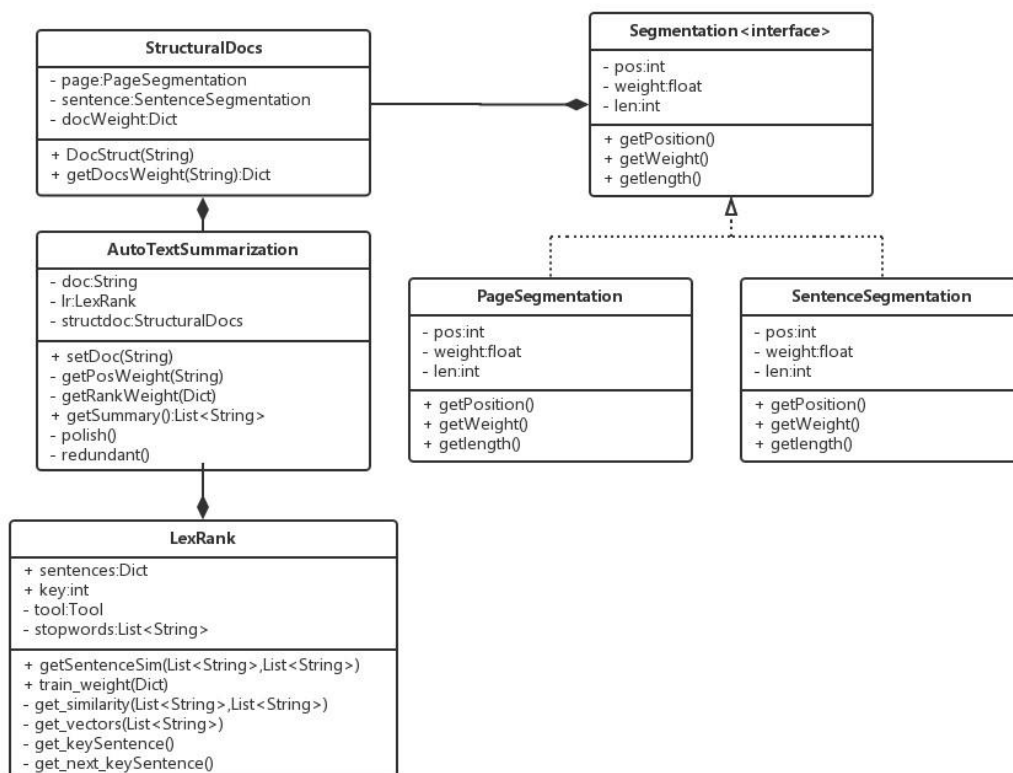


图 4.16 自动文本摘要模块类设计

### 4.3 本章小结

本章介绍了本文实验系统的总体设计思路和系统的总体流程图结构。

下一章将主要介绍如何抽取对于命名实体类型的答案以及如何从可能的答案文档中抽取摘要作为答案以及最终的研究成果。

## 5 实验方案及结果验证

本章介绍了如何抽取对于命名实体类型的答案以及如何从答案文档中抽取摘要作为答案，评估了文本分类、命名实体类型的答案和自动文摘答案的准确率，并分析了最终的实验结果。

### 5.1 实验方案

本文系统实验方案分为平台基础、数据来源和效果评估三个方面，下面将详细介绍这三个方面的内容。

#### 5.1.1 平台基础

本文系统在开发过程中曾使用了多种自然语言处理工具，例如结巴分词、Stanford NLP Group 自然语言处理工具、中科院开发的 ICTCLAS 分词系统，最终评定后选择了百度 NLP 的处理工具。本系统开发工具为：Linux、PHP 2.7。

#### 5.1.2 数据来源

本系统数据来源于百度网页库全量数据，仅去除了部分无用网页（音乐、小说及被标注为色情暴力页面的网页），筛选的网页数据注入到 Indri 建库。

#### 5.1.3 效果评估

答案提取主要包括两个方面，一是对于答案只是命名实体的问题的答案提取，二是对于答案需要总结信息的问题的答案提取。对于命名实体类型，通过第三章的方法，可得到大量可能表述某实体关系的句子，再通过命名实体进行答案提取；对于长句子类型的答案，需要从可能的答案文档抽取其中的文摘句作为最终答案。经过系统处理后提取到的 SPO 数据将使用众包策略进行质量评估，通过百度众测分发评估任务来判断系统工作的准确率。对于命名实体类型，选取答案提取后置信度前 1000 的数据，通过众测分发给用户进行人工标记，用户对准确的关系数据标注“是”，错误的标注“否”，在评估后通过统计对同一关系数据标准为“是”的统计结果来判断数据是否准确；对于自动文摘类型，选取 1000 条最近发生的新闻事实类数据，通过众测分发给用户进行人工标记，用户对文摘描述的准确性和可读性进行评估，评分范围从 1-5，在评估后汇总统计

平均数据。

## 5.2 文本分类

机器学习的一般部分见表 5.1 机器学习处理思路。接下来本文会详细介绍对数据进行预处理和使用不同机器学习算法训练数据，并对文本类别进行判定的过程。

表 5.1 机器学习处理思路

收集数据	本文数据来源于 Web 中获取的网页正文文本，见 2.1
准备数据	根据需要将数据处理成数值型或布尔型数据，见 3.3.1
训练算法	使用不同机器学习算法训练分类器，见 3.3.2
测试评估	统计测试集结果，评估算法召回率和准确率，见 3.3.2

### 5.2.1 数据准备

数据是机器学习解决问题的根本，而好的数据在很大程度上可以帮助提高结果的准确率。训练集可以在网页库中抽取两种数据作为正负数据，一种是包含 SPO 的数据，一种是只有 S 和 P 确定没有 O 的数据，最终再对数据进行一次抽样评估，准备训练数据时还需要注意，待解决的问题数据本身的分布要尽量一致，非必要情况不要做采样，因为采样可能使实际数据分布发生变化，但如果评估结果认定此训练数据存在大量噪音，必须先去除不靠谱的噪音数据，保证训练数据的准确性和有效性。机器学习方法对数据的缩放和尺度计算很敏感，如果数据差异太大，使用可能需要很长时间才能得到结果，因此在开始计算之前需要将特征向量缩放到  $[-1,1]$ ，若干特征变量  $x_1, x_2 \dots \dots x_n$ ，使用公式 5.1。

$$x' = \frac{x - \max x}{\max x - \min x} \quad (\text{公式 5.1})$$

也可以将每个样本缩放到单位范数，也就是对数据进行正则化(Normalization)，对每个样本计算其 P-范数，然后对该样本中每个元素除以该范数（本文使用 2 范数，向量元素绝对值的平方和再开方），这样处理后的数据 P-范数等于 1，P-范数计算公式为 5.2。

$$\|x\|_p = \left( \sum_{i=1}^N |x_i|^p \right)^{\frac{1}{p}} \quad (\text{公式 5.2})$$

第二章介绍了样本标注和特征设计过程（文本特征见图 5.1 文本特征），为了让依存分析结果和命名实体或属性、关系、Head 结点紧密关联，将二者合并为特征名；词



性标注结果数据是有限的，可将每一个词性转化为唯一对应的数值型数据作为特征值。因此每个句子在去掉停用词后都可以转化为多个（特征名，特征值）的有序集合，目前文本表示模型主要包括两类，一类是布尔模型，另一类是向量空间模型（VSM），本文采用第二类模型，使用  $N$  维空间向量来表示一个文本。

由于提取的特征结果是无序特征，可以使用 One-hot 方法把每个无序特征转化为一个数值向量，也就是使用词向量模型，变量长度等于训练数据中的特征名长度，每个词对应于向量中的一个元素。文本分类问题的特征属性很多，再加上本文的训练集太大，特征维度太高，如果保留所有的特征名和特征值和二者对应关系，会占用过多内存。因此本文使用 Feature Hashing，把高维的输入向量哈希变换到一个低维的特征空间中，从而加快算法训练与预测的计算速度并减少内存消耗。由于可能存在不同的特征被哈希到了相同的索引上，本文使用的是 Signed Feature Hashing，实现方法是计算每个特征名到一个低维向量的索引上，然后使用另一个哈希函数来确定是将该特征对应的特征值累加到该低维向量的索引上或是在低维向量的索引上减去此特征值，这样哈希冲突会被抵消而不是错误的累加。在文本分类问题中，特征维度一般取  $2^{10}$  即可，实验表明这样的维度虽然可能仍然存在哈希冲突问题，但对算法的精度影响很小。

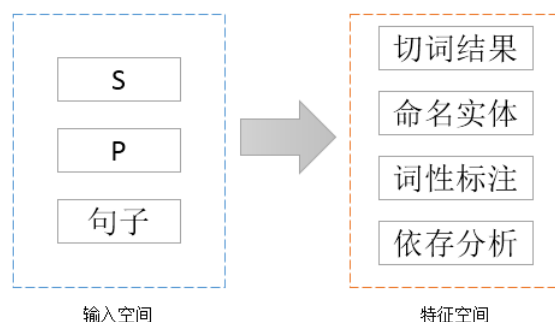


图 5. 1 文本特征

### 5.2.2 文本类别判定

将不同机器学习算法得到的结果进行比较，如果训练集和测试集得到的效果都小于期望目标值，那么当前模型可能出现欠拟合（Underfitting），模型没有正确学到训练数据内在关系，可能是模型假设空间太大或者模型假设空间偏离；如果训练集得到的效果大于期望目标值，而测试集的效果却远差于训练集，则可能出现过拟合（Overfitting），模型过度拟合了训练数据的内在关系，其产生的原因是巨大的模型假设空间与稀疏的

数据之间的矛盾。对于欠拟合问题，需要进一步清洗数据，增加特征并删除噪音特征，考虑使用非线性模型或者使用多个模型组合；对于过拟合问题，需要增加训练数据，对特征进行选择并降低特征维度，考虑使用线性模型。模型的优化的最终目标是在交叉验证中，训练集数据大于期望目标值，测试集数据接近或略逊于训练集。

在文本分类后将进行答案提取，因此需要计算各个分类器对新的准确率和召回率，

	正例	反例
预测正例	A	B
预测反例	C	D

图 5. 2 准确率和召回率

根据图 5. 2 准确率和召回率，准确率和召回率的计算公式见 5.3 和 5.4。

$$\text{Recall} = \frac{A}{(A + C)} \quad (\text{公式 5.3})$$

$$\text{Precision} = \frac{A}{(A + B)} \quad (\text{公式 5.4})$$

综合考虑准确率和召回率的性能度量，使用 F1 度量的一般形式  $F_\beta$ ，见公式 5.5。

$$F_\beta = \frac{(1 + \beta^2) \times \text{Precision} \times \text{Recall}}{(\beta^2 \times \text{Precision}) + \text{Recall}} \quad (\text{公式 5.5})$$

基于本系统对准确率和召回率综合考虑，本文需要文本分类结果里判定为正例的数据中正确的数据居多，也就是相对而言在文本分类阶段更重视召回率，因此上述公式中的  $\beta = 1.5$ 。

本文使用大量包含指定 SPO 和大量包含指定 SP 但不包含 O 的数据作为训练集，在完成了特征设计、抽取，并对特征进行了处理后，本文使用不同机器学习算

算法比较

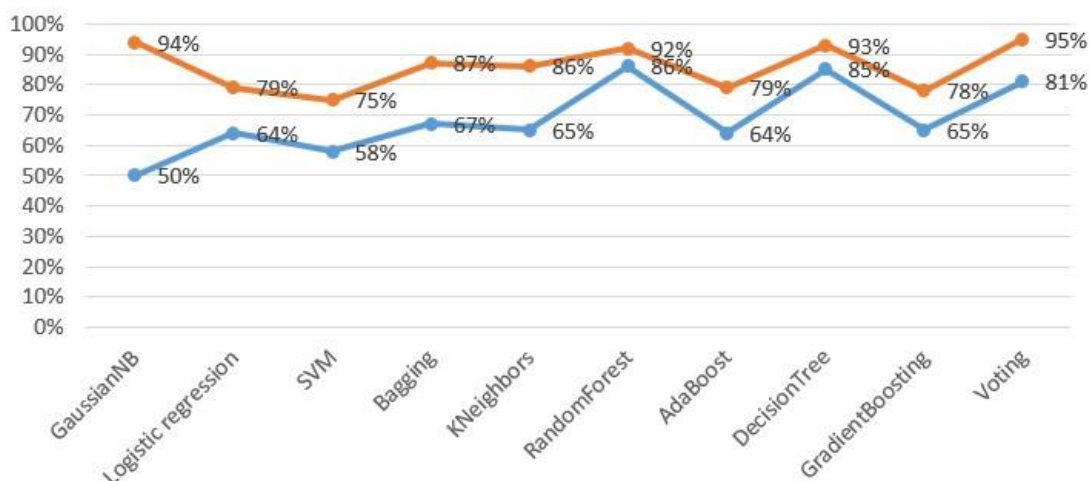


图 5. 3 测试集

法进行训练，在得到训练模型之后，需要评估模型针对新数据的泛化能力，可以在训练时保留部分数据作为测试数据，对训练结果进行交叉验证，通过交叉验证，各个不同算法的在测试集中的精度见图 5.3 测试集。

根据  $F_\beta$  的结果和交叉验证的精度结果，本文最终选择了决策树进行文本分类。

### 5.2.3 实验结果分析

本文最终使用不同机器学习算法进行验证后得到的准确率和召回率见图 5.3 测试集，本文文本分类主要存在的问题如下：

1. 本文分类算法使用无标记数据，训练数据中存在噪音文本。
2. 文本特征提取可以进行进一步的优化；
3. 测试集中 Web 网页文本中存在一些错误语料，尤其是部分语料中表述了需要的关系，但并非与需要的 S 和 O 无关，例如：“范冰冰母亲是张传美，李冰冰母亲是谁？”，句子中包含<范冰冰，母亲，张传美>，但实际需要抽取的 S 是李冰冰。

下面分析了本文使用的部分机器学习算法的优劣。

#### 1. 朴素贝叶斯

朴素贝叶斯方法是一种基于贝叶斯定理的有监督的机器学习算法，需要一批已经标注好的数据作为训练和测试分类器的样本。贝叶斯分类分为两个步骤，第一建立分类模型，根据预定义的不同分类的数据集，通过特征抽取得到其属性来构造贝叶斯分类模型；第二使用分类模型对新的数据集进行划分。本文样本获取主要通过网页抓取并使用第二章的句子特征提取自动标注。对于给定的一类变量  $y$  和若干特征变量  $x_1, x_2, \dots, x_n$  根据贝叶斯定理<sup>[21]</sup>，见公式 5.6。

$$P(y|x_1, x_2, \dots, x_n) = \frac{P(y)P(x_1, x_2, \dots, x_n|y)}{P(x_1, x_2, \dots, x_n)} \quad (\text{公式 5.6})$$

朴素贝叶斯假设每一对特征数据都是独立，也就是说一个特征值对其所属分类的影响独立于其他特征值，特征间不存在相互约束的关系，则可以得到公式<sup>[21]</sup>，见公式 5.7。

$$P(x_i|y, x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i|y) \quad (\text{公式 5.7})$$

对于任意  $i$ ，此关系可以被简化为<sup>[21]</sup>，见公式 5.8。

$$P(y|x_1, x_2, \dots, x_n) = \frac{P(y) \prod_{i=1}^n P(x_i|y)}{P(x_1, x_2, \dots, x_n)} \quad (\text{公式 5.8})$$

在实际中，因为  $P(x_1, x_2, \dots, x_n)$  不依赖于  $y$  并且特征  $x_i$  的值是给定的，分母可以被认为是一个常量，分类规则可以演变为<sup>[21]</sup>，见公式 5.9。

$$P(y|x_1, x_2, \dots, x_n) \propto P(y) \prod_{i=1}^n P(x_i|y) \quad (\text{公式 5.9})$$

这就意味着在上述假设下，朴素贝叶斯分类器使用上述概率模型和最大后验概率

（Maximum A Posteriori）决策准则，相应分类器公式<sup>[21]</sup>，见公式 5.10。

$$\text{classify}(y) = \arg \max_y P(y) \prod_{i=1}^n P(x_i|y) \quad (\text{公式 5.10})$$

为了估计特征的分布参数，可以假设训练集数据满足某种分布或者非参数模型，如高斯模型、多项式模型和伯努利模型。高斯模型假设一个特征属于某个类别的所有观测值符合高斯分布，见公式 5.11。

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right) \quad (\text{公式 5.11})$$

多项式模型常用于文本分类问题，一个分类 $y$ 可用 $\theta_y = (\theta_{y1}, \dots, \theta_{yn})$ 表示， $\theta_{y1}, \dots, \theta_{yn}$ 为其每个特征 $i$ 出现在样本中并属于分类 $y$ 的概率 $P(x_i|y)$ ， $N_{ki} = \sum_{x \in T} x_i$ 表示 $x_i$ 在训练集 $T$ 中类 $y$ 的所有文档下出现的总次数， $N_y = \sum_{i=1}^{|T|} N_{yi}$ 是类 $y$ 下所有特征总数， $x_i$ 对于指定类别 $y$ 的条件先验概率估计见公式 5.12。

$$P(x_i|y) = \frac{N_{ki} + \alpha}{N_y + \alpha n} \quad (\text{公式 5.12})$$

伯努利模型中，一个样本使用的是全局特征，每个特征的取值是布尔型的，在文本分类中，就是指一个特征有没有在一个文档中出现，即公式 5.13。

$$P(x_i|y) = \begin{cases} P(x_i = 1|y) & x_i = 1 \\ 1 - P(x_i = 1|y) & x_i = 0 \end{cases} \quad (\text{公式 5.13})$$

朴素贝叶斯分类器有两个假设，一是统计意义上特征独立，举个例子，假设词语范冰冰出现在演员后面与出现在教师后面的概率相同；二是每个特征同等重要，但在实际表达中，一个句子表述是否合理，可能不需要看完所有词语特征，而只需要看一部分关键特征就可以了。尽管上述两个假设都存在一些瑕疵，朴素贝叶斯的实际效果却很好，与其他机器学习算法相比比较简单快捷，当数据集满足假设时，能得到很好的效果，并且离群点、误差点对分类结果的影响甚小，因为贝叶斯分类器对每个检测样本的归类，都是根据全部训练集的信息得到的，并不受到个别点的影响。

## 2. 逻辑回归

利用逻辑回归进行分类的主要思想是：根据已有训练数据对分类边界线建立回归公式，并以此进行分类。通过自然文本处理拿到数据的特征和标签后可以得到一组训练数据见公式 5.14。

$$D = (x^1, y^1), (x^2, y^2) \dots (x^n, y^n) \quad (\text{公式 5.14})$$

其中 $x^i$ 是一个 $m$ 维的向量， $x^i = [x_1^i, x_2^i \dots x_m^i]$ ， $y$ 在 $\{0,1\}$ 中取值，1表示此关系或属性的正类，0表示负类。逻辑回归本质上是线性回归，在特征到结果的映射中先把特征线性求和，然后使用 Sigmoid 函数 $g(h)$ 作为假设函数来进行预测。当 $h$ 为0时，Sigmoid 函数值为0.5；随着 $h$ 增大，对应值将逼近于1；随着 $h$ 减小，对应值将逼近于0，逻辑

回归的假设函数 $h_{\theta}(x)$ 见公式 5.15 和公式 5.16。

$$g(h) = \frac{1}{1 + e^{-h}} \quad (\text{公式 5.15})$$

$$h_{\theta}(x) = \frac{1}{1 + e^{-\theta^T \cdot x}} \quad (\text{公式 5.16})$$

线性逻辑回归分类器实现思路是在每个特征上乘以一个回归系数，并加所有的结果相加，将总和带入 Sigmoid 函数，进而得到一个范围在 0~1 的值，通过预设的阈值（通常为 0.5）来判断分类。对于线性边界，Sigmoid 函数的输入  $z$  边界形式为： $z = \theta_0 + \theta_1 x_1 + \dots + \theta_n x_n = \sum_{i=0}^n \theta_i x_i$ ，采用向量的写法，上述公式可以写成 $z = \theta^T \cdot x$ ，它表示将这两个数值向量对应元素相乘然后全部相加得到  $z$  值，向量 $\theta$ 也就是最近参数（系数）。本文使用对数似然损失函数来评估  $\theta$  是否比较合适地拟合数据，描述 $h_{\theta}(x)$ 函数不准确的程度，逻辑回归使用对数似然损失函数见公式 5.17。

$$L(h_{\theta}(x), y) = \begin{cases} -\log(h_{\theta}(x)) & y = 1 \\ -\log(1 - h_{\theta}(x)) & y = 0 \end{cases} \quad (\text{公式 5.17})$$

因此逻辑回归的损失函数为见公式 5.18。

$$J(\theta) = \sum_{i=1}^m (y^{(i)} \log g(\theta^T x^{(i)}) + (1 - y^{(i)}) \log(1 - g(\theta^T x^{(i)}))) \quad (\text{公式 5.18})$$

使用批量梯度下降法求解损失函数的最小值，每一步都计算训练数据中所有样本对应的梯度，梯度等于逻辑回归损失函数的偏导， $\theta$ 沿着梯度方向进行迭代，从而得到 $\theta$ 的最优解。其对应的迭代公式为 5.19。

$$\theta_j := \theta_j - \alpha \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} \quad (\text{公式 5.19})$$

逻辑回归的数学模型和求解相对比较简洁简单高效易于实现，但容易欠拟合，分类的精度可能不高。通过对特征做离散化和其他映射，逻辑回归也可以处理非线性问题，在实际应用中，如果有许多低层次特征（含义比较特定的特征，比如本文词性标注、依存分析、命名实体识别的结果）时，可以考虑使用逻辑回归解决问题。

### 3. 支持向量机

支持向量机是一种二类分类模型，其基本模型定义为特征空间上的间隔最大线性分类器。对于线性可分数据，可以将数据集分隔开的直线称为分隔超平面，如果数据点都在二维平面上，此时分隔超平面就是一条直线，当数据集是高维的，那么分隔数据的对象被称为超平面，也就是分类的决策边界。分隔在超平面一侧的数据都属于一个类别，另一侧属于另一类类别，如果数据点离决策边界越远，其最后预测的结果越可信，因此得寻找有最大间隔的超平面。

分隔超平面的形式可以写出 $w^T x + b$ ，计算点 A 到分隔超平面的距离，需要得到点

到分隔超平面的法线或者垂线的长度  $x = \frac{|w^T A + b|}{||w||}$ ，其中  $||w||$  为  $w$  的二阶范数。根据 Heaviside step 函数，对  $w^T x + b$  作用得到  $f(w^T x + b)$ ，当  $w^T x + b < 0$  时输出  $label = -1$ ，反之输出  $label = +1$ 。这样无论数据点处于正方向还是负方向，当其离分隔平面很远， $label * (w^T x + b)$  都是一个很大的正数。要找到合适的  $w$  和  $b$  需要找到具有最小间隔的数据点，也就是支持向量，并对间隔最大化，可以写作公式 5.20。

$$\arg \max_{w,b} \left\{ \min_n \left( label \cdot (w^T x + b) \cdot \frac{1}{||w||} \right) \right\} \quad (\text{公式 5.20})$$

在求解时，设定约束条件  $label \cdot (w^T x + b) \geq 1.0$ ，引入拉格朗日乘子，将超平面写成数据点的形式，当然并不是所有数据都可以 100% 线性可分，引入松弛变量来允许部分数据点可以处于分割面的错误一侧，于是优化目标函数最后可以写成公式 5.21。

$$\max_a \left[ \sum_{i=1}^m \alpha - \frac{1}{2} \sum_{i,j=1}^m label^{(i)} \cdot label^{(j)} \cdot a_i \cdot a_j \langle x^{(i)}, x^{(j)} \rangle \right] \quad (\text{公式 5.21})$$

$$\text{约束条件: } \left( C \geq \alpha \geq 0, \sum_{i=1}^m a_i \cdot label^{(i)} = 0 \right) \quad (\text{公式 5.21})$$

支持向量机的优点是泛化错误低，计算开销不大，结果比较易于解释，然而其对参数调节和核函数的选择敏感，原始分类器不加修改仅适用于处理二类问题。

#### 4. K-近邻

K-近邻算法的思路是：训练样本集的每个样本存在着一个类别标签，对于测试数据样本，将新数据的每个特征和样本集中数据对应的特征进行比较，然后提取样本集中特征最相似的 K 个数据的类别标签，统计得到出现次数最多的分类，作为测试样本数据的分类。

一般特征表示为 N 维向量空间，如向量  $X = (x_1, x_2 \dots x_n)$  与  $Y = (y_1, y_2 \dots y_n)$ ，K-近邻的距离度量方法可以是欧式距离，即二者距离  $d_{XY} = \sqrt{(X - Y)(X - Y)^T}$ ，以及曼哈顿距离，即  $d_{XY} = \sum_{k=1}^n |X_k - Y_k|$ ，或者是向量的余弦值等等。

K-近邻的优点是精度高、对异常值不敏感、无数据输入假定，但其计算复杂度高、空间复杂度也很高，当样本不平衡（一个类别占的总量很大），计算新数据时得到的大容量类的样本容易占多数，分类不准确。

#### 5. AdaBoost

AdaBoost 是 adaptive boosting（自适应 boosting）的缩写，其主要思路是：对训练样本集的每个样本赋予一个权重，开始这个权重都初始化为相等值，这些权重构成向量 D。首先使用这些样本训练一个弱分类器并计算其错误率，然后重新调整每个样本的权重，



其中第一次分对的样本权重会降低，分错样本权重会提高，再在同一样本集上再次训练弱分类器。算法给每个分类器都分配了一个权重 $\alpha$ ，其定义公式如公式 5.22 和公式 5.23：

$$\varepsilon = \frac{\text{未正确分类的样本数目}}{\text{所有样本数目}} \quad (\text{公式 5.22})$$

$$\alpha = \frac{1}{2} \ln \left( \frac{1 - \varepsilon}{\varepsilon} \right) \quad (\text{公式 5.23})$$

计算得到每个分类器的 $\alpha$ 之后，可以对权重向量  $D$  进行更新，使得那些正确分类的样本权重降低而错分权重升高。向量  $D$  计算方法如公式 5.24：

$$D_i^{(t+1)} = \begin{cases} \frac{D_i^{(t)} e^{-\alpha}}{\text{sum}(D)} \\ \frac{D_i^{(t)} e^{\alpha}}{\text{sum}(D)} \end{cases} \quad (\text{公式 5.24})$$

计算出向量  $D$  后，AdaBoost 开始下一轮迭代，不断重复训练和调整权重，直到训练错误率为 0 或弱分类器数目达到指定阈值为止。

AdaBoost 泛化错误率低，易于编码，可以用在大部分的分类器上，无参数调整，但对离群点敏感。

### 5.3 命名实体答案提取

对于命名实体类型，本文使用大量包含指定 SPO 和包含指定 SP 但不包含 O 的数据作为训练集，并使用只包含 SP 的数据作为测试集，使用 5.2 文本分类方法可得到大量可能表述某实体关系的句子，再通过选取命名实体统计校验进行答案提取。

#### 5.3.1 置信度校验

通过文本分类得到表述指定 SP 的数据后，下一步便是从句子中抽取需要的命名实体，以人物关系为例，下一步便会抽取出所有待评分的命名实体类型为 PERSON 的抽取结果，保留抽取的结果作为备选的答案，并使用 3.4 命名实体类型答案提取的权重计算

```
high score current : 范冰冰      muqin  张传美
high score result( score 0.89406779661 length 472) : 范冰冰      muqin  张传美 6.32642973693
high score result( score 0.592369477912 length 498) : 范冰冰      muqin  李晨 2.17880952381
high score result( score 0.486257928118 length 473) : 范冰冰      muqin  董卿 2.0
high score result( score 0.545081967213 length 488) : 范冰冰      muqin  贾云 1.20523809524
high score result( score 0.563524590164 length 488) : 范冰冰      muqin  蔡国庆 0.644475524476
```

图 5. 4 样例

方法计算每个答案的评分，再使用已知  $S$  和备选  $O$  去检索新的文本数据，并将得到的使用对应的  $P$  的分类器进行判断，统计分类器判断结果为  $P$  的数据得到答案的 Accuracy Score 和抽取到包含  $SO$  的文本总数 length，见图 5.4 样例。

最终结果理论上讲应该是得分最高的答案，但在实际工业应用中，将保留抽取结果的前 5 到 10 个数据以及抽取的文本的来源，做进一步的置信度计算和交叉验证。详细的校验方法如下：

1. 使用已有知识库校验是否和已有答案有冲突，比如  $O$  同时是多个人的母亲；
2. 对  $O$  进行性别校验，比如  $P$  是母亲但  $O$  是男性则错误；
3. 通过判断抽取的文本来源网站的置信度，来判断此结果的准确度，比如此答案的大部分抽取结果来源于不可信网站则错误；
4. 用提取的  $S$  和  $O$  进行进一步检索，再使用分类器判断得到的语句属于相应领域  $P$  的概率，去除可能性过低的数据；



图 5.5 准确率

根据图 5.5 准确率，命名实体答案提取的准确率计算方法如公式 5.25。

$$\text{Precision} = \frac{A}{(A + B)} \quad (\text{公式 5.25})$$

以人物领域为例，经过众测评估抽取结果准确率见表 5.2 准确率。

表 5.2 准确率

关系	准确率
妻子	90%
丈夫	80%
父亲	93%
母亲	90%
儿子	90%
女儿	70%
男友	83%
女友	85%

### 5.3.2 实验结果分析

本文人物领域为例，对于命名实体答案提取的最终准确率如表 5.2 准确率。以母亲为例，使用 1800 个  $S$  和  $P$  进行训练，再使用 3000 个  $S$  和  $P$  进行检索判定分类



并抽取结果，统计了抽取器答案分析其错误原因，部分样例见表 5.3 错误样例，不准确的数据主要原因有：

1. 语料问题，语料过少或者语料中有错误数据，约占错误答案的 10%；
2. 文本中包含多重人物关系，分类器没有正确去除此类数据，约占错误数据的 12%；
3. 多个答案（比如毛泽东，多个前妻，暂时作为错误答案进行评估），约占错误答案的 20%；
4. 切词错误（比如希拉里·黛安·罗德姆·克林顿切词结果为希拉里），约占错误答案的 5%；
5. 统计结果为 S 的艺名/别名（比如徐熙媛艺名大 S），约占错误答案的 5%；
6. 命名实体识别错误，约占错误答案的 15%；
7. S 对应的 P 并没有答案 24%。

表 5.3 错误样例

S	P	正确的答案	抽取器答案	错误原因
哈林	母亲	张正芬	伊能静	文本包含多重人物关系
朱棣	母亲	孝慈高皇后	马皇后	同一个人，需消歧
粟戎生	母亲	楚青		未召回
刘楚玉	母亲	王宪嫔	文穆皇后	同一个人，需消歧
曹华恩	母亲	吴速玲	曹格	文本包含多重人物关系，需要排除干扰
张嫣	母亲	鲁元	鲁元公主	同一个人，需消歧
刑嘉倩	母亲	张天爱	林青霞	父亲的前妻

## 5.4 自动文本摘要

对于部分领域问题的答案并不是一个命名实体，它可能是有多个句子组成的答案，此类问题从检索后的文档中抽取出合适的文摘 作为答案。在密西根大学的 Gunes Erkan 和 Dragomir R Radev 两位教授提出的 LexRank 算法通过计算句子间相似度来计算句子权重，如果一个句子与很多其他句子相似，那么这个句子就是比较重要的<sup>[22]</sup>。其计算句子权重的主要思路是根据句子相似度生成文章无向图，无向图每个节点表示一个句子，节点的度为节点相连的边数。如果两个句子之间相似度大于一个阈值，就认为两个句子语义相关并连接起来，节点的度越大，句子越重要，包含的信息量越多， 通过这样的方式构建了图 G，然后使用图排序算法，就得到了每

个节点（也就是每个句子）的权重。

本文在此基础上对每个句子权重其他影响因素，包括句子在此文本中的段落位置和段落中的句子位置，其流程如图 5.6 自动文本摘要。

1. 使用需要提取答案的 S 和 P 进行检索得到可能包含答案的文档；
2. 对文档进行分句、分词并去掉停用词，并计算每个句子所在的段落权重和在段落中的位置权重；
3. 计算各个句子的相似度，得到文章无向图 G；
4. 使用 LexRank 和句子其他特征计算句子的权重；
5. 根据权重排序并抽取句子，去除冗余句子；
6. 根据语言表达特点对文摘进行可读性加工。

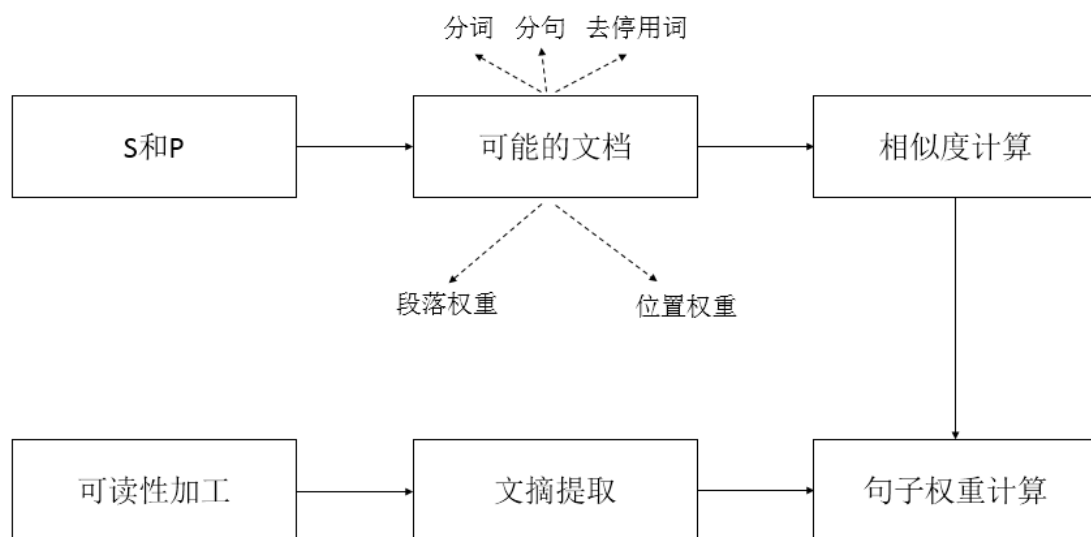


图 5.6 自动文本摘要

#### 5.4.1 句子相似度判断

同第三章一样，句子相似度判断也是采用向量空间模型来表示文本，每个句子表示成一个  $N$  维向量  $V = (w_1, w_2, \dots, w_n)$ ，但使用的句子特征并不是第二章介绍的句子特征， $w_n$  是对应词在文本中出现的次数。因此对于两个句子  $X$ 、 $Y$ ，其向量表示为  $V_x = (w_{x1}, w_{x2}, \dots, w_{xn})$  和  $V_y = (w_{y1}, w_{y2}, \dots, w_{yn})$ ，句子相似度计算方法公式见 5.26 和 5.27。

$$\text{Occur}_i = \begin{cases} 1 & w_{xi} * w_{yi} > 0 \\ 0 & w_{xi} * w_{yi} \leq 0 \end{cases} \quad (\text{公式 5.26})$$

$$\text{Sim}(X, Y) = \frac{\sum_{i=1}^n \text{Occur}_i}{\log \text{len}(X) + \log \text{len}(Y)} \quad (\text{公式 5.27})$$

### 5.4.2 句子权重计算

得到文章无向图  $G$ ，通过 LexRank 算法对每个节点迭代投票得到句子权重记作 LexRankScore，再加上句子位置特征得分 PositionScore，公式见 5.28。

$$\text{PositionScore} = \alpha \cdot \frac{(\text{len}(\text{page}) - \text{pos}(\text{page}))}{\text{len}(\text{page})} \cdot \frac{(\text{len}(\text{para}) - \text{pos}(\text{para}))}{\text{len}(\text{para})} \quad (\text{公式 5.28})$$

其中  $\text{len}(\text{page})$  表示文本段落总数， $\text{pos}(\text{page})$  表示句子所在段落位于文本第几段， $\text{len}(\text{para})$  表示句子所在段落句子总数， $\text{pos}(\text{para})$  表示句子在其所在段落的第几句。

因此句子  $X$  权重  $\text{Weight}_x$  公式见 5.29。

$$\text{Weight}_x = \text{LexRankScore}_x + \text{PositionScore}_x \quad (\text{公式 5.29})$$

### 5.4.3 文摘可读性加工

计算了句子权重后，需要对备选集合去除冗余并对最终的文摘句进行润色加工，步骤如下：

1. 对句子权重排序后选取排序前  $K$  个句子作为备选语句集合  $S = (s_1, s_2 \dots s_k)$ ；
2. 计算二者的最长公共子序列（LCS），通过在  $X$  和  $Y$  的某些位置进行删除操作能得到某个字符串，基于这种方法得到的  $X$  和  $Y$  的最长公共字符串就是二者的 LCS；
3. 计算两个句子的编辑距离  $d(X, Y)$ ，编辑距离等于  $d(X, Y) = \text{len}(X) + \text{len}(Y) - 2 \cdot \text{LCS}$ ；
4. 如果两个句子编辑距离低于阈值  $\beta$ ，则认为两个句子冗余，只取其一并再从排序后的句子列表中抽取新的高权重句子；
5. 循环步骤 2，去除备选语句集合的冗余句子直到集合最终包含  $K$  个可选句子；
6. 备选语句进行润色加工，如果备选语句词首出现代词（如他、他们）、连词（如但是、然而），语句在原文中前一个句子也加入到文摘备选语句集合中；
7. 将这些句子使用原来语序排序得到最终的文摘句。

### 5.4.4 实验结果分析

本文新闻时事类文摘抽取经过众测评估摘要的准确率和可读性为 61%。

经过抽样分析，其主要存在的问题如下：

1. 摘要取出的句子仍然存在连贯性问题，文章中存在一些省略、同义词、指代等内在逻辑关系，从中直接取一些关键句子作为摘要时，部分句子脱离上下文因而不能

正确理解;

2. 一篇文章可能包含多个要点,通过现在的方式取出的文章句可能没有覆盖原文的所有重点;

3. 现在文章的准确率和可读性评价指标可能不够公允。众测分发评估,一方面答题人本身对摘要的准确率和可读性的理解不一样,另一方面在阅读了大量文章后可能出现了疲惫等现象,评估标准相对主观;

## 5.5 本章小结

本章介绍了如何抽取对于命名实体类型的答案以及如何从答案文档中抽取摘要作为答案以及最终的研究成果。

下一章将主要介绍本文工作总结和下一步工作的展望。

## 6 总结与展望

本章主要介绍本文工作总结和下一步工作的展望。

### 6.1 全文总结

问答系统答案提取是实现自动问答的关键步骤。对于基于知识图谱的问答系统，从互联网数据中抽取命名实体及其属性关系，构建知识库关系三元组十分重要；针对知识图谱很难挖掘到非客观的长答案，对于需要整理信息的问题，长文档自动提取文摘也是答案提取的关键之一。本文针对以上两点对问答系统答案提取进行了详细的研究，主要研究内容如下：

1. 鉴于网页库中的网页数据没有固定的正文提取方式，分析了网页特征并给出了正文提取方法；
2. 介绍本文自然文本处理所需要的关键，分词、词性标注、句子依存分析和命名实体识别，以及其实现方法；
3. 研究文本分类方法中的特征提取方法，并提出了本文的文本特征；
4. 详细分析文本问题，对问题进行了抽象，介绍不同机器学习算法的优缺点；
5. 介绍本文对文本特征数据的基本处理，使用不同机器学习算法对本文数据进行了文本分类，统计了不同算法的提取效果；
6. 对于两种不同的答案，介绍了各自答案提取的方法：
  - 1) 命名实体类型的答案介绍了其答案权重计算方法，最终的准确率以及错误分析，并提出一些置信度计算的思路；
  - 2) 自动文摘类型的答案介绍了相似度计算、权重计算和文摘可读性处理的方法；
7. 设计和实现了一个答案提取的系统，包括检索模块、基础文本处理模块、基础算法模块、命名实体答案提取模块和自动文本摘要模块。

### 6.2 展望

本文在答案自动提取技术上作了一定的研究及探索，所提出的解决方案仍存在一些不足，需要进行进一步的讨论和研究，主要包括：

- 1) 文本特征提取需要进行进一步的优化；
- 2) 命名实体答案提取各个待选答案的打分思路可以再优化；
- 3) 命名实体答案提取的置信度计算工作应该进一步加强，其中主要包括：
  - a) 语料过滤小道消息，或者降低小道消息抽取到的答案的得分，解决绯闻问题

- b) 冲突关系校验，尝试解决“妻子”关系找到“前妻”这类问题
- 4) 自动文摘的相似度计算可加入句法分析和领域特征，提高对关键句子获取的准确率；
- 5) 自动文摘的句子冗余计算的方法可以进一步迭代；

由于本人学识有限，论文中难免出现一些不足之处，欢迎各位专家批评指正。

## 参考文献

- [1] 郑实福, 刘挺, 秦兵, 李生. 自动问答综述 [J]. 中文信息学报. 2002, 16(6):47–53
- [2] 李东园, 白宇, 蔡东风. 面向中文问答的信息检索系统及评测 [J]. 沈阳航空工业学院学报. 2009, 26(3):86 – 89
- [3] 黄波. 中文问答系统中答案抽取的研究与实现 [D]. Ph.D. thesis, 吉林: 吉林大学计算机科学与技术学院, 2010
- [4] 余正涛, 樊孝忠, 宋丽哲, 高盛祥. 汉语问答系统答案提取方法研究 [J]. 计算机工程. 2006, 32(3):183 – 185
- [5] 黄勋, 游宏梁, 于洋. 关系抽取技术研究综述 [J]. 现代图书情报技术. 2013, 29(11):30 – 39
- [6] Eugene Agichtein, Luis Gravano. Snowball: Extracting relations from large plain-text collections[C]. Proceedings of the fifth ACM conference on Digital libraries. ACM, 2000, 85–94
- [7] Mike Mintz, Steven Bills, Rion Snow, Dan Jurafsky. Distant supervision for relation extraction without labeled data[C]. Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2. Association for Computational Linguistics, 2009, 1003–1011
- [8] Sebastian Krause, Hong Li, Hans Uszkoreit, Feiyu Xu. Large-scale learning of relation-extraction rules with distant supervision from the web. The Semantic Web–ISWC 2012, Springer, 2012. 263–278
- [9] Tara McIntosh. Reducing Semantic Drift in Biomedical Lexicon Bootstrapping[D]. Ph.D. thesis, Citeseer, 2009
- [10] Andrew Carlson, Justin Betteridge, Richard C Wang, Estevam R Hruschka Jr, Tom M Mitchell. Coupled semi-supervised learning for information extraction[C]. Proceedings of the third ACM international conference on Web search and data mining. ACM, 2010, 101–110
- [11] Bei Shi, Zhenzhong Zhang, Le Sun, Xianpei Han. A Probabilistic Co-Bootstrapping Method for Entity Set Expansion.[C]. COL-ING. 2014, 2280–2290
- [12] Shingo Takamatsu, Issei Sato, Hiroshi Nakagawa. Reducing wrong labels in distant supervision for relation extraction[C]. Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. Association for Computational Linguistics, 2012, 721–729
- [13] Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, Christopher D Manning. Multi-instance multi-label learning for relation extraction[C]. Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Association for Computational Linguistics, 2012, 455–465
- [14] 胡侠, 林晔, 王灿, 林立. 自动文本摘要技术综述 [J]. 情报杂志. 2010,

- 29(8):144–147
- [15] 秦兵, 刘挺, 李生. 多文档自动文摘综述 [J]. 中文信息学报. 2005,19(6):15–22
- [16] 余正涛, 樊孝忠, 郭剑毅, 耿增民. 基于潜在语义分析的汉语问答 系统答案提取 [J]. 计算机学报. 2006, 29(10):1889 – 1893
- [17] 李岩. 文本情感分析中关键问题的研究 [D]. Ph.D. thesis, 万方数据资源系统, 2014
- [18] 张奇. 信息抽取中实体关系识别研究 [D][D]. Ph.D. thesis, 中国科学技术大学, 2010
- [19] 马渊. 短文本情感分析技术研究 [D]. Master’ s thesis, 万方数据资源系统, 2011
- [20] Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, Alexander Yates. Unsupervised named-entity extraction from the web: An experimental study[J]. Artificial intelligence. 2005,165(1):91–134
- [21] Harry Zhang. The optimality of naive Bayes[J]. AA. 2004, 1(2):3
- [22] Günes Erkan, Dragomir R Radev. LexRank: Graph-based lexical centrality as salience in text summarization[J]. Journal of Artificial Intelligence Research. 2004:457–479



## 致 谢

感谢指导我的导师\*\*\*老师，在写论文过程中，\*\*\*老师多次帮助我发现问题，梳理论文思路，如果不是老师的严格要求，很多地方不会更加深入的去思考和挖掘。

感谢在实习过程中给予我信任的经理刘晓波，帮助我的同事们宋勋超、吴家林、施文祥、谢新文，以及给我很多建议的\*\*\*老师。在做这个项目之前，很多东西对我而言都是全新的，刘晓波经理并没有因为我是本科或者只是实习生而不信任我，反而放心交给我去做调研，给我机会让我自由去实现自己的想法，如果没有经理，我肯定不会坚持下去把项目做完；在做这个项目的时候，我的同事们也很信任我，会用心听我的想法并给我提供了很多建议，大家的支持和帮助是我解决问题的关键；我在实习过程中，\*\*\*老师在讲机器学习，当时在老师的课上和他交流了自己做的东西，老师给了我很多关于文本分类特征选择的建议，感谢这些建议让我跳出了以前的思维局限。

感谢大学四年教导过我的老师们，\*\*\*老师、\*\*\*老师、\*\*\*老师、\*\*\*老师、\*\*\*老师、\*\*\*老师、\*\*\*老师、\*\*\*老师、\*\*\*老师、\*\*\*老师和大一上实践课东软的老师，以及我选修过的\*\*\*老师和\*\*\*老师，谢谢老师们的耐心教导，大学四年我过得很充实，也学到了很多，无论是专业上的还是其他，这些将是我一生的财富，谢谢各位老师！

感谢交大，从不后悔自己在这里度过人生最重要的四年！

## 附 录

### 附录 A 英文原文

#### Abstract

Modern models of relation extraction for tasks like ACE are based on supervised learning of relations from small hand-labeled corpora. We investigate an alternative paradigm that does not require labeled corpora, avoiding the domain dependence of ACE style algorithms, and allowing the use of corpora of any size. Our experiments use Freebase, a large semantic database of several thousand relations, to provide distant supervision. For each pair of entities that appears in some Freebase relation, we find all sentences containing those entities in a large unlabeled corpus and extract textual features to train a relation classifier. Our algorithm combines the advantages of supervised IE (combining 400,000 noisy pattern features in a probabilistic classifier) and unsupervised IE (extracting large numbers of relations from large corpora of any domain). Our model is able to extract 10,000 instances of 102 relations at a precision of 67.6%. We also analyze feature performance, showing that syntactic parse features are particularly helpful for relations that are ambiguous or lexically distant in their expression.

#### Introduction

At least three learning paradigms have been applied to the task of extracting relational facts from text (for example, learning that a person is employed by a particular organization, or that a geographic entity is located in a particular region).

In supervised approaches, sentences in a corpus are first handlabeled for the presence of entities and the relations between them. The NIST Automatic Content Extraction (ACE) RDC 2003 and 2004 corpora, for example, include over 1,000 documents in which pairs of entities have been labeled with 5 to 7 major relation types and 23 to 24 subrelations, totaling 16,771 relation instances. ACE systems then extract a wide variety of lexical, syntactic, and semantic features, and use supervised classifiers to label the relation mention holding between a given pair of entities in a test set sentence, optionally combining relation mentions (Zhou et al., 2005; Zhou et al., 2007; Surdeanu and Ciaramita, 2007).

Supervised relation extraction suffers from a number of problems, however. Labeled training data is expensive to produce and thus limited in quantity. Also, because the relations

are labeled on a particular corpus, the resulting classifiers tend to be biased toward that text domain.

An alternative approach, purely unsupervised information extraction, extracts strings of words between entities in large amounts of text, and clusters and simplifies these word strings to produce relationstrings (Shinyama and Sekine, 2006; Banko et al., 2007). Unsupervised approaches can use very large amounts of data and extract very large numbers of relations, but the resulting relations may not be easy to map to relations needed for a particular knowledge base.

A third approach has been to use a very small number of seed instances or patterns to do bootstrap learning (Brin, 1998; Riloff and Jones, 1999; Agichtein and Gravano, 2000; Ravichandran and Hovy, 2002; Etzioni et al., 2005; Pennacchiotti and Pantel, 2006; Bunescu and Mooney, 2007; Rozenfeld and Feldman, 2008). These seeds are used with a large corpus to extract a new set of patterns, which are used to extract more instances, which are used to extract more patterns, in an iterative fashion. The resulting patterns often suffer from low precision and semantic drift.

We propose an alternative paradigm, distant supervision, that combines some of the advantages of each of these approaches. Distant supervision is an extension of the paradigm used by Snow et al. (2005) for exploiting WordNet to extract hypernym (is-a) relations between entities, and is similar to the use of weakly labeled data in bioinformatics (Craven and Kumlien, 1999; Morgan et al., 2004). Our algorithm uses Freebase (Bollacker et al., 2008), a large semantic database, to provide distant supervision for relation extraction. Freebase contains 116 million instances of 7,300 relations between 9 million entities. The intuition of distant supervision is that any sentence that contains a pair of entities that participate in a known Freebase relation is likely to express that relation in some way. Since there may be many sentences containing a given entity pair, we can extract very large numbers of (potentially noisy) features that are combined in a logistic regression classifier.

Thus whereas the supervised training paradigm uses a small labeled corpus of only 17,000 relation instances as training data, our algorithm can use much larger amounts of data: more text, more relations, and more instances. We use 1.2 million Wikipedia articles and 1.8 million instances of 102 relations connecting 940,000 entities. In addition, combining vast numbers of features in a large classifier helps obviate problems with bad features.

Because our algorithm is supervised by a database, rather than by labeled text, it does not suffer from the problems of overfitting and domain-dependence that plague supervised systems. Supervision by a database also means that, unlike in unsupervised approaches, the output of our

classifier uses canonical names for relations. Our paradigm offers a natural way of integrating data from multiple sentences to decide if a relation holds between two entities. Because our algorithm can use large amounts of unlabeled data, a pair of entities may occur multiple times in the test set. For each pair of entities, we aggregate the features from the many different sentences in which that pair appeared into a single feature vector, allowing us to provide our classifier with more information, resulting in more accurate labels.

Table 1 shows examples of relation instances extracted by our system. We also use this system to investigate the value of syntactic versus lexical (word sequence) features in relation extraction. While syntactic features are known to improve the performance of supervised IE, at least using clean hand-labeled ACE data (Zhou et al., 2007; Zhou et al., 2005), we do not know whether syntactic features can improve the performance of unsupervised or distantly supervised IE. Most previous research in bootstrapping or unsupervised IE has used only simple lexical features, thereby avoiding the computational expense of parsing (Brin, 1998; Agichtein and Gravano, 2000; Etzioni et al., 2005), and the few systems that have used unsupervised IE have not compared the performance of these two types of feature.

#### Previous work

Except for the unsupervised algorithms discussed above, previous supervised or bootstrapping approaches to relation extraction have typically relied on relatively small datasets, or on only a small number of distinct relations. Approaches based on WordNet have often only looked at the hypernym (is-a) or meronym (part-of) relation (Girju et al., 2003; Snow et al., 2005), while those based on the ACE program (Doddington et al., 2004) have been restricted in their evaluation to a small number of relation instances and corpora of less than a million words.

Many early algorithms for relation extraction used little or no syntactic information. For example, the DIPRE algorithm by Brin (1998) used string-based regular expressions in order to recognize relations such as author-book, while the SNOWBALL algorithm by Agichtein and Gravano (2000) learned similar regular expression patterns over words and named entity tags. Hearst (1992) used a small number of regular expressions over words and part-of-speech tags to find examples of the hypernym relation. The use of these patterns has been widely replicated in successful systems, for example by Etzioni et al. (2005). Other work such as Ravichandran and Hovy (2002) and Pantel and Pennacchiotti (2006) use the same formalism of learning regular expressions over words and part-of-speech tags to discover patterns indicating a variety of relations.

More recent approaches have used deeper syntactic information derived from parses of the

input sentences, including work exploiting syntactic dependencies by Lin and Pantel (2001) and Snow et al. (2005), and work in the ACE paradigm such as Zhou et al. (2005) and Zhou et al. (2007).

Perhaps most similar to our distant supervision algorithm is the effective method of Wu and Weld (2007) who extract relations from a Wikipedia page by using supervision from the page's infobox. Unlike their corpus-specific method, which is specific to a (single) Wikipedia page, our algorithm allows us to extract evidence for a relation from many different documents, and from any genre.

#### Freebase

Following the literature, we use the term 'relation' to refer to an ordered, binary relation between entities. We refer to individual ordered pairs in this relation as 'relation instances'. For example, the person-nationality relation holds between the entities named 'John Steinbeck' and 'United States', so it has <John Steinbeck, United States> as an instance.

We use relations and relation instances from Freebase, a freely available online database of structured semantic data. Data in Freebase is collected from a variety of sources. One major source is text boxes and other tabular data from Wikipedia. Data is also taken from NNDB (biographical information), MusicBrainz (music), the SEC (financial and corporate data), as well as direct, wiki-style user editing. After some basic processing of the July 2008 link export to convert Freebase's data representation into binary relations, we have 116 million instances of 7,300 relations between 9 million entities. We next filter out nameless and uninteresting entities such as user profiles and music tracks. Freebase also contains the reverses of many of its relations (book-author v. author-book), and these are merged. Filtering and removing all but the largest relations leaves us with 1.8 million instances of 102 relations connecting 940,000 entities. Examples are shown in Table 2.

#### Architecture

The intuition of our distant supervision approach is to use Freebase to give us a training set of relations and entity pairs that participate in those relations. In the training step, all entities are identified in sentences using a named entity tagger that labels persons, organizations and locations. If a sentence contains two entities and those entities are an instance of one of our Freebase relations, features are extracted from that sentence and are added to the feature vector for the relation.

The distant supervision assumption is that if two entities participate in a relation, any sentence that contain those two entities might express that relation. Because any individual sentence may give an incorrect cue, our algorithm trains a multiclass logistic regression

classifier, learning weights for each noisy feature. In training, the features for identical tuples (relation, entity1, entity2) from different sentences are combined, creating a richer feature vector.

In the testing step, entities are again identified using the named entity tagger. This time, every pair of entities appearing together in a sentence is considered a potential relation instance, and whenever those entities appear together, features are extracted on the sentence and added to a feature vector for that entity pair. For example, if a pair of entities occurs in 10 sentences in the test set, and each sentence has 3 features extracted from it, the entity pair will have 30 associated features. Each entity pair in each sentence in the test corpus is run through feature extraction, and the regression classifier predicts a relation name for each entity pair based on the features from all of the sentences in which it appeared.

Consider the location-contains relation, imagining that in Freebase we had two instances of this relation: <Virginia, Richmond> and <France, Nantes>. As we encountered sentences like ‘Richmond, the capital of Virginia’ and ‘Henry’s Edict of Nantes helped the Protestants of France’ we would extract features from these sentences. Some features would be very useful, such as the features from the Richmond sentence, and some would be less useful, like those from the Nantes sentence. In testing, if we came across a sentence like ‘Vienna, the capital of Austria’, one or more of its features would match those of the Richmond sentence, providing evidence that <Austria, Vienna> belongs to the location-contains relation.

Note that one of the main advantages of our architecture is its ability to combine information from many different mentions of the same relation. Consider the entity pair <Steven Spielberg, Saving Private Ryan> from the following two sentences, as evidence for the film-director relation.

[Steven Spielberg]’s film [Saving Private Ryan] is loosely based on the brothers’ story.

Allison co-produced the Academy Awardwinning [Saving Private Ryan], directed by [Steven Spielberg]...

The first sentence, while providing evidence for film-director, could instead be evidence for film-writer or film-producer. The second sentence does not mention that Saving Private Ryan is a film, and so could instead be evidence for the CEO relation (consider ‘Robert Mueller directed the FBI’). In isolation, neither of these features is conclusive, but in combination, they are.

#### Features

Our features are based on standard lexical and syntactic features from the literature. Each feature describes how two entities are related in a sentence, using either syntactic or non-

syntactic information.

#### Lexical features

Our lexical features describe specific words between and surrounding the two entities in the sentence in which they appear:

- The sequence of words between the two entities
- The part-of-speech tags of these words
- A flag indicating which entity came first in the sentence
- A window of  $k$  words to the left of Entity 1 and their part-of-speech tags
- A window of  $k$  words to the right of Entity 2 and their part-of-speech tags

Each lexical feature consists of the conjunction of all these components. We generate a conjunctive feature for each  $k \in \{0, 1, 2\}$ . Thus each lexical row in Table 3 represents a single lexical feature.

Part-of-speech tags were assigned by a maximum entropy tagger trained on the Penn Treebank, and then simplified into seven categories: nouns, verbs, adverbs, adjectives, numbers, foreign words, and everything else.

In an attempt to approximate syntactic features, we also tested variations on our lexical features: (1) omitting all words that are not verbs and (2) omitting all function words. In combination with the other lexical features, they gave a small boost to precision, but not large enough to justify the increased demand on our computational resources.

#### Syntactic features

In addition to lexical features we extract a number of features based on syntax. In order to generate these features we parse each sentence with the broad-coverage dependency parser MINIPAR (Lin, 1998).

A dependency parse consists of a set of words and chunks (e.g.

‘Edwin Hubble’, ‘Missouri’, ‘born’), linked by directional dependencies (e.g. ‘pred’, ‘lex-mod’), as in Figure 1. For each sentence we extract a dependency path between each pair of entities. A dependency path consists of a series of dependencies, directions and words/chunks representing a traversal of the parse. Part-of-speech tags are not included in the dependency path.

Our syntactic features are similar to those used in Snow et al. (2005). They consist of the conjunction of:

- A dependency path between the two entities
- For each entity, one ‘window’ node that is not part of the dependency path

A window node is a node connected to one of the two entities

and not part of the dependency path. We generate one conjunctive feature for each pair of left and right window nodes, as well as features which omit one or both of them. Thus each syntactic row in Table 3 represents a single syntactic feature.

#### Named entity tag features

Every feature contains, in addition to the content described above, named entity tags for the two entities. We perform named entity tagging using the Stanford four-class named entity tagger (Finkel et al., 2005). The tagger provides each word with a label from (person, location, organization, miscellaneous, none).

#### Feature conjunction

Rather than use each of the above features in the classifier independently, we use only conjunctive features. Each feature consists of the conjunction of several attributes of the sentence, plus the named entity tags. For two features to match, all of their conjuncts must match exactly. This yields low-recall but high-precision features. With a small amount of data, this approach would be problematic, since most features would only be seen once, rendering them useless to the classifier. Since we use large amounts of data, even complex features appear multiple times, allowing our high precision features to work as intended. Features for a sample sentence are shown in Table

### 3. Implementation

#### Text

For unstructured text we use the Freebase Wikipedia Extraction, a dump of the full text of all Wikipedia articles (not including discussion and user pages) which has been sentence-tokenized by Metaweb Technologies, the developers of Freebase (Metaweb, 2008). This dump consists of approximately 1.8 million articles, with an average of 14.3 sentences per article. The total number of words (counting punctuation marks) is 601,600,703. For our experiments we use about half of the articles: 800,000 for training and 400,000 for testing.

We use Wikipedia because it is relatively up-to-date, and because its sentences tend to make explicit many facts that might be omitted in newswire. Much of the information in Freebase is derived from tabular data from Wikipedia, meaning that Freebase relations are more likely to appear in sentences in Wikipedia.

#### Parsing and chunking

Each sentence of this unstructured text is dependency parsed by MINIPAR to produce a dependency graph.

In preprocessing, consecutive words with the same named entity tag are ‘chunked’, so that Edwin/PERSON Hubble/PERSON becomes [Edwin Hubble]/PERSON. This chunking is



restricted by the dependency parse of the sentence, however, in that chunks must be contiguous in the parse (i.e., no chunks across subtrees). This ensures that parse tree structure is preserved, since the parses must be updated to reflect the chunking. Training and testing

For held-out evaluation experiments (see section 7.1), half of the instances of each relation are not used in training, and are later used to compare against newly discovered instances. This means that 900,000 Freebase relation instances are used in training, and 900,000 are held out. These experiments used 800,000 Wikipedia articles in the training phase and 400,000 different articles in the testing phase.

For human evaluation experiments, all 1.8 million relation instances are used in training. Again, we use 800,000 Wikipedia articles in the training phase and 400,000 different articles in the testing phase.

For all our experiments, we only extract relation instances that do not appear in our training data, i.e., instances that are not already in Freebase. Our system needs negative training data for the purposes of constructing the classifier. Towards this end, we build a feature vector in the training phase for an ‘unrelated’ relation by randomly selecting entity pairs that do not appear in any Freebase relation and extracting features for them. While it is possible that some of these entity pairs are in fact related but are wrongly omitted from the Freebase data, we expect that on average these false negatives will have a small effect on the performance of the classifier. For performance reasons, we randomly sample 1% of such entity pairs for use as negative training examples. By contrast, in the actual test data, 98.7% of the entity pairs we extract do not possess any of the top 102 relations we consider in Freebase.

We use a multi-class logistic classifier optimized using L-BFGS with Gaussian regularization. Our classifier takes as input an entity pair and a feature vector, and returns a relation name and a confidence score based on the probability of the entity pair belonging to that relation. Once all of the entity pairs discovered during testing have been classified, they can be ranked by confidence score and used to generate a list of the  $n$  most likely new relation instances.

Table 4 shows some high-weight features learned by our system.

We discuss the results in the next section.

### Evaluation

We evaluate labels in two ways: automatically, by holding out part of the Freebase relation data during training, and comparing newly discovered relation instances against this held-out data, and manually, having humans who look at each positively labeled entity pair and mark whether the relation indeed holds between the participants. Both evaluations allow us to

calculate the precision of the system for the best N instances.

#### Held-out evaluation

Figure 2 shows the performance of our classifier on held-out Freebase relation data. While held-out evaluation suffers from false negatives, it gives a rough measure of precision without requiring expensive human evaluation, making it useful for parameter setting.

At most recall levels, the combination of syntactic and lexical features offers a substantial improvement in precision over either of these feature sets on its own.

#### Human evaluation

Human evaluation was performed by evaluators on Amazon’s Mechanical Turk service, shown to be effective for natural language annotation in Snow et al. (2008). We ran three experiments: one using only syntactic features; one using only lexical features; and one using both syntactic and lexical features. For each of the 10 relations that appeared most frequently in our test data (according to our classifier), we took samples from the first 100 and 1000 instances of this relation generated in each experiment, and sent these to Mechanical Turk for human evaluation. Our sample size was 100.

Each predicted relation instance was labeled as true or false by between 1 and 3 labelers on Mechanical Turk. We assigned the truth or falsehood of each relation according to the majority vote of the labels; in the case of a tie (one vote each way) we assigned the relation as true or false with equal probability. The evaluation of the syntactic, lexical, and combination of features at a recall of 100 and 1000 instances is presented in Table 5.

At a recall of 100 instances, the combination of lexical and syntactic features has the best performance for a majority of the relations, while at a recall level of 1000 instances the results are mixed. No feature set strongly outperforms any of the others across all relations.

#### Discussion

Our results show that the distant supervision algorithm is able to extract high-precision patterns for a reasonably large number of relations.

The held-out results in Figure 2 suggest that the combination of syntactic and lexical features provides better performance than either feature set on its own. In order to understand the role of syntactic features, we examine Table 5, the human evaluation of the most frequent 10 relations. For the top-ranking 100 instances of each relation, most of the best results use syntactic features, either alone or in combination with lexical features. For the top-ranking 1000 instances of each relation, the results are more mixed, but syntactic features still helped in most classifications.

We then examine those relations for which syntactic features seem to help. For example,

syntactic features consistently outperform lexical features for the director-film and writer-film relations. As discussed in section 4, these two relations are particularly ambiguous, suggesting that syntactic features may help tease apart difficult relations. Perhaps more telling, we noticed many examples with a long string of words between the director and the film:

Back Street is a 1932 film made by Universal Pictures, directed by John M. Stahl, and produced by Carl Laemmle Jr.

Sentences like this have very long (and thus rare) lexical features, but relatively short dependency paths. Syntactic features can more easily abstract from the syntactic modifiers that comprise the extraneous parts of these strings.

Our results thus suggest that syntactic features are indeed useful in distantly supervised information extraction, and that the benefit of syntax occurs in cases where the individual patterns are particularly ambiguous, and where they are nearby in the dependency structure but distant in terms of words. It remains for future work to see whether simpler, chunk-based syntactic features might be able to capture enough of this gain without the overhead of full parsing, and whether coreference resolution could improve performance.

#### Acknowledgments

We would like to acknowledge Sarah Spikes for her help in developing the relation extraction system, Christopher Manning and Mihai Surdeanu for their invaluable advice, and Fuliang Weng and Baoshi Yan for their guidance. Our research was partially funded by the NSF via award IIS-0811974 and by Robert Bosch LLC.

## 附录 B 中文翻译

### 摘要

现在关系抽取的方法如 ACE 是基于监督型学习方法，从少量人工标记的语料库中抽取关系。我们研究了一个不需要标注的语料库的方法，来避免对 ACE 类型算法的领域依赖问题，此系统也可用于其他大小的语料库。我们的研究使用了 Freebase，一个大型的包含徐哥种类关系的数据库，来提供距离监督。对于每个出现在 Freebase 的实体对，我们发现一个未标注的语料库中包含大量实体对，抽取其文本特征来训练关系分类器。我们算法包含了监督型 IE 系统（包含 400000 噪音数据的模板特征）的优点和非监督型 IE 系统（在所有领域中抽取大量关系）。我们模型能抽取 10000 个实例，包含 102 种关系，准确达到 67.6%。我们也分析了特征性能，表明句法分析特征对关系抽取有用。

### 引言

至少三种学习方法被应用在文本关系抽取任务中（例如，学习一个人被特定组织雇佣，特定地区的地理信息）。

在监督性学习领域，语料库中的句子是表述实体和关系的被人工标记的资料。以 NIST 自动内容抽取 (ACE) RDC 2003 和 2004 语料库为例，其包含超过 1000 篇文档中的实体对被标注为 5 到 7 中主要的关系类型和 23 到 24 种子关系，总共 16771 个关系实例。

ACE 系统抽取了一个广泛的词法、句法和语义特征并使用监督型分类方法标注了在测试数据集中给定实体对的关系 (Zhou et al., 2005; Zhou et al., 2007; Surdeanu and Ciaramita, 2007)。

然而监督型关系抽取依然面临许多问题。标记训练数据十分重要，也限制了最终的质量。同样，由于关系被标记于同一个特定语料库中，分类器的结果也倾向于语料库本身的文本领域。

完全使用非监督信息抽取的一个可选的方法是抽取大量文本实体间的词串，再进行聚类 and 简化词串来生成关系串 (Shinyama and Sekine, 2006; Banko et al., 2007)。非监督方法可以使用大量数据并抽取更多的实体，但其结果不能简单映射到一个特定知识库的关系中。

第三种方法使用少量的种子实例或模板来做 bootstrap 学习 (Brin, 1998; Riloff and Jones, 1999; Agichtein and Gravano, 2000; Ravichandran and Hovy, 2002; Etzioni et al., 2005; Pennacchiotti and Pantel, 2006; Bunescu and Mooney, 2007; Rozenfeld and Feldman, 2008)。这些种子使用大量语料库来抽取一些新的模板，这些模板用来抽取更多的实例，再使用这些实例抽取更多模板，循环迭代。其结果通常准确率比较低，也容易出现语义漂移现象。

象。

我们提出了一个可选方法距离监督,其结合了这些方法的优点。距离监督是由 Snow et al. (2005) 提出的方法的扩展,其方法用在 WordNet 上抽取实体关系的上位词 (is-a), 其使用了实体间的生物关系学的弱标记数据 (Craven and Kumlien, 1999; Morgan et al., 2004)。

我们的算法使用 Freebase (Bollacker et al., 2008) 提供了关系抽取的距离监督方法, Freebase 是一个大型语料数据库。Freebase 包含 9 百万实体对的 7300 关系共 116 百万实例。距离监督的本质是使用一个包含 Freebase 里已知关系的实体对的句子。由于可能有许多句子包含给定实体对,我们能抽取大量的特征(可能有噪音)用于逻辑回归分类器。

因此虽然监督型训练集使用了少量语料库中标记数据作为训练数据,我们的算法仍可使用大量的数据:文本,关系,实例。我们使用 1.2 百万 Wikipedia 文章和 940000 个实体的 102 种关系的 1.8 百万实例。另外,在分类器中结合了大量特征帮助我们避免无效特征。

因为我们的算法依赖于一个数据库,而不是标记文本,它并不存在过拟合和领域依赖的问题。这也意味着不同于非监督型方法,我们分类器的输出使用了典型的关系名。我们的方法提供了在多个句子中集成数据的方法来确定是否实体对间存在某个关系。因为我们算法可以使用大量非标记文本,实体对在一个测试集中可能出现多次。对于每个实体对,我们合并了不同句子的特征,使用一个实体向量表示,给分类器提供更多的信息来精确我们的标记。

表 1 是一个由我们系统抽取的关系实例的例子,我们也使用了此系统调研了关系抽取句法和词法特征。我们已知句法特征能提高监督型的 IE 系统的性能,至少在使用人工标记的 ACE 数据中 (Zhou et al., 2007; Zhou et al., 2005), 我们不确定是否句法特征能提供非监督或者距离监督的 IE 系统。为了避免解析需要的计算时间,当前在 bootstrapping 和非监督 IE 系统方面的研究只使用了监督的词法特征 (Brin, 1998; Agichtein and Gravano, 2000; Etzioni et al., 2005)。少量使用非监督信息抽取的系统并没有比较两种特征的优劣。

以前的工作

除了以上已讨论过的非监督型算法,关系抽取的以前监督型方法或者 bootstrap 方法依赖于相对小的数据集,或者只使用了少量的关系。基于 WordNet 的方法通常只关注于属于 (is-a) 包含 (part-of) 关系 (Girju et al., 2003; Snow et al., 2005), 基于 ACE 的方法限制于少量的关系实例语料库也少于一百万字。

许多早期关系抽取算法使用很少甚至是没有句法信息。例如, Brin 的 DIPRE 算法使用了基于字符串的正则表达式来识别关系如作者-书籍, Agichtein 和 Gravano 的

SNOWBALL 算法使用相似的字符和命名实体间的正则表达式。Hearst 使用了少量字符和词性标记间的正则表达式来找到新实例。基于模式匹配的方法已经被用于许多成功的系统中，例如 Etzioni et al. (2005)。其他工作如 Ravichandran and Hovy (2002) 和 Pantel and Pennacchiotti (2006) 使用相同的字符和词性标记的正则表达式来发现关系并抽取实例。

更多近期的研究已使用了一些更深层次的输入语句解析后的句法信息，包括 Lin and Pantel (2001) 和 Snow et al. (2005) 对句法依赖的研究，和 Zhou et al. (2005) 和 Zhou et al. (2007) 在 ACE 方法上的工作。

与我们距离监督算法更相似的研究是 Wu and Weld (2007) 从 wikipedia 中使用页面信息的监督方法抽取关系的研究。不同于他们的关注与一个 wikipedia 的方法，我们算法可以从更多不同的问答不同种类数据中抽取关系信息。

#### Freebase

由文学语言的原因，我们使用“关系”这个词来表示有序的二元实体关系。我们将独立的有序关系对称为“关系实例”。例如，人-民族关系中的一个实体对“John Steinbeck”和“United States”，其表示方式为 <John Steinbeck, United States>。

我们的关系和关系实例来源于 Freebase，它是一个免费开源的在线结构化数据库。Freebase 的数据有许多来源。其中最重要的是来自 wikipedia 的文本和表格数据。其中也包含来自 NNDB（生物信息）、MusicBrainz（音乐），SEC（经济和企业数据）以及使用者直接编辑的数据。经过 2008 九月的一些基本处理将 Freebase 的数据转化为二元关系，我们有了 116 百万的关系实例，其中包括 7300 种关系和 9 百万实体。我们将过滤出无用或不感兴趣的实体例如用户信息和音乐页面。Freebase 也包含了许多关系（书 -作者或者作者 -书），这些关系也被合并了。过滤并重建数据后我们仍有 102 种关系 940000 中实体的 1.8 百万的实例。表 2 是一些例子。

#### 架构

距离监督方法使用 Freebase 来提高一个关系和实体对的训练集。在训练阶段，句子中的所有实体使用命名实体标记，命名实体标签包括人，组织和地点。如果一个句子包括两种实体并且这个实体是我们 Freebase 的一个实例，就将从句子抽取其特征加入到关系的特征向量中。

如果两个实体对属于一个关系，所有包含两个实体对的句子都将表述这一关系。因此句子可能有错误，我们算法训练了多分类逻辑回归分类器，学习噪音特征的权重。在训练阶段，来自不同句子的标记的实体对（关系，实体 1，实体 2）的特征将被合并变成一个更全面的特征向量。

在测试阶段，实体将再次使用命名实体来标记。这次每个出现在一个句子中的实体对将被认为一个可能的关系实例，只用两个实体出现在一起，就将抽取句子特征并添加

到实体对的特征向量中，例如，如果一个实体对出现在一个特征集的 10 个句子中，并且每个句子有 3 个特征向量，那么这个实体对将有 30 合并向量。测试集中的每个句子的实体对都会进行特征抽取，回归分类器将根据所有出现的句子特征给一个实体对预测一个关系名字。

以位置 -包含关系为例，加黑色在 Freebase 中有两个实例 <Virginia, Richmond> 和 <France, Nantes>。有两个句子 “Richmond, the capital of Virginia” 和 “Henry’s Edict of Nantes helped the Protestants of France” 需要抽取特征。一些特征是有用的，例如 Richmond 这个句子的特征，但一些特征是无用的，如 Nantes 的例子。在测试阶段，如果有一个句子如 “Vienna, the capital of Austria”，它将有一个或多个特征匹配 Richmond 这个句子，并抽取出 <Austria, Vienna> 这个实体对。

需要注意我们架构最主要的优势是其合并相同关系的不同信息的能力。例如一个实体对 <Steven Spielberg, Saving Private Ryan> 来自于下面两个电影 -导演关系的句子。

[Steven Spielberg]’s film [Saving Private Ryan] is loosely based on the brothers’story.

Allison co-produced the Academy Awardwinning [Saving Private Ryan], directed by [Steven Spielberg]...

第一个句子表达了电影 -导演关系，可以被替代为电影 -作者或者电影 -出品人关系。第二个句子并没有提到 Saving Private Ryan 是一个电影，也可以被 CEO 关系代替（如 “Robert Mueller directed the FBI”）。总而言之，没有一种关系是唯一的，但通过合并数据，它们则可被唯一标识。

### 特征

本文特征基于标准的文学意义上的词法和句法特征。每一种特征描述了一个句子中两个相关的实体如何被关联起来。

#### 词法特征

我们的词法特征描述了同一个句子中在两个实体之间或周围的特定词语：

- 实体间的词语序列
- 词语的 POS 标记
- 实体的前后顺序
- 实体 1 前 K 个词和他们的 POS 标记
- 实体 2 后 K 个词和他们的 POS 标记

每个词法特征包含了上述所有特征，我们可得到  $K \in \{0,1,2\}$  的一个联合特征。如表 3 所示。

词性标记使用利用 Penn Treebank 训练的最大熵标注器，并被简化为 7 中类型：名词，动词，副词，数词，外文和其他词。

在试图得到词法特征的过程中，我们也测试了其他词法特征：(1) 省略所有非动词 (2) 省略所有虚词。在和其他词法特征合并时，他们对精确度有少量提高，但不足以弥补计算资源的问题。

#### 句法特征

除了词法特征，我们也抽取了句法特征。为了生成这些特征，我们使用 MINIPAR(Lin, 1998) 提供的依赖解析器解析了每个句子。

依赖解析结果包括一组词和块（例如 ‘Edwin Hubble’，‘Missouri’，‘born’），和直接依赖关系（如 ‘pred’，‘lex-mod’），如图 1 所示。对于每种句子我们抽取了一个实体对间的依赖路径。依赖路径包含了一系列的依赖，指向和词/块信息。词性标注结果将不再包含在依赖路径中。

我们句法特征和 Snow et al. (2005) 使用的相似。它包括：

- 实体对间的依赖关系
- 每个实体对，一个不再依赖路径的 ‘window’ 结点

‘window’ 结点是连接到两个实体对但并在依赖路径的结点。我们将每个实体对左右的 window 结点联合，每行句法分析结果如表 3 所示。

#### 命名实体特征

除了上面所述，每个特征还包括实体的命名实体标签。我们使

用 Stanford(Finkel et al., 2005) 提供的命名实体标注器。标注器提供了 (人，地点，组织，杂项，无) 几种标签。

#### 特征合并

并非将上述特征独立用于分类器，我们使用了联合特征。每种特征包含了句子多个属性的联合，加上命名实体标签。两个特征要想匹配，他们所有的信息必须都匹配，这样会带来低召回高精度率。数据越少，这种方法越有问题，因为大量特征可能只用了一次，其对分类器是无用的。由于我们使用了大量数据，复杂特征甚至会出现很多次，使用可以得到高精度率的数据。特征如表 3 所示。

#### 实现文本

我们使用了 Freebase 和 wikipedia 的非结构化数据。包括近 1.8 百万篇文章，每篇文章平均 14.3 个句子。总字数为 601600703。为了研究我们使用了一般的文章：800000 用于训练，400000 用于测试。

我们使用 Wikipedia 因为他数据很新，句子倾向于现在的事实。

Freebase 里大量信息来源于 wikipedia，也意味着 freebase 关系可能出现在 wikipedia 中。

#### 解析和分块

每个非结构化的句子解析依赖于 MINIPAR 来产生依赖图。



数据处理时，相同的命名实体标签会被分块，如 Edwin/PERSON

Hubble/PERSON 会变成 [Edwin Hubble]/PERSON. 分块会限制句子的依赖关系，但分块必须是在同一个解析树中（不能跨子树）。这确保了解析树结构被保证下来，因为解析结果会给分块更新。

#### 训练和测试

每个关系有一半的实例在训练是不会被使用，之后将用于比较新发现的实例。这意味着 900000 个 Freebase 的关系实例被用于训练，900000 将被移出。实验在训练阶段使用 800000 篇 wikipedia 文摘，在测试阶段 400000 篇不同的文章。

实验时，1.8 百万干系实例被用于训练。实验在训练阶段使用

800000 篇 wikipedia 文摘，在测试阶段 400000 篇不同的文章。

我们实验中抽取的关系实例不会再出现在训练集中，如实例不

会出现在 Freebase 中。我们系统训练数据需要反例来构建分类器。我们在训练集中构建了一个特征向量，随机选择没出现在 Freebase 中的实体对并抽取特征。可能没出现在 Freebase 中的实体对知识被忽略了，我们期望这样的反例会对分类器有一个小的影响。由于性能原因，我们随机选择了 1% 的数据使用反例训练，而在测试集中与之相反的，我们抽取的 98.7% 的实体对并不属于 Freebase 的前 102 个关系。

逻辑回归分类器参数使用 L-BFGS 和高斯正则化。分类器将实体对和特征向量作为输入，返回一个关系名和实体对属于此关系的可能性得分。一旦测试阶段所有被发现的实体对被分类结束，将会按照置信度得分排序生成前  $n$  个最有可能是新关系的列表。

表 4 是得到的权重最高的特征。我们将在下一章讨论这个结果。

#### 评估

我们使用了两种方式评估：自动的方式是训练时移出一部分 Freebase 关系数据，用移出的部分数据比较新发现的关系实例；人工的方式是使用标记的实体对标注其关系是否正确。两种评估方式帮助我们计算系统的准确率。

#### 移出评估

图 2 是我们分类器使用移出评估方法的性能结果。因为移出评估存在错误可能，它在没有人工评估的情况下给我们了一个初步的评价，对于参数设定是非常有用的。

召回率最好的地方，句法和词法的联合提供了准确率的提高。

#### 人工评估

人工评估使用了亚马逊的 Mechanical Turk 服务，其在自然语言处理中非常有用。我们使用了 3 词实验：一是只使用句法特征；意识只使用词法特征；一是使用两者。通过我们的测试（根据我们的分类器），每次实验我们选择了前 100 到 1000 个实例，并通过 Mechanical Turk 分发进行人工评估。每次预估实例由 1 到 3 个标记员标记为真或假。我们统计了平均结果，词法特征、句法特征和合并特征结果见表 5。

100 个实例中，句法特征和词法特征的合并特征是性能最好的，

虽然在 1000 个实例时这个结果是混合了得。但没有一种特征集合可以在所有关系中都表现很好。

### 讨论

我们的结果表示距离监督算法能够在大量关系中抽取高准确率的答案。

图 2 移出评估的结果表明词法和句法特征的结果能提供一个更好的性能。为了理解句法特征的角色，我们使用了人工评估结果见表 5. 对于前 100 的数据，最好的结果是使用句法特征，而不是合并特征。但前 1000 个关系实例，结果是混合的，但是句法特征在分类器中依旧有效。

然后我们研究句法特征有效的关系。举个例子，在导演 -电影和作者 -电影关系上句法分析始终比词法分析结果好。这两个关系非常相似，因此句法分析可能能保住区分复杂关系。我们注意到在电影和导演这类句子中许多例子有很多词语：

**Back Street is a 1932 film made by Universal Pictures, directed  
by John M. Stahl, and produced by Carl Laemmle Jr.**

很长的句子有很长的词法特征，但只有很短的依赖路径。句法分析能抽象出句法们俩去来解析这些句子。

我们的结果表明句法特征在距离监督信息抽取中十分有用，句法的最佳运用是在十分含糊的关系中。未来的工作是研究是否简单的，分块的句法特征可以朴拙到最佳解析结果，并且是否共指消解可以提高性能。

### 致谢

我们非常感谢 Sarah Spikes 在开发关系抽取系统中的帮助，以及 Christopher Manning 和 Mihai Surdeanu 宝贵的建议，和 FuliangWeng 、Baoshi Yan 的指导。