

分类号 TP391.1

密 级

UDC 621.3

学校代码 10500



湖北工业大学  
HUBEI UNIVERSITY OF TECHNOLOGY

# 硕士学位论文

(学历教育-专业学位)

题 目：用于网络舆情分析的深度学习自然语言处理系统

英文题目：Deep Learning Natural Language Processing System  
for Network Public Opinion Analysis

学位申请人姓名：王仲昊

申请学位学科专业：电气工程

指导教师姓名：万相奎

二〇二〇年六月

分类号 TP391.1

密 级                     

UDC 621.3

学校代码 10500



**湖北工业大学**  
HUBEI UNIVERSITY OF TECHNOLOGY

# 硕士学位论文

题 目 用于网络舆情分析的深度学习自然语言处理系统

英文题目 Deep Learning Natural Language Processing System

for Network Public Opinion Analysis

研究生姓名（签名） 王仲良

指导教师姓名（签名） 万松全 职 称 教授

申请学位学科名称 电气工程 学科代码 080805

论文答辩日期 2020年6月14日 学位授予日期 2020年6月14日

学院负责人（签名） 张晓曼

评阅人姓名 苏义鑫 评阅人姓名 宋斌

2020年6月30日

# 湖北工业大学

## 学位论文原创性声明和使用授权说明

### 原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的科研成果。除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本声明的法律结果由本人承担。

学位论文作者签名：



日期：2020年6月30日

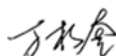
### 学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权湖北工业大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

学位论文作者签名：



指导教师签名：



日期：2020年6月30日

日期：2020年6月30日

## 摘 要

随着互联网的高速发展, 网络社交信息爆炸式增长的同时也带来了网络舆情分析的问题。传统的网络舆情分析模式采用的是词库的方式, 语料直接与词库进行比对后进行判断。这种方式由于中文的复杂性, 例如存在近音词、同义词、缩略词、暗语等非规范中文表达, 使得舆情分析的效果不佳。结合深度学习来对语料进行处理, 可以有效的提高对非规范中文表达进行分析时结果的准确性。

本文根据这一方法, 深入研究基于深度学习的自然语言处理, 以求在分析自然语言的词相似性中得到更准确的结果, 并结合这一方法开发用于网络舆情分析的深度学习自然语言处理系统, 本文研究内容主要包括以下部分。

基于 python 的 Scrapy 网络爬虫研究。本系统将使用网络爬虫获取网络实时语料数据保证数据库的时效性, 通过这种方式可以有效的提高自然语言处理对非规范性语言的覆盖程度, 提高分析的效果;

搭建语料数据库服务器。本文在处理数据的过程中需要不断地更新现有的语料库, 所以需要搭建语料数据库用于存储实时的语料数据, 并在数据库中完成对语料数据的初步处理, 通过正则表达式和分词得到可以用于深度学习的数据;

基于 TensorFlow 的自然语言处理深度学习算法设计与实现。本文采用了一种动态权重多模型相融合的词相似性分析方法, 根据语料的特点选取不同的语料库, 并结合多种模型进行计算, 提高词相似性分析的准确性, 使得自然语言处理所得到的结果对网络舆情分析有更好的支持度, 本文通过实验发现多模型相融合的方法得到的结果比单一模型更好, 在使用 NLPCC-ICCPOL 2016 中文词语相似度比赛中 PKU-500 数据集作为评价的参考标准时, 本文所采用动态权重多模型融合的词相似性分析法, 获得 0.568 的斯皮尔曼等级相关系数, 与该比赛第一名的结果相比提高了 9.6%, 因此多模型相融合的方法可以提高计算词相似性时的准确率;

整合以上各部分构建网络舆情分析系统。搭建出的网络舆情分析系统将实现自动实时收集网络语料并加入语料库进行深度学习计算, 不断更新计算结果, 提高网络舆情分析系统的时效性, 同时提供词相似性查询功能, 使用者可以通过该系统直接得到两词相似性的量化结果。

**关键词:** 舆情分析, 自然语言处理, 深度学习, 动态权重

## Abstract

With the rapid development of the Internet, the explosive growth of online social information has also brought the problem of online public opinion analysis. The traditional network public opinion analysis mode uses the method of thesaurus, and the corpus is directly compared with the thesaurus for judgment. Due to the complexity of Chinese, such as the presence of non-standard Chinese expressions such as near-tone words, synonyms, acronyms, and cryptic words, the effect of public opinion analysis is poor. Combining deep learning to process the corpus can effectively improve the accuracy of the results when analyzing non-standard Chinese expressions.

Based on this method, this paper deeply researches natural language processing based on deep learning, and applies deep learning to natural language processing in order to obtain more accurate results in analyzing the similarity of words in natural language, and combines this method to develop Deep learning natural language processing system for network public opinion analysis. The research content of this article mainly includes the following parts.

Research on Scrapy web crawler based on python. The article will use web crawlers to obtain real-time corpus data from the network to ensure the timeliness of the database. In this way, the coverage of natural language processing on non-standard languages can be effectively improved, and the analysis effect can be improved;

Build a corpus database server. This paper needs to constantly update the existing corpus in the process of processing data, so it is necessary to build a corresponding corpus data server to store real-time corpus data and complete the preliminary processing of corpus data in the database. Segmentation to get data that can be used for deep learning;

Design and Implementation of Deep Learning Algorithms for Natural Language Processing Based on TensorFlow. This article adopts a method of word similarity analysis based on the fusion of dynamic weights and multiple models. It selects different corpora according to the characteristics of the corpus and combines multiple models to perform calculations to improve the accuracy of word similarity analysis. The obtained results have better support for online public opinion monitoring. In this article, it is found through experiments that the multi-model fusion method has better results than the single model. The PKU-500 dataset was used in the NLPCC-ICCPOL 2016 Chinese Word Similarity Competition. As a reference standard for evaluation, the word similarity analysis method of the dynamic weight multi-model fusion adopted in this system obtained a Spearman rank correlation coefficient of 0.568, which was an increase of 9.6% compared to the

result of the first place in the competition, so the multi-model The fusion method can improve the accuracy when calculating word similarity;

Integrate the above parts to build a network public opinion analysis system. The built network public opinion analysis system will automatically collect real-time network corpora and join corpora for deep learning calculations, continuously update the calculation results, improve the timeliness of the network public opinion analysis system, and provide word similarity query functions. Users can use this system Get the quantitative result of the similarity of two words directly.

**Keywords:** public opinion analysis, natural language processing, deep learning, dynamic weighting

## 目 录

摘 要.....	I
Abstract.....	II
目 录.....	IV
第 1 章 引 言.....	1
1.1 课题的研究背景及意义.....	1
1.2 国内外研究现状.....	1
1.3 本文的研究内容.....	4
1.4 本文章节安排.....	5
第 2 章 网络舆情分析系统的设计.....	6
2.1 网络舆情分析系统设计的理论分析.....	6
2.2 网络舆情分析系统的结构设计.....	7
2.2.1 语料收集与预处理子系统设计.....	8
2.2.2 自适应自然语言处理量化子系统设计.....	10
2.3 词相似性评测数据集.....	15
2.4 本章小结.....	16
第 3 章 动态权重多模型相融合的词相似性算法.....	17
3.1 动态权重多模型相融合的词相似性算法的设计.....	17
3.1.1 统计模型的选择.....	17
3.1.2 词典模型的选择.....	17
3.1.3 动态权重多模型融合的方法.....	18
3.1.4 算法的整体结构.....	19
3.2 计算过程与结果分析.....	21
3.2.1 计算结果的评价标准.....	21
3.2.2 计算对比试验的方案.....	22
3.2.3 统计模型的词相似性计算.....	22
3.2.4 词典模型的词相似性计算.....	23
3.2.5 简单权重多模型融合的词相似性计算.....	24
3.2.6 动态权重多模型融合的词相似性计算.....	25
3.3 本章小结.....	25

# 湖北工业大学硕士学位论文

第 4 章 网络舆情分析系统的实现 .....	26
4.1 网络舆情分析系统需求分析 .....	26
4.2 网络舆情分析系统的实现过程 .....	26
4.2.1 网络爬虫设计与搭建 .....	26
4.2.2 数据库设计与数据的收集处理 .....	29
4.2.3 词相似性分析设计与结果量化的实现 .....	30
4.2.4 可视化界面设计与实现 .....	32
4.3 系统测试与结果分析 .....	34
4.4 本章小结 .....	37
第 5 章 总结与展望 .....	38
5.1 全文总结 .....	38
5.2 研究展望 .....	39
参考文献 .....	40
致 谢 .....	43



## 第1章 引言

### 1.1 课题的研究背景及意义

随着互联网的迅速发展,网络信息的舆情是了解互联网民意的重要风向标。舆情的分析技术可以帮助政府机构对民意进行分析,从而为制定政策提供参考,也可以了解热点事件的舆论走向,在满足人民舆论需求的情况下为事件的解决提供建议。同时,企业也可利用该技术来了解自身在民间的真实评价,市场动向,以及广告的实际效果等等。对舆情分析的合理疏导有利于社会稳定的同时,也可以给普通民众带来一个优良的网络环境。

社交媒体的信息爆炸标志着大数据时代的来临。然而,伴随着网络用户的各种亚文化圈子的形成,大量的非规范表达不断产生,这些文字的使用组成了庞大的网络中文语料库。这些非规范表达是中文表达中的主要组成部分,对中文自然语言处理的效果有至关重要的意义。现有的网络识别系统大多是基于词库的传统系统,这种系统对非规范表达识别的准确性比较差,如果依旧采用传统的基于词库的方法,这些非规范表达往往无法被准确识别出来,从而造成一些重要信息的丢失和误判,给自然语言处理舆情分析和相关任务带来很多问题和挑战。在网络信息爆炸式增长的时代,很多网络社交都使用非规范表达来进行交流,使得网络社交环境和网络舆情非常复杂。基于深度学习的网络舆情分析系统能够有效地分析出实际的网络舆情,所以这一系统的建立就显得非常重要。

本文中所建立的系统首先在语料库的收集上进行了改进,将网络中的规范表达和非规范表达都添加到了语料库中,提高了语料库对词汇的覆盖率,同时通过基于深度学习的自然语言处理方法,有效的提高了对各种网络表达方式分析的准确性,在分析网络舆情时可以根据上下文具体理解每个词在文本中的含义,避免受非规范表达的影响。最后将非规范表达添加到词库中,实现词库的自我更新,这样的舆论分析系统可以实现网络舆情准确而有效的分析。

### 1.2 国内外研究现状

舆情分析是国内外研究的一个重要课题,互联网爆炸式的发展使得互联网承载了巨大的信息,任何网络信息都可能在很短的时间内大范围的传播,造成舆论风暴,所以网络舆情的分析成为了重要任务。网络舆情分析最开始大多采用的是词库

模式，在互联网还不够普及的时代是一种有效的手段，但随着时代的发展，网络已走入千家万户，数据量大大增加的同时，网络交流方式也日新月异，传统词库模式既不能得到有效的舆情信息，甚至可能会获得错误的舆情信息，针对这种问题，研究人员希望将基于深度学习的自然语言处理应用于网络舆情分析中，实现更加准确高效的舆情分析。

自然语言处理（Natural Language Processing, NLP）是人工智能和语言学领域的分支学科。此领域探讨如何处理及运用自然语言；自然语言处理基本包括认知、理解、生成等部分。自然语言认知和理解是让电脑把获取的自然语言变成有意义的符号并建立相应的关系，然后再根据使用目的进行处理。近年来自然语言处理成为热门研究方向，其中机器翻译是自然语言处理最早的研究工作。自然语言处理的主要任务是研究表示语言能力和语言应用的模型，建立和实现计算框架并提出相应的方法不断地完善模型，以及实现用自然语言与计算机之间的通信。

自然语言处理最开始是国外的科学家以英文为基础来进行分析，自然语言处理的早期阶段，科学家们提出了基于语法规则的自然语言处理方法<sup>[1]</sup>，多数自然语言处理系统是以一套复杂且人工订定的规则为基础<sup>[2]</sup>。科学家们原本认为随着对自然语言语法概括得越来越全面以及计算机计算能力的提高，就可以解决自然语言处理的问题，但是基于语法规则的自然语言处理方法存在两个问题，第一是想要总结出能覆盖所有真实语句的规则几乎是不可能完成的任务，覆盖 20%的真实语句就需要至少几万条语法规则，即使能够写出涵盖所有自然语言现象的语法规则合集，也很难用计算机来进行处理，因为描述自然语言的文法和计算机高级程序语言的文法是不同的，其所需要的计算力和计算时间非常庞大。第二是自然语言中词的多义性很难用规则描述，部分多义词严重依赖上下文，而对于另一部分多义词，已经不能通过上下文来理解词的意思，需要通过常识来理解。

由于这些原因，基于规则的句法分析在上世纪 70 年代被基于统计的方法取而代之，以此让计算机处理自然语言的基本问题变成为自然语言这种上下文相关的特性建立数学模型<sup>[2]</sup>。基于统计的方法在判断一个句子是否合理时，只看其存在可能性的概率。基于统计的方法通过一个长度为  $n$  的句子  $S$  存在的可能性来判断一个句子是否合理，而在理解句子中某个词的含义时，则是计算该词在句中的位置上出现概率最大的词是什么来判断。通过统计学的方法，将计算机的自然语言处理问题从一个语言学问题变为了数学问题，在计算机计算力不断增长的今天，基于统计学的方法的识别准确率和计算效率已经大大提高。现今自然语言处理的主要策略都是采用的基于统计的方法，其广泛应用于机器翻译、语音识别，印刷体或手写体识别、拼写纠错和汉字输入等方面。

根据国外科学家们所提出的基于统计的方法，国内的科学家们也试图在汉语领域发展自然语言处理技术。汉语的自然语言处理有一个基本问题不同于英语，英语这种语言的词与词之间以空格分割，每个词的边界非常明确，但汉语不同，汉语除短句间使用标点符号分割外，短句的词之间并没有边界，这就给中文的自然语言处理带来了额外的麻烦，例如“乒乓球拍卖完了”这句话可以分割为“乒乓球”和“拍卖完了”，也可以分割为“乒乓球拍”和“卖完了”，这两种不同的分割方式得到完全不同的含义，所以中文分词是中文自然语言处理中一个不可或缺的任务。在处理中文分词的方法探讨中，中国的科学家同样也将基于统计模型的方法应用于其中，将某个句子由哪些词组成更合理这样一个语言学问题变成了某个句子由这些词组成的概率更高这样的数学问题，较好的解决了中文分词任务，为中文的自然语言处理打好了基础。在中文分词和统计模型的共同配合下，中文自然语言处理的准确率和效率如今也有了显著提高。

深度学习是机器学习的一个分支并且起源于机器学习，而机器学习是人工智能的一个分支，机器学习的研究是为了使计算机能够拥有自主学习的能力，目标是开发程序使计算机在处理不同的数据时能够自适应的进行变化来满足新数据的处理。机器学习试图通过大量数据来提升计算机的学习能力，发现数据的模式并变更计算行为。简而言之就是编写相应的算法使计算机从大量的数据中学习数据的规律，使计算机对新的样本数据可以智能识别或对未来的数据进行预判。机器学习的发展分为两个阶段，浅层学习（Shallow Learning）和深度学习（Deep Learning）。浅层学习起源于上世纪 20 年代的反向传播算法（Back-propagation），这使得基于统计的机器学习算法得到了广泛的应用。这时候的人工智能神经网络算法被称之为多层感知机（Multiple layer Perception），由于在当时多层网络训练十分困难，大多是只有一层隐含层的浅层模型。在机器学习的基础上，研究人员将神经网络这一概念融入机器学习而提出了深度学习，深度学习是机器学习中一种基于对数据进行表征学习的算法。深度学习的优势在于非监督或半监督的特征学习以及分层特征提取的高效算法来替代手工获取特征。表征学习的目标是寻求更好的表示方法并创建更好的模型来从大规模未标记数据中学习这些表示方法。表示方法由神经科学而来，构造类似神经网络中信息处理和通信模式的结构，例如神经编码，试图定义拉动神经元反应之间的关系以及大脑中的神经元电活动之间的关系。至今已有若干种深度学习的框架，如深度神经网络、卷积神经网络、深度置信网络和循环神经网络已被应用在计算机视觉、语音识别、自然语言处理、音频识别与生物信息学等领域并获取了极好的效果。

近些年，深度学习由于人工智能的发展成了一个令人瞩目的研究方向，已经有

科学家将之应用于自然语言处理。首先将机器学习算法引进到语言处理中，主要有两个原因：运算能力的稳定增大以及基于转换-生成文法的乔姆斯基语言学理论丧失主导。决策树等部分早期使用的机器学习算法是硬性的，“if-then”规则构成的系统是类似当时已存在的人工编写规则的系统。词性标记将隐马尔可夫模型引入 NLP，并且研究日益聚焦于软性的、以概率计算结果的统计模型，该模型的基础是将输入数据里每一个特征以分量代表。这种模型通常能够满足非预期输入数据的处理，尤其是输入有错误的情况，因为实际的数据无法避免包含错误的数据，并且在集成到包含多个子任务的较大系统时，可以得到比较可靠的结果。近年来 IBM 在研究机器翻译领域获得了一定的成果，并且进一步研究后拓展成了更复杂的统计模型。这些研究使用了加拿大和欧盟现有的语料库，这是由于当地法律规定政府的会议必须翻译成所有的官方语言。但其它大部分系统必须自己构建语料库，并且语料库一直都是限制系统效果的一个主要因素，所以大量的研究从如何在有限的的数据下更高效地学习方面下手。而最新的研究开始着眼于非监督学习算法和半监督学习算法，这些算法可以从无人工标注的数据中学习。总体而言，这种算法相对于监督学习更加困难，并且在数据量相同的情况下，通常得到的结果准确率稍差，但这些包含了互联网语料数据的无人工标注的数据量非常庞大，一定程度上解决了结果不够准确的问题。

### 1.3 本文的研究内容

本文主要研究了一种多模型相融合的词相似计算方法，设计一个用于网络舆情分析的深度学习自然语言处理系统，实现对网络语料的采集、处理、分析以及对结果的反馈与展示等功能。

研究的主要内容如下：

- (1) 语料的获取与处理以及语料库的建立
- (2) 词相似性的计算方法
- (3) 网络舆情分析系统的设计

本文将多模型相融合的词相似计算方法应用于网络舆情的分析中，并且为其开发相关的网络爬虫、数据库以及用户交互界面，实现一个可交互式的网络舆情分析系统。本系统将基于深度学习的自然语言处理应用于词相似性的计算中，使系统在处理语料时不再是单纯的对比词库进行查找，而是对语料进行理解，不再受限于词库的容量，实现了网络舆情分析的准确性和效率的提高。

## 1.4 本文章节安排

本文的安排如下：

第 1 章是引言，主要介绍本课题的研究背景及意义，对舆情分析和自然语言处理的发展现状进行了梳理，介绍了本文的研究内容及结构框架。

第 2 章介绍了网络舆情分析系统的设计。并将中文分词、词相似性的计算方法、网络爬虫和数据库等工具与网络舆情分析系统相结合来完成系统的设计，最后介绍词相似性的评测数据集来用于评价系统的性能。

第 3 章详细介绍了多模型相融合的词相似性计算方法。重点介绍本文所搭建系统采用的算法，该方法将词典模型和统计模型结合使用，并采用动态权重来互补各方法间的不足，并使用该算法在评测数据集上进行分析，判断该算法的计算效果。

第 4 章介绍了网络舆情分析系统的实现，该系统包括网络爬虫、语料数据库、用户交互界面等部分的设计要求，并且详细描述了各部分的实现过程，最后完成对系统的测试并对结果进行分析。

第 5 章是本文的总结及展望。

## 第2章 网络舆情分析系统的设计

### 2.1 网络舆情分析系统设计的理论分析

舆情分析,根本目标便是了解民众的真实想法。但思想本身是存在于神经元之间,随着时间不断变化的,高度动态的生物电化学过程,以现有的科技,难于直接读取。马克思的哲学理论指出:‘语言是思维的外壳’。语言可以将抽象的思维具体化(即使不用于交流,也可帮助人类实现思考的过程,比如人类进行某种谋划的时候常常在脑海中也是使用语言来进行策划),语言可以在个体与个体,群体与群体之间实现思维的交互;语言也可以作为一种存储方式,将大脑中稍纵即逝的电化学过程以文字记录的方式保存下来。而互联网的出现,更是将语言的交互,提升到一个前所未有的高度。互联网所形成的虚拟空间,其实并不虚拟,它是一个超级思维集合体,各种各样的思想,以语言为载体充满了网络,形成了各种舆论的流动。在互联网中,每个亚文化圈子都有自己独特的语言,而语言的这种独特性则反映了亚文化之间的差异。从表面上看,纷繁复杂,尽管都是中文,然而不同的圈子在用词的方法,情感的表达,特殊构词等方面,有着极大的差异,难以从海量的文字中提取真正有用的信息,传统的方法只能看到思想错综复杂的流动而难以提取真实的民意。

哲学与科学的实践指出:在问题纷繁复杂的表象下面,常有一个共通的本质。只要掌握这个本质,就能解决万千的表面问题。例如,机械的运动和电路中的电流变化是不同的物理现象,但都可以以简谐运动为基础的理论对其进行分析。自然语言处理和通信技术是不同的课题但目前都使用马尔可夫过程作为分析和解决问题的工具。这便是哲学中广泛使用的‘透过现象看本质’。互联网舆情,从现象上看,千变万化,千奇百怪,但根据马斯洛的理论,可归结为马斯洛模型中不同层次需求的表达,即:千变万化的互联网语言背后有着共同的本质,进行有效的舆情分析这一目标是可实现的。

传统的舆情分析方法,基于传统的编程理论。传统的编程,本质上是将人类的知识,通过计算机语言写成程序,交由计算机进行执行。对于自然科学和工程技术,这种方法是适用的,例如飞行器的飞控,编程人员只需要将所掌握的所有关于空气动力学的知识写入其中,该程序便能长期有效的运行。但是,这种编程思想一旦面临包含人为因素的问题时,却常常难以有效解决。例如,传统的编程方法可以写出

高效的飞控是因为其对抗的是空气动力学，而车辆的自动驾驶难于用传统的方法解决是因为其对抗的是人（路面上的行人和其他驾驶人员）。

传统的编程方法不适用于解决人的问题的根源在于人的特性。人，是生物的一种。达尔文的进化论指出，‘适应’是生物的一项关键特征。生物在演化过程中会根据环境的变化而改变自身的行为甚至基因。而传统的程序，却是基于命令与服从，没有有效的适应和学习机制，使其无法针对人的变化而做出相应的调整。当人的行为发生变化以后，传统的编程技术便需要程序员将新的变化考虑到应对方案中，通过计算机语言以程序的方式灌注到计算机中。实践证明，这种方式在面对舆情分析这样高度动态化的，充满了人为因素的问题时响应速度慢，成本高昂，有效性差。这便是系统设计中一种常见的失败原因：解决问题的方法和问题的模型不匹配。

传统的基于命令与服从的编程方法难于解决人的问题，那么一种解决问题的思路便是让系统变得像人。如上文所述，人作为生物，适应是一个重要特征，那么也需要在分析系统中加入适应的能力。自适应的概念，在工程中得到广泛的使用，其使系统可以在环境发生变化时让系统仍然能够高效运作。工程界对自适应给出了准确定义：自适应就是在处理和分析过程中，根据处理数据的数据特征自动调整处理方法、处理顺序、处理参数、边界条件或约束条件，使其与所处理数据的统计分布特征、结构特征相适应，以取得最佳的处理效果的过程。一般而言，自适应系统包含两个主要子系统：对环境进行感知提取特征的子系统，对自身进行调整以适应环境的子系统。

## 2.2 网络舆情分析系统的结构设计

前节的分析指导了本课题舆情分析系统的设计理念：以适应为核心。同时也指出了设计方向：模仿人类访问互联网的方式对系统进行设计。人类在阅读并理解互联网的过程中，首先通过大量阅读，在大脑中存储大量的互联网语言，这便是自适应系统中的第一个子系统；然后，发现语言规律，结合上下文，理解特定词汇，对自己大脑中的语言模型进行调整，从而能轻松的理解互联网语言，这便是自适应的第二个子系统。

综上所述，对于本课题的舆情分析系统的顶层设计，系统主要由两个子系统构成：语料收集与预处理子系统由网络爬虫和数据库构成，以模拟人类浏览和阅读互联网的行为，自适应自然语言处理量化子系统是可自动调整的分析和量化模块，以模拟人类动态调整对语言理解的大脑模型。

本课题采用自顶向下的设计方法。在上文中，已经根据问题的本质，将系统划

分为两个大的子模块，为了完成最终的设计，需要不断细化，不断具体化，才能得到一个可实现的系统。

### 2.2.1 语料收集与预处理子系统设计

语料收集与预处理子系统需要模仿人类浏览和阅读。人类理解文本的前提是获得足够文本信息，对于尚未理解的文本，必须通过大量的浏览和阅读，获取足够的信息才能对文本进行分析理解，所以大量语料数据的获取是必不可少的一步。随着技术的发展，虽然网络文本数量及其庞大，但也出现了相应的工具来进行数据的浏览和收集。网络爬虫技术就是模拟人类浏览阅读网络信息，大量的文本通过这一技术被快速的浏览并记录下来，为之后的语言理解打下基础。而数量庞大的文本是无法在短时间内处理完成的，所以这里通过类似于书本的数据库技术，将这些文本有条理的存储起来，保证了后续的理解过程能够高效且准确。

首先为了实现语料收集与预处理子系统中模仿人类浏览与阅读的目的，需要运用到网络爬虫技术。网络爬虫是一种可以自动浏览互联网的网络机器人<sup>[3]</sup>。网络爬虫存在于统一资源定位系统（Uniform Resource Locator, URL）中，网络爬虫在访问这些 URL 时，会分析出页面之中所有的超链接，然后将这些超链接写入一张待访列表，也就是所谓的爬行疆域。这个疆域上的 URL 将会按照一套策略循环来访问<sup>[4]</sup>。网络爬虫在执行的过程中复制并保存其访问过网站上的信息，并且归档以使信息比较容易被查看。归档文件通常以一种可以在互联网上查看，阅读和导航的方式进行存储，并且保留为“快照”。网络爬虫一般分为数据采集、处理和存储三个部分<sup>[5]</sup>。网络爬虫的工作流程一般分为 4 步，第一步是通过 HTTP 库向目标站点发送一个 Request 并等待服务器响应。第二步是获取响应内容，如果服务器响应正常，会得到一个 Response，其内容就是所要获取页面的内容，一般包括 HTML、JSON 字符串和二进制数据（图片、视频）等类型。第三步是解析内容，如果是 HTML，可以用正则或者是网页解析库来解析，如果是 JSON 则直接转换成 JSON 对象，再来解析，如果是二进制数据可以保存或进行其他处理。第四步是保存数据，获取到的数据可以存为可读文本，也可以保存到数据库中。通过网络爬虫可以很好地获取各种网络信息，并且只要完成网络爬虫的编写，使其具有能够完成任务的功能，就可以实现无人监督下的自动数据爬取<sup>[6]</sup>，在自然语言处理的语料库的建立上有着显著的作用，可以提高数据收集的效率以减少收集时间。网络爬虫的工作流程图如下图 2.1 所示：



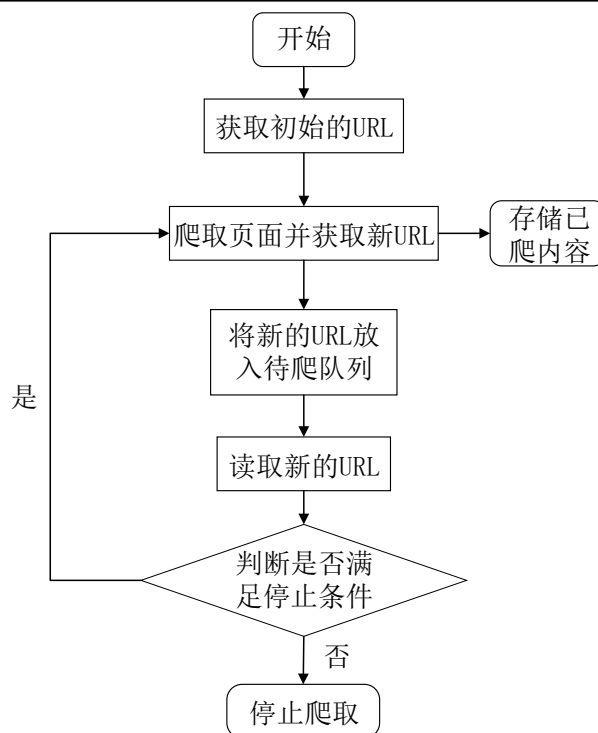


图 2.1 网络爬虫工作流程图

然后为了方便阅读语料数据，本系统需要将其以类似书本的形式有条理的存储起来，所以使用数据库来实现这一任务，数据库是用于存放计算机数据的仓库<sup>[7]</sup>，这个仓库是按照一定的数据结构来对数据进行组织和存储的，使用者可以通过数据库提供的多种方法来管理其中的数据<sup>[8]</sup>。传统的数据库模型一般有层次式数据库、网状数据库和关系型数据库，而现在比较常用的数据库模型主要有关系型数据库和非关系型数据库<sup>[9]</sup>。关系型数据库是为了解决层次式数据库和网状数据库的问题而提出的一种数据库，这两种数据库很好地解决了数据的集中和共享问题，但是在数据独立和抽象级别上仍有欠缺。而非关系型数据库也被称为 NoSQL（Not Only SQL）数据库，NoSQL 的产生是作为传统数据库的补充而不是要彻底否定关系型数据库<sup>[10]</sup>。NoSQL 数据库在特定的场景下可以发挥出更高的效率和更好的性能。随着 web2.0 网站的大量出现，传统的关系型数据库在应对 web2.0 网站，特别是对于规模日益扩大的海量数据、超大规模和高并发的微博、微信、SNS 类型的 web2.0 纯动态网站已经力有不足，暴露了很多的问题。NoSQL 类的数据库就是这样的情景中诞生并得到了非常迅速的发展，它改变了长久以来关系型数据库与 ACID 理论独占市场的局面。NoSQL 数据存储不需要固定的表结构，也不存在连续操作，所以在大数据存取上具备关系型数据库无法比拟的性能优势<sup>[11]</sup>。本系统中数据库作为语料库的存储位置，通过数据库可以将语料库中的语料有序的整理并储存，使得后期在处理语料数据和对语料数据进行深度学习计算时能够高效率的存取。

最后对于数据库中收集到的语料数据在送入自适应自然语言处理量子子系统进行处理前还有一个问题,由于中文语料中词与词之间没有分割,无法直接使用统计模型进行处理,所以需要通过分词算法将中文句子的词间添加空格以进行标识。中文分词效果好不好对信息检索、实验结果具有显著影响,同时分词的背后其实涉及各种各样的算法。中文分词算法按分词的特点可以分为四类,基于理解的分词方法<sup>[12]</sup>、基于语义的分词方法<sup>[13]</sup>、基于统计的分词方法<sup>[14]</sup>和基于规则的分词方法<sup>[2]</sup>。现在使用的分词工具大多是基于统计的分词方法<sup>[15]</sup>,例如 Jieba、SnowNLP、THULAC 等<sup>[16]</sup>,一般在中文分词中性能相对较好的是 Jieba,这种分词工具在现今大数据时代下拥有着极强的可塑性,并且其支持三种分词模式,包括精确模式、全模式和搜索引擎模式<sup>[17]</sup>,一般来说该工具可以较好的完成各类分词任务。

### 2.2.2 自适应自然语言处理量子子系统设计

自适应自然语言处理量子子系统需要模仿人类对语言进行理解。为了完成这一目标,需要对语言的特性进行更加深入的讨论。首先,实用的主流语言需要具备稳定性,语言的巨大变化将会使语言失去传递信息的能力,例如,尽管都是中文,没有学过文言文的中文使用者在阅读古籍时会遭遇重大的障碍;其次,语言在保持一定稳定性的同时,也会不断变化,尤其是现代社会,互联网和全球化加速了语言的变化,新的词汇,甚至新的语法结构不断被引入到语言之中<sup>[18]</sup>。即:现代汉语,尤其是互联网上所使用的汉语,同时具备了稳定性和高度的动态性。

根据这一特点,系统应当构建两个集合 S1, S2, 分别代表汉语中变化性的部分构成的集合与稳定性的部分构成的集合,本文中 S1 集合采用了统计模型,统计模型中的词都是由网络中收集而来的词汇,这些词汇具有变化性,其含义并不是固定的,在不同的时间,不同的语境下可能具有不同的意义。S1 集合除了其词汇特性外,在用于计算时 S1 集合的优点在于其覆盖范围大,只要用于计算的语料库足够大,就可以覆盖几乎所有的词,并且结果的准确率也比较高<sup>[19]</sup>,但也存在一定的缺点,基于统计的方法只是计算每个词在句子特定位置出现的可能性,而不是根据词的含义来分析,所以不能保证此相似度绝对准确,因为两个词相似度很高在基于统计的算法中的意义是两词在同一位置出现的概率大而并非意义接近。

S1 集合中常用的统计模型有词袋模型<sup>[20]</sup> (Bag of Word, BOW) 以及属于 Word2Vec 工具<sup>[21]</sup>的连续词袋模型<sup>[22]</sup> (Continuous Bag of Word, CBOW) 和跳字模型<sup>[22]</sup> (Skip-Gram)。Word2Vec 工具是现在比较热门的研究方向,其中所使用的词向量来进行词相似性的研究方法取得较好的效果,词向量是一种由神经网络学习得到的词的分布式表示,通过 Word2Vec<sup>[23]</sup>训练出的各词的词向量提高了词相似性

的计算结果。各算法的具体细节如下：

(1) 词袋模型就是将所有的词装进一个袋子里，不考虑其的词法和语序问题<sup>[20]</sup>。由于失去了语法和词序，词袋模型不考虑文本中词与词的上下文关系，仅仅只考虑所有词的权重，而权重只与词在文本中出现的频率有关<sup>[24]</sup>。词袋模型首先会进行分词，然后通过统计每个词在文本中出现的次数，就可以得到该文本基于词的特征，如果将各个文本样本的这些词与对应的词频放在一起，也就是向量化。向量化完毕后一般也会使用 TF-IDF 进行特征的权重修正，再将特征进行标准化。再进行一些其他的操作后，就可以将数据带入机器学习模型中计算。词袋模型具有一定局限性，由于其失去了语法和词序，所以丢失了一部分文本的语义，使计算结果不够理想。

(2) 连续词袋模型是 Word2Vec 工具中的一种模型，其主要是通过上下文的多个背景词来预测一个中心词<sup>[25]</sup>。连续词袋模型是在词袋模型上进行改进的一种算法，是考虑词语位置关系的一种模型，通过大量语料的训练，将每一个词语映射到高维度（几千、几万维以上）的向量当中<sup>[26]</sup>，通过求余弦的方式，可以判断两个词语之间的关系，由于将词袋模型不考虑的词法和语序加入到了算法中，所以连续词袋模型的训练输入的是某一个特征词的上下文相关的词对应的词向量<sup>[27]</sup>，而输出就是这特定的一个词的词向量。这种模型由于将词序也作为计算的一部分使得计算结果的效果更好。连续词袋模型示意图 2.2 所示：

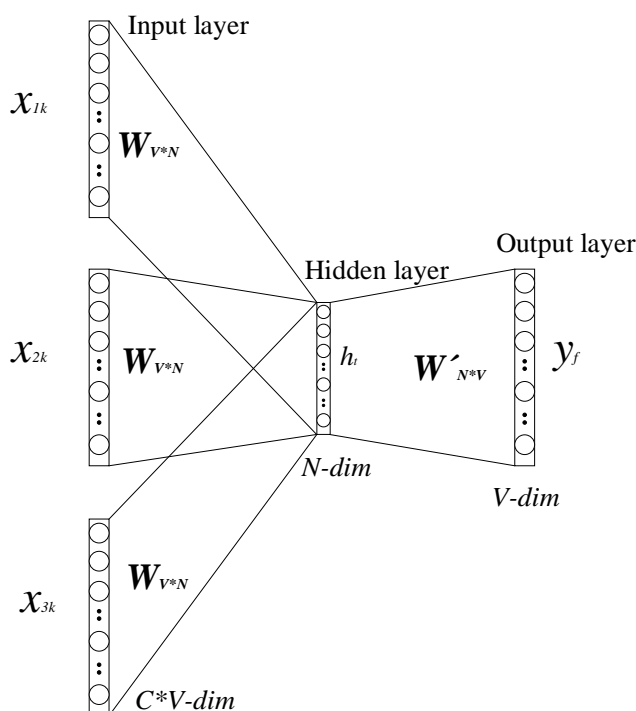


图 2.2 连续词袋模型示意图

(3) 跳字模型也是 Word2Vec 工具中的一种模型，它的整体思路和连续词袋模

型大致一样，但是主要是通过一个中心词来预测上下文的多个背景词<sup>[27]</sup>，与连续词袋模型类似，由于将词序也作为计算的一部分，所以计算结果更加准确。跳字模型示意图如图 2.3 所示：

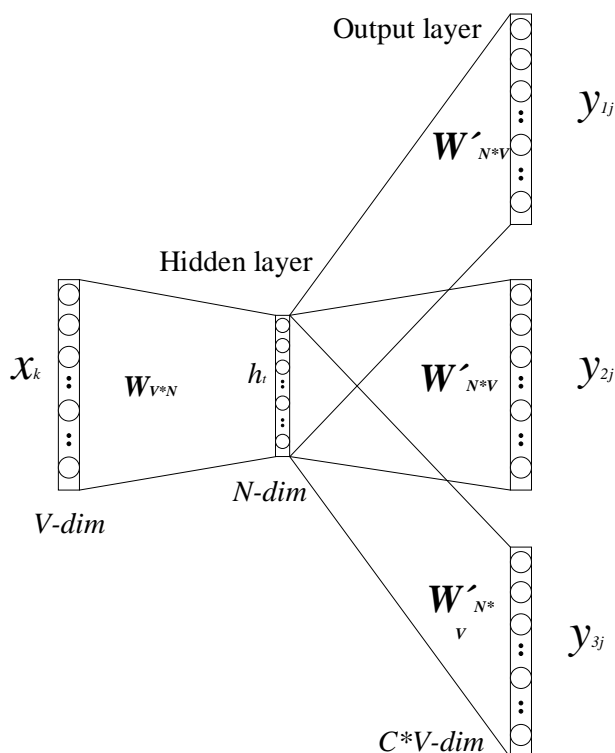


图 2.3 跳字模型示意图

S2 集合采用了词典模型，词典模型中词汇的特点在于其稳定性，词典模型中的词汇是常用的词汇，这些词汇已拥有公众统一认可的理解，有其稳定的含义。同时 S2 集合是根据编写成的词典来判断两词的相似性<sup>[28]</sup>，这种基于词典模型的方法的优点在于结果非常准确，由于同义词词典是由语言学学者们编纂而成的，人工分类所反应的词相似性会更能体现词之间意义的关联性，并且也会更贴近人的使用习惯<sup>[29]</sup>。但是这种方法有三个明显的不足，一是词汇的数量不足，覆盖率不够、二是更新率不高，往往需要一段时间分类才能更新词库、三是词相似度的颗粒度比较粗糙，只能按照相同或不相同两种定义来二分类词的相似度<sup>[30]</sup>。

S2 集合在进行计算时使用词典模型，词典模型是使用由研究者编著的词汇词典作为分析基础的一种模型，中文较常用的有《同义词词林》<sup>[31]</sup>、HowNet<sup>[32]</sup>、现代汉语词典等等。基于词典模型的方法一般是依据语言学研究人员总结定义的词汇的语义来计算词汇的相似性。

(1) HowNet 的每一个词的语义被称为“概念”，“概念”是对词汇语义的一种描述<sup>[33]</sup>，每一个词可以被描述为几个“概念”。“概念”是用一种“知识表示语言”

来描述的,这种“知识表示语言”所用的“词汇”叫做“义原”。“义原”是用于描述“概念”的最小意义单位<sup>[34]</sup>。HowNet 与一般的词汇词典不同,例如同义词词林是一种树状结构体系,HowNet 是使用一个或多个“义原”来对一个“概念”进行描述,HowNet 一共包含 800 多个义原,刘群等<sup>[35]</sup>提出把义原分为四个义原集合,通过计算不同义原之间的路径长度来计算其的相似度,并以此计算词语的相似性。

(2)《同义词词林》<sup>[31]</sup>收录的所有词条按照树状的层次结构组织到一起,将词汇分为了大类 12 个,中类 97 个,小类 1400 个。在每个小类中有若干个词,这些词按照词义的远近和相关性分为若干个词群,然后再将词群中的词细分为若干行,每一行的词语要么是词义相同或十分接近,要么就是语义有很强的相关性<sup>[36]</sup>。小类中的词群就是第四级分类,而每行就是第五级分类,一般称为原子词群、原子类或原子节点,同义词词林的树状结构如图 2.4 所示:

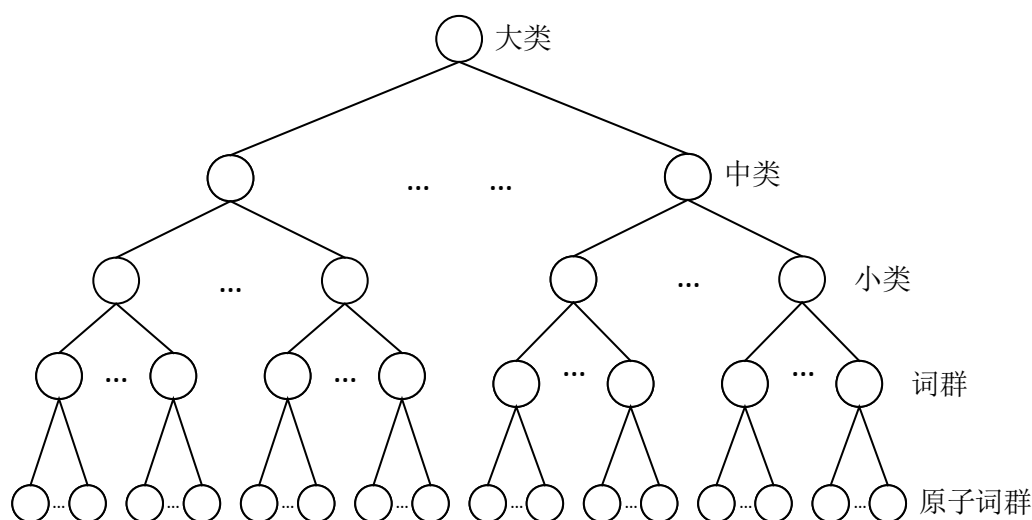


图 2.4 同义词词林的树状结构

《同义词词林》的编写者按照五层编码的模式对词典进行了编码<sup>[31]</sup>,其中大类用大写字母表示,中类用小写字母表示,小类用两位十进制整数表示,词群用大写字母表示,原子词群用两位十进制整数表示,但第五级由于有些是同义词,有些是相关词,有些只有一个词,所以在后面添加一个标记包括“=”、“#”、“@”三种,“=”表示相等或同义,“#”表示不等或同类,“@”表示自我封闭或独立<sup>[37]</sup>。

本文采用在构建两个集合的同时,进一步的针对文本内容的不同对 S1 和 S2 采用不同的权重来分析文本,例如,对微博,论坛用户的发言这类变化性大的文本进行处理时,给予 S1 更高的权重,而针对新闻稿,公告等词语稳定性较强的文本进行处理时,给予 S2 更高的权重,这种因地制宜,因时制宜的动态权重,使得舆论分析系统可以有效理解网络言论,得出实时的网络舆情。

最后由于词相似性是指词相似的程度,而人类表达词相似的程度时没有一个

量化标准的概念,无法直接使用计算机进行处理,所以想要将 S1, S2 两个集合所计算出的结果用于舆情分析的话,就要对其结果进行量化处理以得到直观的数字化结果。

对于统计模型的词相似性计算,由于统计模型采用了词向量将每一个词赋予一个多维向量来表示其在词库中的位置,所以通过计算两个词向量间的夹角就可以量化两词的词相似性,计算公式<sup>[37]</sup>如式(1):

$$sim_1(W_1, W_2) = \cos \theta = \frac{\vec{v}_1 * \vec{v}_2}{|\vec{v}_1| * |\vec{v}_2|} \quad (1)$$

其中  $\vec{v}_1$  和  $\vec{v}_2$  是  $W_1$  和  $W_2$  的词向量,公式中  $\cos \theta$  取值范围为  $[0,1]$ ,当两个词为同义词或完全相同的词时,两词向量的夹角为  $0^\circ$ ,词相似性为 1,而当两个词为完全不同的词时,两词向量的夹角为  $90^\circ$ ,词相似性接近于 0<sup>[39]</sup>。

对于 HowNet<sup>[32]</sup>,刘群等<sup>[35]</sup>人提出一种量化计算的方式。所有的义原按照上下位结构组成了一个层次体系,在这一体系中,词语间的远近距离反映了词语间语义的相似程度,其数学关系如公式(2)所示:

$$\begin{cases} \lim_{d \rightarrow \infty} sim(W_1, W_2) = 0 \\ \lim_{d \rightarrow 0} sim(W_1, W_2) = 1 \end{cases} \quad (2)$$

也就是说,词间的距离越短则词义越相似,词间的距离越长则词义越不相似<sup>[34]</sup>。通过计算语义距离就可以量化词相似性。于是就定义两个义原之间的语义距离的计算方法如公式(3):

$$Sim_2(W_1, W_2) = \frac{\alpha}{d(W_1, W_2) + \alpha} \quad (3)$$

其中  $W_1$  和  $W_2$  表示两个义原,  $d(W_1, W_2)$  表示  $W_1$  和  $W_2$  在义原层次体系中的最短路径的长度<sup>[40]</sup>。 $\alpha$  表示一个可调节的参数。通过这一公式可以得出两义原的相似度的取值范围为  $[0,1]$ ,当两个词为同义词或完全相同的词时,  $d(W_1, W_2)$  为 0,词相似性为 1,而当两个词为完全不同的词时,  $d(W_1, W_2)$  为一个很大的值,词相似性接近于 0。

对于《同义词词林》,由于《同义词词林》采用五层的树状结构,所以给每一层赋予一个权值,然后根据两个词所在的最低层次来计算两词的相似性,一般两词的相似性就是该层的权值,所有的权值在  $[0,1]$  之间取值,定义两词  $W_1$  和  $W_2$  的相似性的方法<sup>[41]</sup>如式(4):

$$sim_3(W_1, W_2) = \begin{cases} a & \text{在同第五级分支下且标志位为= 时} \\ b & \text{在同一第四级分支下} \\ c & \text{在同第五级分支下且标志位为# 时} \\ d & \text{在同第三级分支下} \\ e & \text{在同第二级分支下} \\ f & \text{在同第一级分支下} \\ g & \text{标志位为@ 时} \end{cases} \quad (4)$$

## 2.3 词相似性评测数据集

最后,对于判断网络舆情分析系统计算效果的优良,需要制定一个评定标准,而在判断词相似性的计算是否准确时一般会将计算机对两词相似性的打分与人工打分结果进行比对,其中人工打分是由研究人员对词对进行相似性判断得出的,也就是词相似性的评测数据集,对英文词相似性的评测数据集一般有 WS-353 数据集, SimLex-999 数据集等等,对于中文词相似性的评测数据集一般有 SemEval-2012、PKU-500<sup>[42]</sup>等数据集,由于本文是对中文词相似性的分析,所以重点对中文评测数据集进行介绍。

SemEval-2012 数据集是为 SemEval-2012 汉语词语语义相似度任务所编写的评测数据集,该数据集提供了 348 个词以及词对间的相似性评分,在该数据集的构建中,首先由两名研究生翻译 WS-353 数据集中的所有词对,在翻译完成后,有 169 个词对两人的翻译结果相同,而剩下的 184 个词对不同,对于不同的词对由第三位研究生基于两个规则进行再次校对,一是翻译结果类似时取长度更长的词,二是翻译结果类似时取常用的词。最后将翻译结果完全不同的 5 个词对从数据集中去除。接着挑选出 20 位母语为汉语且研究方向为汉语言学的研究生,按照两词完全不同打 0 分,两词完全相同打 5 分的规则给 348 个词打分,最后取 20 位研究生打分的平均分作为评测数据集的结果。

PKU-500 数据集是为 NLPCC-ICCPOL 2016 汉语词语语义相似度任务<sup>[43]</sup>所构建的评测数据集,该数据集提供了 500 个词对以及词对间的相似性评分,在该数据集的构建中,研究人员从人民日报语料中挑选了 514 个词语,从微博语料中挑选 202 个词语。对挑选出的 716 个词语中的每一个目标词,从《同义词词林扩展版》中随机抽 3 个候选词语:第一个候选词来自目标词所在的同义词集合;第二个候选词来自父结点的词语;第三个候选词随机挑选,然后研究者从语言学角度去除一些候选词并增加一些新词,最终得到了 470 个词对<sup>[44]</sup>。最后再从 WS-353 数据集

中选取 30 组英文词对并翻译成汉语词对,加入之前的 470 个词对构成了一共 500 个词对的评测数据集。接着挑选出 20 位母语为汉语且研究方向为汉语言学的研究生,按照两词完全不同打 1 分,两词完全相同打 10 分的规则对 500 个词对进行相似度打分,并取 20 位研究生打分的平均分作为评测数据集的结果。PKU-500 数据集是为了测试汉语词语语义相似度计算任务构建的评测数据集,该数据集涵盖实义词如名词、动词、形容词等,以及功能词如副词、连词等,每个词对都由 20 位研究生人工打分,是一个完备的语义相似度计算评测数据集,更适合用来进行评测计算结果。

## 2.4 本章小结

本章首先介绍了网络舆情分析系统的整体设计,从网络舆情的本质入手,分析了网络舆情分析中传统方法所面对的问题,并对这些问题提出了解决方案,将网络舆情系统按自顶向下的设计思路分为了两个子系统。在语料收集与预处理子系统中结合网络爬虫、数据库和中文分词来完成模拟人类浏览和阅读的目标,自适应自然语言处理量子系统则使用了结合深度学习的自然语言处理的各模型来完成分析语料的任务。最后为了完成对网络舆情分析系统的评估,对当前常用的中文词相似性计算的主流评测数据集进行了详细的介绍。



## 第3章 动态权重多模型相融合的词相似性算法

### 3.1 动态权重多模型相融合的词相似性算法的设计

第二章中对系统整体进行了设计，而本节将对所设计的这个系统所包含的关键技术展开讨论，详细介绍动态权重多模型相融合的词相似性计算方法中各模型的选择、动态权重的计算方法以及算法的整体结构。

#### 3.1.1 统计模型的选择

现有的词相似性分析中，统计模型实际反映的是在一定上下文中某个词出现的可能性，因此一个词的意义是通过对其上下文的建模而计算得到的。常用的统计模型包括 BOW 模型和属于 Word2Vec 的 CBOW 模型和 Skip-gram 模型<sup>[45]</sup>。Skip-gram 模型主要是根据某一词预测上下文，与本系统的主要需求不符，所以不做考虑。而 BOW 模型忽略掉了文本的语序和语法等要素<sup>[46]</sup>，仅仅是将其看作若干单词的集合，文本中每个单词的出现都是独立的，因此该模型的计算结果准确度有限，因此也不采用这一模型。CBOW 模型针对 BOW 模型的缺点进行了改进，将文本的语序作为一个要素加入了计算中，大幅提高了计算的准确性和训练速度<sup>[22]</sup>。CBOW 将词向量化<sup>[47]</sup>并加入神经网络使得统计模型可以运用深度学习进行计算，进一步提高了计算效率<sup>[20]</sup>。因此本文采用了属于 Word2Vec 的 CBOW 模型来对语料库进行计算以提升词相似度的准确性。

#### 3.1.2 词典模型的选择

基于中文的常用近义词词典有 HowNet 和《同义词词林》，词典作者根据其词汇分类体系的结构特点制作了这两个近义词词典。本文将采用这两个词典模型。

在 HowNet 中，词语由一个或多个义项组成，而每个义项又由更小的语义单位义原和几十种动态角色组合而成<sup>[48]</sup>，在 HowNet 中的义原有 1500 个。HowNet 中每一个词语一般有一个或多个概念。例如“北京”一词在 HowNet 中的描述如图 3.1 所示。通过图 3.1 可以看出义原是以多层结构体系分布。通过构成词义的义原之间的相对距离计算可以得到词语间的相似度<sup>[49]</sup>，本文根据这一方法，使用 HowNet 词典来分析词之间的相似度，并将其用于多模型融合的方法中。



有效地提高了计算词相似性的准确性。本文设计的动态权重多模型融合有以下两个研究要点：

第一是根据语料的特点将语料分为规范性语料（例如新闻，论文等）和非规范性语料（例如微博，百度贴吧等）分别进行词向量模型的计算，得到同一个词分别在规范性语料和非规范性语料中的两个不同的向量，在判断词的相似性之前，首先根据语料的特点选取相应的词向量，这种方法有效地提高了词相似性的准确率。

第二是将统计模型和词典模型融合，根据对计算结果的分析发现，统计模型对 PKU-500 数据集中极端词的相似性判断并不好，其中主要的原因是词向量模型并不是真正的反映词的意义，而是词在上下文出现的概率，导致词向量很难出现如同 PKU-500 中 1 分或 10 分的极端情况。对于这一部分词就要采用词典模型中的结果，词典模型中只能以相同或不相同两种定义来反映词的相似性，正好对应词向量模型中的极端情况。通过模型融合计算可以增加词相似性的准确率。

本文提出一种多模型融合的词相似性计算方法，将 Word2Vec、HowNet、《同义词词林扩展版》三个模型进行融合，通过动态调整权重的方式，对词的相似度进行了计算，有效地将各模型整合在一起，相互弥补各模型间的缺陷，提高了词相似性的准确率。

### 3.1.4 算法的整体结构

本文将三个不同语料库计算出的统计模型和两个词典模型通过对词类型的分析采用动态权重，整体系统的结构如图 3.3 所示。

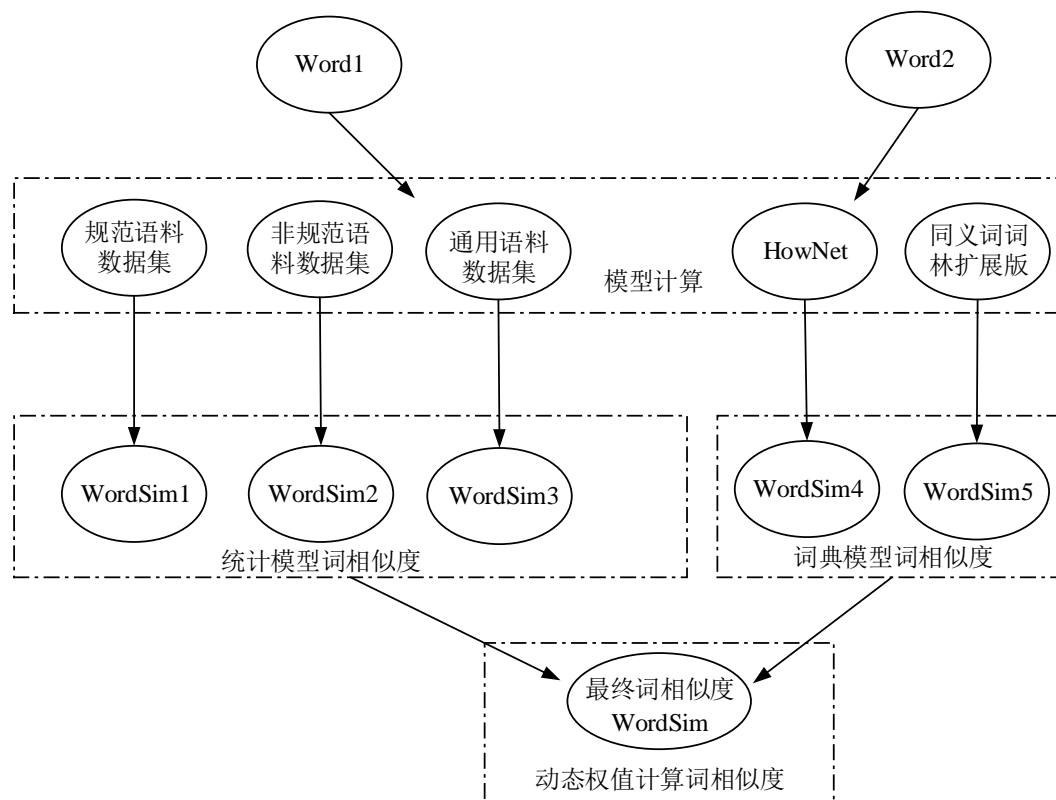


图 3.3 系统整体结构图

整体框架结构主要包括 3 层，第一层是将两个需要计算相似度的词输入到 5 个模型中。第二层是 5 个模型进行计算，分别得出两个词在该模型下的相似度。第三层是分别根据三个统计模型相似度的标准差和两个词典模型相似度的标准差动态确定每个模型计算的相似度在最终结果中的权重。最终可以得到两个词的相似度。

在模型的第二层中，统计模型中两个词语相似度是通过两词所对应的词向量之间的余弦夹角所定义的，即余弦相似性，计算公式如 2.2.2 中式(1)，词典模型中 HowNet 模型按照李群等人的方法计算得到，计算公式如 2.2.2 中式(3)，《同义词词林扩展版》的相似度是根据每个词所处的层次的不同赋予了不同的权值，计算公式如 2.2.2 中式(4)。

在模型的第三层中，将第二层所计算出的结果分别按照统计模型和词典模型分别计算。由于整体系统采用动态权重，所以这里根据不同统计模型或词典模型间的标准差来分配权值，因为标准差可以反映一个数据集的离散程度。若不同模型间的离散程度大则说明部分模型的结果误差较大，这个时候则需要给单一模型情况下准确率高的模型分配更大的权值来减小误差。而不同模型间离散程度小则说明计算结果大体一致，给各模型均匀分配权值即可。通过标准差的引入，可以进一步提高整体计算效果。其中权值的计算方法如下：

为了导出动态加权方法的数学描述，对于一个集合  $S$ ，定义其指数函数  $1_s(\cdot)$ ，如式(5)：

$$1_s(x) = \begin{cases} 1 & x \in S \\ 0 & \text{其他} \end{cases} \quad (5)$$

其次，令  $\sigma_s^2$  为各统计模型计算结果的方差，将方差的分布范围划分为若干个区域，如式(6)：

$$\sigma_s^2 \in [b_0, b_1) \cup [b_1, b_2) \cup \dots \cup [b_{i-1}, b_i) \quad (6)$$

为每个区间分配一个对应的权值矢量  $\mathbf{h}_{si}$ ，如式(7)：

$$[b_{i-1}, b_i) \rightarrow \mathbf{h}_{si} \quad (7)$$

由于本文提出的方法使用了三个统计模型，因此权值矢量包含三个元素，即  $\mathbf{h}_{si} \in \mathbb{R}^3$ ，最后，在获得了三个统计模型的方差后，用式(8)计算权值矢量：

$$h_s(\sigma_s^2) = \sum_i 1_{[b_{i-1}, b_i)}(\sigma_s^2) \mathbf{h}_{si} \quad (8)$$

同理，可得两个词典模型所使用的权值计算公式，如式(9)：

$$h_D(\sigma_D^2) = \sum_j 1_{[b_{j-1}, b_j)}(\sigma_D^2) \mathbf{h}_{Dj} \quad (9)$$

其中  $\sigma_D^2$  为两个词典模型计算结果的方差， $[b_{j-1}, b_j)$  代表方差分布的第  $j$  个区间， $\mathbf{h}_{Dj} \in \mathbb{R}^3$  代表第  $j$  个区间对应的权值矢量。

## 3.2 计算过程与结果分析

### 3.2.1 计算结果的评价标准

本章采用斯皮尔曼等级相关系数(Spearman rank correlation coefficient)来评价本章采用的多模型融合计算词相似性的效果。斯皮尔曼等级相关系数  $\rho$  是评价两个变量之间相关程度的一个值，一般来说两组数所计算出的斯皮尔曼等级相关系数越大，则说明两组数的相关性越高。在本文中对算法计算出的两词相似度与 PKU-500 数据集中人工给定的相似度进行斯皮尔曼等级相关系数的计算，斯皮尔曼等级相关系数越高说明算法计算的结果与人工标定的结果有更高的相关性，也就是说明算法计算的结果更符合人的实际使用场景。斯皮尔曼等级相关系数的计算公式如式(10)。

$$\rho=1-\frac{6\sum_{i=1}^n(R_{xi}-R_{yi})^2}{n(n^2-1)} \quad (10)$$

### 3.2.2 计算对比试验的方案

本文对词的相似度计算对比试验的方案是将统计模型、词典模型、简单权重的多模型融合、动态权重的多模型融合分为四组，每组计算得到相应的斯皮尔曼等级相关系数，通过斯皮尔曼相关系数评价各方案性能。对比试验各组的设置如下：

(1) 统计模型词相似度性能评价实验分别对容量为 4GB 的贴吧/微博数据集、4GB 百科数据集和腾讯开源 NLP 数据集计算词相似度，并与 PKU-500 数据集计算斯皮尔曼等级相关系数；

(2) 词典模型词相似度性能评价实验分别对 HowNet 和《同义词词林扩展版》计算词相似度，并与 PKU-500 数据集计算斯皮尔曼等级相关系数；

(3) 简单权重的多模型融合的词相似度性能评价实验中将容量为 4GB 的贴吧/微博数据集、4GB 百科数据集、腾讯开源 NLP 数据集、HowNet、《同义词词林扩展版》计算词相似度的权重设置为固定权重。计算结果与 PKU-500 数据集计算斯皮尔曼等级相关系数；

(4) 动态权重的多模型融合的词相似度性能评价实验中将统计模型和词典模型中各数据集的权重分开计算，统计模型的权重会根据三个统计模型计算的词相似度的标准差来判断应该用什么样的权重，词典模型也是采用相同的策略，然后再根据词典模型和统计模型分别求出来的词相似度再次分配权值并得到最终的词相似度，并与 PKU-500 数据集计算斯皮尔曼等级相关系数。

### 3.2.3 统计模型的词相似性计算

在统计模型的词相似度计算中，基于 Word2Vec 对各数据集进行计算。其中腾讯开源 NLP 数据集为已计算完成的词向量所以可直接使用，而 4GB 贴吧/微博数据集和 4GB 百科数据集需要计算后得到结果。“4GB 非规范”表示容量为 4GB 的贴吧/微博数据集，“4GB 规范”表示 4GB 百科数据集，“Tencent-NLP”表示腾讯的开源 NLP 数据集，这三个数据集对 PKU-500 数据集词汇的覆盖率如下表 3.1 所示：

表 3.1 不同数据集对 PKU-500 数据集的词汇覆盖率

数据集	未包含词语/个	词语覆盖率/%
-----	---------	---------

4GB 非规范	84	91.6
4GB 规范	16	98.4
Tencent-NLP	5	99.5

表 3.1 可以发现同样大小的数据集下, 规范语料库会比非规范语料库的覆盖率更高, 而 Tencent-NLP 数据集由于是通用数据所以覆盖率最高, 但该数据集中只含有中文词汇, 对于 PKU-500 数据集中的英文缩写类似于 WTO, GDP, WHO 等则无法覆盖。

根据各数据集训练所得的统计模型分别计算了 PKU-500 中词对的词相似度, 并进一步计算了与 PKU-500 数据集的斯皮尔曼等级相关系数, 结果如表 3.2 所示:

表 3.2 不同统计模型对 PKU-500 数据集词汇相似度计算效果

统计模型	斯皮尔曼等级相关系数
4GB 非规范	0.384
4GB 规范	0.396
Tencent-NLP	0.497

通过表 3.2 的数据可以发现数据集越大则最终的计算结果会越准确, 而规范语料库比非规范语料库拥有更好的效果是由于 PKU-500 的人工打分是由专家学者完成的, 其词的意义更接近于规范语境的使用情况, 所以在这一数据集下规范语料库的效果会比非规范语料库的效果更好。

### 3.2.4 词典模型的词相似性计算

在词典模型的词相似度计算中, 《同义词词林扩展版》需要根据词在词典中的位置与词典的结构来定义其相似度的具体参数, 该词典模型参数的设置如下表 3.3 所示:

表 3.3 《同义词词林扩展版》相似度参数

参数	值	参数所使用的条件
a	0.95	在同第五级分支下且标志位为“=”时
b	0.7	在同一第四级分支下
c	0.5	在同第五级分支下且标志位为“#”时
d	0.4	在同第三级分支下
e	0.2	在同第二级分支下
f	0.1	在同第一级分支下
g	0	标志位为“@”时

而 HowNet 词典的词相似性根据李群等<sup>[35]</sup>的方法计算。两个词典模型对 PKU-500 数据集词汇的覆盖率如下表 3.4 所示：

表 3.4 不同词典对 PKU-500 数据集的词汇覆盖率

数据集	未包含词语/个	词语覆盖率/%
HowNet	170	83.0
同义词词林扩展版	86	91.4

对两词典模型和两词典模型的加权融合分别计算了 PKU-500 中词汇组的词相似度，并计算了与 PKU-500 数据集的斯皮尔曼等级相关系数，结果如表 3.5 所示：

表 3.5 不同词典模型对 PKU-500 数据集词汇相似度计算效果

数据集	斯皮尔曼等级相关系数
HowNet	0.373
同义词词林扩展版	0.460
HowNet+同义词词林	0.476

通过表 3.5 的数据可以发现。同义词词林在合适的权重下计算词相似度的效果会好于 HowNet，主要有两个原因，一是同义词词林的词覆盖率更高，二是同义词词林的分布结构更为合理。而由于两词典所无法覆盖的词并不相同，所两词典模型加权可以使得词覆盖率进一步提高，所以两词典进行加权计算时斯皮尔曼等级相关系数进一步的增大，计算效果也更好。

### 3.2.5 简单权重多模型融合的词相似性计算

在简单权重的多模型融合的词相似度计算中将 4G 贴吧微博语料库、4G 百科语料库、腾讯开源 NLP 数据集、HowNet、《同义词词林扩展版》五个模型分两种不同的权重相对比，一组为 0.2、0.2，0.2、0.2、0.2 另一组为 0.10、0.10、0.3、0.25、0.25。这两组的斯皮尔曼等级相关系数如表 3.6 所示。

表 3.6 简单权重多模型融合对 PKU-500 数据集的词相似度计算效果

权重	斯皮尔曼等级相关系数
权重组合 1	0.503
权重组合 2	0.516

根据表 3.6 的结果可知，当多模型融合时可以提高词相似度计算的效果，比单一模型的效果要好，而且权重不同的效果也不同，权重组合 2 的效果优于权重组合 1 的效果，这是由组合 2 的词覆盖率更大、词相似度效果更好的模型所占的权重更大，所以根据词的实际情况选取合适的权重可以有效提升词相似度的计算效



果。

### 3.2.6 动态权重多模型融合的词相似性计算

在动态权重的多模型融合的词相似度计算中权重将由分两步确定，第一步分别确定统计模型和词典内部模型的权重。

首先对统计模型的计算结果的方差分布划分为两个区间：

$$\sigma_s^2 \in [b_0, b_1) \cup [b_1, b_2)$$

其中  $b_0 = 0, b_1 = 0.12^2, b_2 = \infty$ ，两个区间对应的权值分别为：

$$\begin{cases} \mathbf{h}_{s1} = [0.15 & 0.15 & 0.2]^T \\ \mathbf{h}_{s2} = [0.05 & 0.05 & 0.4]^T \end{cases}$$

然后对词典模型的计算结果的方差分布划分为两个区间：

$$\sigma_d^2 \in [b_0, b_1) \cup [b_1, b_2)$$

其中  $b_0 = 0, b_1 = 0.15^2, b_2 = \infty$ ，两个区间对应的权值分别为：

$$\begin{cases} \mathbf{h}_{d1} = [0.2 & 0.3]^T \\ \mathbf{h}_{d2} = [0.1 & 0.4]^T \end{cases}$$

动态权重的多模型融合的词相似度计算在使用以上动态权重时，对 PKU-500 中人工打分的斯皮尔曼等级相关系数为 0.568。通过对比发现本文的方法比 NLPCC-ICCPOL 2016 评测比赛中第一名的方法高出 9.6%。以上结果表明，在多模型相融合中，引入动态权重的策略可以有效提高词相似性的计算效果。

## 3.3 本章小结

本章介绍了多模型相融合的词相似性分析的方法，将 Word2Vec、HowNet、同义词词林扩展版三个模型融合在一起，通过动态权重的方法提高了词相似性的准确率。对于 PKU-500 数据集，本章所采用的多模型相融合的相似性分析，获得 0.568 的斯皮尔曼等级相关系数，与 NLPCC2016 第一名的结果相比提高了 9.6%。本章的各模型结果的动态权重策略还有进一步优化的空间，选取更合理的权重可以进一步提高词相似性的准确率。

## 第 4 章 网络舆情分析系统的实现

### 4.1 网络舆情分析系统需求分析

网络舆情分析系统需要具备以下功能：

- (1) 使用者输入来自网络的语料，系统对语料进行分词计算并反馈分词结果；
- (2) 使用者输入词语，将与该词意义最相似的 5 个词反馈给使用者；
- (3) 使用者输入一对词语，将这一对词的相似性反馈给使用者。

这些功能可以为使用者提供全面的网络语料处理，并将量化结果返回给使用者，通过这些功能，就可以分析网络语料的实际含义，有效地分析网络舆情的实际情况，最终实现网络舆情分析系统的设计目。

### 4.2 网络舆情分析系统的实现过程

本系统为了实现这些功能，首先需要保证语料库的实时性，所以语料库不断地对最新的网络语料进行采集并将其存储到数据库中，然后将会对收集到的语料进行处理并使用深度学习算法进行计算，最后要能够将计算的结果通过交互界面反馈给使用者。

针对以上这些需求，本系统将使用网络爬虫、MySQL 数据库、词相似性算法和用户交互界面编程等工具分别实现网络语料数据的收集，网络语料库的搭建，词相似性的计算和可视化交互界面的实现。

网络舆情分析系统将在 Windows10 上开发，并使用 Python 语言进行编程。Windows 系统的使用十分广泛，在桌面操作系统上有 80%以上的占有率，所以基于 Windows 开发的软件可以兼容更多使用者的设备。Python 编程语言在各种系统上都有很好的兼容性，编写难度也不高，编写出的程序非常简单且不需很高的硬件配置就可以流畅运行，并且在各种主流系统如 Windows 和 Linux 上基于 Python 编写的程序只需修改少量的代码甚至不需修改代码就可以实现移植。所以采用 Python 在 Windows 上编写软件将使软件具有很好的泛用性和兼容性。

#### 4.2.1 网络爬虫设计与搭建

网络社交语料数据在网络舆情分析系统中是计算的基础，所以必须要获取实

时数据。采用网络爬虫来进行语料的收集主要有两个原因，一是各个社交平台不一定都有应用程序接口(Application Programming Interface, API)接口，所以编写合适的网络爬虫可以有效的获取各网络社交平台的语料数据，二是经过编写的网络爬虫可以实现自动化数据获取，现今互联网的社交平台极多，语料数据的数据量极大，如果不能全自动获取语料，而是由人工操作，那么将带来极大的工作量，并影响数据的时效性。所以本系统将编写一个可以自动获取网络语料数据的网络爬虫程序，该网络爬虫程序将使用 Scrapy 框架来搭建。

Scrapy 框架是基于 python 的爬取网站并从中提取结构化数据的应用程序框架，Scrapy 的一个重要的特点就是其仅仅是一个框架，可以根据使用者的实际需求进行修改和自定义，由于网络社交平台的网页设计各不相同，所以需要根据各网站的具体情况修改网络爬虫程序，这个时候 Scrapy 可高度自定义的优势就体现了出来，只需要修改其中少量的模块就能完成任务。Scrapy 框架的架构如图 4.1 所示：

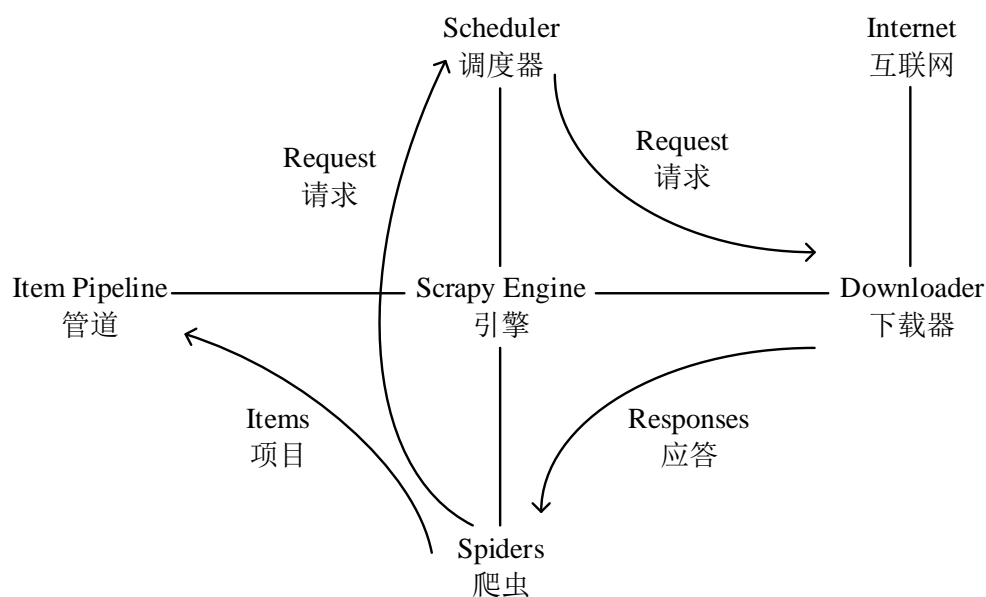


图 4.1 Scrapy 框架的架构

其中引擎负责调度器、下载器、爬虫、管道之间的信号与数据交换等。调度器负责接收引擎传送的请求，并按照一定的方式进行整理、排列和入队，当引擎需要时，交还给引擎。下载器负责下载引擎发送的所有请求，并将其获取到的应答交还给引擎，由引擎交给爬虫来处理。爬虫负责处理所有应答并从中提取并分析数据，获取项目需要的数据，并将需要跟进的 URL 提交给引擎，再次进入调度器。管道负责处理爬虫中获取到的项目并进行后期处理。下载中间件可以当作是一个可以自定义扩展下载功能的组件。爬虫中间件是一个可以自定义扩展和操作引擎与爬虫通信的功能组件。

本系统的网络爬虫将使用基于 Python 的 Scrapy 网络爬虫来爬取网络语料数据，接下来将以对贴吧数据进行爬取的贴吧语料网络爬虫为例来介绍网络爬虫的搭建过程。首先分析贴吧的语料结构，贴吧的语料为两层树状结构，一篇帖子包括若干楼层的回复，每个楼层下有若干楼中楼。这种树状结构，需要针对每层编写不同的爬取规则，对于帖子下所有楼层的回复需要爬取回复的楼层编号、楼层 id、回复者用户名、楼层内容、发布时间以及楼层回复数量。而对于楼中楼，需要爬取楼中楼 id、楼中楼回复者用户名、楼中楼内容以及楼中楼发布时间。对于这些数据，网络爬虫将访问网页并从网页源代码中获取，网页代码在相关数据上都遵循一定的规则，例如楼层内容源代码如下所示：

```
<div id="post_content_130484460658" class="d_post_content j_d_post_content "
style="display:;> 嗯嗯</div>
```

所有的回复均遵守这种编写规则，所以采用正则表达式就可以自动匹配出所有的回复，正则表达式获取数据的代码如下所示：

```
content = floor.xpath("../../../div[contains(@class,'j_d_post_content')]").extract_first()
```

其他所需的数据也和楼层内容一样遵循一定的代码格式，采用类似的正则表达式就可以自动的收集所有的信息。在网络爬虫获取数据时还有一个重要部分就是能够实现自动翻页，贴吧的每个帖子一般包括多页，所以要能够获取每一页的 URL，而一般网页的 URL 也是以相同规则写的，采用正则表达式找出每页的 URL 后即可实现翻页。

对于语料数据还有两个需要关注的问题，一是要爬取热门的语料来保证其时效性，二是要能过滤语料中无用数据的功能。对于保证语料时效性网络爬虫必须要具有按一定规则获取语料的功能，比如在爬取命令中加入爬取页面的范围、爬取帖子的类型和爬取的用户类型。而过滤语料中无用数据是因为网络交流时，用户的发言大多不会遵守常用的语言规则，比如在语句中加入大量空格、表情、图片等无用数据，所以必须编写一定的规则过滤这些部分。

为了实现这些功能，在编写爬虫时加入了一些规则，并在运行命令中增加了这些规则，网络爬虫的运行命令如下所示：

```
Scrapy run 篮球 -gs -p 5 12 -f thread_filter
```

这一命令可以实现网络爬虫的运行、爬取范围、爬去规则等条件的设置。通过运行命令的构建，只需要使用简单的一条命令就可以实现获取用户所需网络语料的任务。

#### 4.2.2 数据库设计与数据的收集处理

本系统采用数据库来存取语料数据，由于网络语料数据的特点是数据量大但数据结构简单，所以对数据库的要求是体积小、结构简单、运行速度快以及容易使用，最好还要能在多系统上运行并支持多种编程语言。根据这些要求，在现有的各种数据库中 MySQL 这种关系型的二维数据库最为合适，适合网络语料数据库搭建的任务。

MySQL 是一种关系型数据库管理系统(Relational Database Management System, RDBMS)，采用结构化查询语言进行数据库的管理，关系型数据库管理系统将数据保存在不同的表中而非集中储存在一起，使得数据库可以灵活并快速地存取数据。RDBMS 中数据是以表格形式出现，表格中的每行为各种记录名称，每列为记录名称所对应的数据域，这种结构很适合网络语料库的构建，例如一个网络语料一般包括发布时间，发布人员，语料内容等多种元素，一个二维的表格就可以很好地将每一条网络语料详细的记录下来。同时 MySQL 也支持 python 语言，由于网络爬虫 Scrapy 框架也是基于 python，那么就可以使用同一种语言完成两个部分的编写，降低系统在编程语言上的复杂性，同时 python 和 MySQL 都具有很好的系统兼容性，采用 MySQL 数据库和 python 可以不需要很复杂的移植就能在常用的操作系统如 Windows 和 Linux 上使用。所以本系统采用 MySQL 数据库可以在实现任务的同时保证数据库构建的效率，体积和兼容性。

MySQL 数据库在 Python 中有相应的插件可以使用，并且 Scrapy 也与 MySQL 相兼容，可以直接在 Scrapy 的 pipelines 中编写相应的程序来将网络爬虫中获取的数据传入 MySQL 数据库中，pipelines 中部分代码如下所示：

```
def __init__(self, settings):
    dbname = settings['MYSQL_DBNAME']
    tdbname = settings['TIEBA_NAME']
    if not dbname.strip():
        raise ValueError("No database name!")
    if not tdbname.strip():
        raise ValueError("No tieba name!")
    self.settings = settings
    self.dbpool = adbapi.ConnectionPool('MySQLdb',
        host=settings['MYSQL_HOST'],
        db=settings['MYSQL_DBNAME'],
        user=settings['MYSQL_USER'],
```

```
passwd=settings['MYSQL_PASSWD'],
charset='utf8mb4',
cursorclass = MySQLdb.cursors.DictCursor,
init_command = 'set foreign_key_checks=0' #异步容易冲突
)
```

然后语料数据将以表格的形式被存入 MySQL 数据库中，数据结构如图 4.2 所示，其中每一行是贴吧帖子中每个楼层所需要的全部数据，包括这一楼层的 id，用户名，用户内容等信息。这些数据以表格形式存在于数据库中使用户将很容易获取他们想要的信息，也可以便捷的提取出语料数据交由自然语言处理算法来进行处理。

id	author	content	time	post_id
23273742437	时间泛黄了微笑	习惯了 自己什么都不是 拿球就是不给别人 乱丢 还说你不丢抢篮板	2012-08-19 10:00:00	2210867325
23333262515	yuanjiangjun	这我练过一次，练完后弹跳短时间内增强了不少，但是感觉对膝盖的伤害非常大	2012-08-20 18:47:16	7294921749
23340902141	958765649	要是扔出街边了咋办，，	2012-08-20 21:49:28	13704039192
23349829554	纪代沂焱吧	你在旁边支个摊子，，兼职卖水，，生意红火啊，，，	2012-08-21 05:48:06	13704272625
23362278426	永远亿光年	<a href="https://gsp0.baidu.com/5aAHeD3nKh12p27j8lqW0jdnxx1xbK/tb/editor/images/">https://gsp0.baidu.com/5aAHeD3nKh12p27j8lqW0jdnxx1xbK/tb/editor/images/</a>	2012-08-21 13:23:42	23362167358
23362590262	快乐运动身体好	@永远亿光年 在我的亲自指导下你一定可以的...	2012-08-21 13:31:24	23362167358
23362727108	永远亿光年	回复 快乐运动身体好： <a href="https://gsp0.baidu.com/5aAHeD3nKh12p27j8lqW0jdnxx1xl">https://gsp0.baidu.com/5aAHeD3nKh12p27j8lqW0jdnxx1xl</a>	2012-08-21 13:34:52	23362167358
23363838870	密小小	太长 没看。	2012-08-21 14:03:31	23362167358
23370128870	填不完的缺口	太长 没看	2012-08-21 16:51:41	23362167358
23394592700	3Q很难说	看过 好帖子	2012-08-22 09:34:23	10472814204
23409137964	fifox	只能说楼主没有道德观，符合道德观的就是正义的，反之就是邪恶的。	2012-08-22 15:59:37	17296540497
23477221599	只恨此花飞尽	回复 LL破晓Y :这么神?	2012-08-24 10:17:44	7294910169
23494794085	3Q很难说	乔神	2012-08-24 18:33:29	7295109890
23540513618	牛犊的青春	表示高一刚入学摸不上板 高一打了多半年 高二了能摸环下面 ~ ~ ~ 多打也很有用 ~ ~	2012-08-25 21:36:00	7294921749
23575259851	冷羽翎清秋	回复 牛犊的青春 :你说下蹲的时候大腿能动吗?	2012-08-26 20:04:46	7294921749
23576081702	收服女流氓	这丫的坐球! 坐球是对球的 不尊敬。。。	2012-08-26 20:24:12	7295073722
23602404582	呱呱呱呱不倒花	有点凶 其实日和星长的挺温柔的	2012-08-27 14:10:10	14343043345

图 4.2 MySQL 数据库

贴吧语料网络爬虫是为贴吧语料数据爬取而专门编写的爬虫程序，整个系统中包含多个网络爬虫来有针对性的爬取不同社交平台的语料数据，使用这些网络爬虫可以获取到用户想要的各种语料。

### 4.2.3 词相似性分析设计与结果量化的实现

在计算词相似性时将使用动态权重的多模型相融合的算法，该算法同时使用了多种模型，其中词典模型对网络中的规范表达支持较好，而统计模型对非规范表达有很好的支持。在计算时该算法会根据词语的特点，选取不同的权重配置，如果是规范表达会给词典模型分配更高的权重，而非规范表达则会给统计模型分配更高的权重。这是因为规范表达大多都可以被词典模型所覆盖，并且词义和词型也都符合汉语言的标准习惯，所以词典模型得出的结果更能反映规范表达的词相似性，而非规范表达实时性比较强并且不一定符合传统的汉语言规则，所以词典模型无法覆盖，很多词在作为非规范表达时被赋予了更多的含义，针对这种情况所提出的

统计模型就能较好地实现高覆盖率和准确率。

通过使用动态权重的多模型相融合的算法，大幅度提高了词相似性计算的准确性，保证了系统在使用时能够更加准确的识别网络语料中词语的实际含义，实现网络舆情有效分析的目的

在词相似性计算时将采用动态权重的多模型融合的词相似性计算方法，在第三章中已经介绍了该算法的整体结构，现在将根据本系统的实际需求来对算法中的各种参数进行设置，在该算法中会使用 Word2Vec 模型、HowNet 以及《同义词词林扩展版》来计算词相似性，其中 HowNet、《同义词词林扩展版》将沿用第三章中所采用的参数，而 Word2Vec 模型并未在第三章详述参数的设置，所以在本节将详细说明 Word2Vec 模型参数的设置及这样设置的原因。

Word2Vec 模型中有几个比较重要的参数和计算方法，参数上有滑动窗口的窗口值、批处理次数、嵌入维度和迭代次数，在计算方法上有古德-图灵估计、优化方法的选择、损失函数的选择。下面将详细介绍这些参数和计算方法。

滑动窗口的窗口值是 Word2Vec 模型的一个重要参数，因为在 Word2Vec 中每个词在计算时是要根据其上下文来计算每个词的词向量，一般来说窗口值越大计算的结果就越准确，但窗口值的增大会使计算消耗的资源指数级增大，并且窗口值的增大在超过阈值后计算效果就不会显著提升，所以选择一个合适的窗口值可以实现计算效果和计算效率的平衡。在本系统中根据硬件平台的计算力并经过多次试验后，发现将窗口值设置为 3 时有比较好的效果。

批处理次数、嵌入维度和迭代次数在 Word2Vec 模型中被称之为超参数，批处理次数决定了每一次送入计算的词的个数，每次送入计算的词越多迭代次数就会变少，计算速度会增加但计算效果会变差，而嵌入维度是计算所得的每个词向量最终的维度，维度越高计算效果越好，但计算速度会下降。一般情况下批处理次数与嵌入维度取相同值。在本系统中经过多次试验，发现这两个参数设置为 200 时效果最好。迭代次数时指一批数据多次计算的次数，迭代次数越多计算结果越好，但计算速度会下降，迭代次数与批处理次数相关，迭代次数为总词数与批处理次数的商，所以迭代次数由总词数与批处理次数来决定。

古德-图灵估计是统计学中的一个重要方法，对于一个语料库，其中有些词可能不在其中，但是并不代表这些词不存在，可能只是这些词未被收集到，所以对于这些词我们并不能直接将其出现的概率定为 0，这会使得数据不平滑。古德-图灵估计的原理是，对于我们没有观察到的事物，不等于其不存在或者说发生的概率为 0，所以需要从概率总量中分配一个小的比例到这些没有观察到的事物中，通过这种方法使得数据变得平滑。古德-图灵估计的概念图如图 4.3 所示：

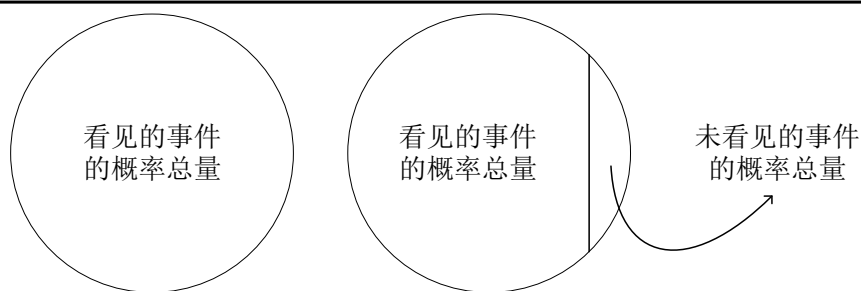


图 4.3 古德-图灵估计的概念图

优化算法将选择自适应学习优化器，本系统采用了 Adagard 算法，该算法的主要优点是无需手动调整学习率。算法对每一个参数做出单独的调整，让学习率适应参数，对于出现次数较少的特征对其采用较大的学习率，对于出现次数较多的特征对其采用较小的学习率，这种优化算法在处理语料数据时较为合适。损失函数将使用噪音对比估计（NCE, Noise Contrastive Estimation），这种算法能够用来解决神经网络的复杂计算问题，适合解决自然语言处理问题。结果量化将采用第二章中的所介绍的方法来完成。

#### 4.2.4 可视化界面设计与实现

为了使用者可以更加直观的使用网络舆情分析系统，必须要设计一个方便使用的可视化界面，这一界面将包括不同界面的切换标签、输入、输出、结果反馈等功能。针对这些需求将开发出一款易于使用的软件。

网络舆情分析系统软件将包括三个标签页，分别是两词相似度、语料分词和相似词 top5。其中两词相似度部分将包括两个输入框、一个按钮和一个数据显示框，输入框用于输入需要计算的两个词，按钮按下时会在数据显示框中显示计算结果。语料分词部分包括一个输入框、两个选择键、一个按钮和一个数据显示框，输入框用于输入需要进行分词的语料，选择按钮可以选择要进行分词的模式，按钮按下后会在数据显示框中显示分词结果。相似词 top5 部分包括一个输入框、一个按钮和一个数据显示框，输入框用于输入需要进行计算的词语，按钮按下后会在数据显示框中显示与该词最相似的 5 个词以及这 5 个词的相似度。该软件还包含工具栏，工具栏中包含两个功能，一个是选项，一个是帮助，选项中包括两个功能，一个是登录，登录功能可以让使用者登录账户记录计算数据，退出可以退出软件，帮助中的关于功能会介绍软件的使用方法。通过这一具有可视化界面的软件，使用者可以便捷的使用网络舆情分析系统，并直观的得到各功能的数据反馈。

根据系统需求和设计，将使用基于 python 的 Tkinter 插件来编写具有可视化界面的软件，软件包括三个标签页，分别是两词相似度、语料分词和相似词 top5。其



中两词相似度如图 4.4 所示，其中包括两个词语的输入框、一个显示结果按钮和一个结果显示框。语料分词如图 4.5 所示，其中包括一个语料输入框、两个选择键、一个显示结果按钮和一个结果显示框。语料分词如图 4.6 所示，其中包括一个语料输入框、一个显示结果按钮和一个结果显示框。软件的工具栏包括两个按钮，一个是选项一个是帮助，这两个按钮点击后如图 4.7，图 4.8 所示显示下拉菜单。

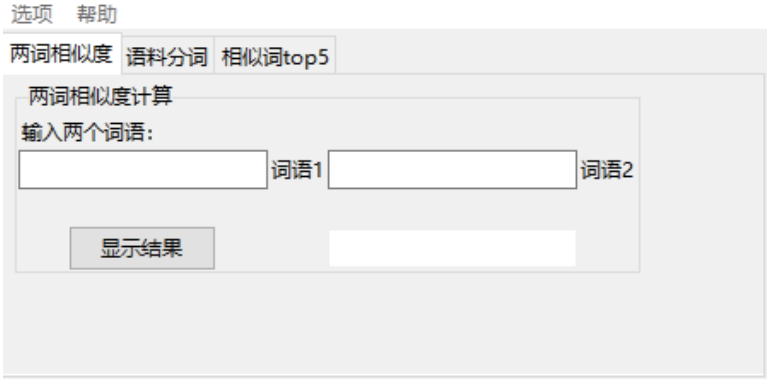


图 4.4 两词相似度标签页

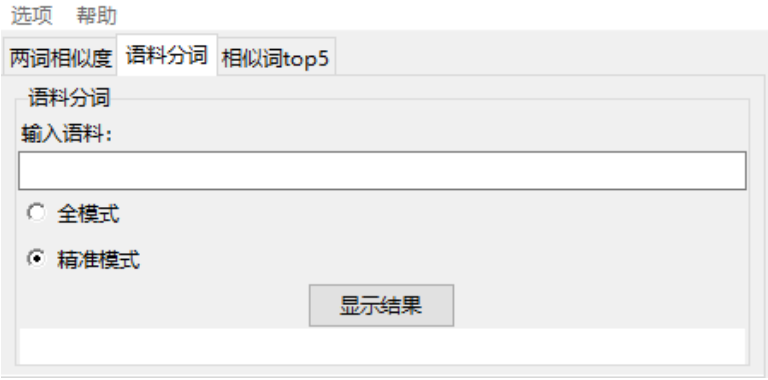


图 4.5 语料分词标签页

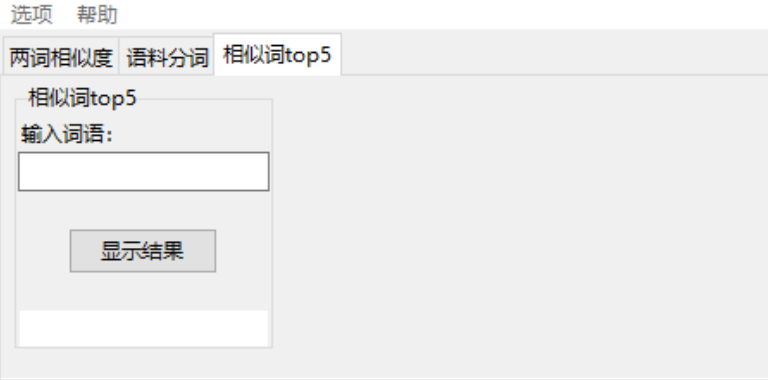


图 4.6 相似词 top5 标签页

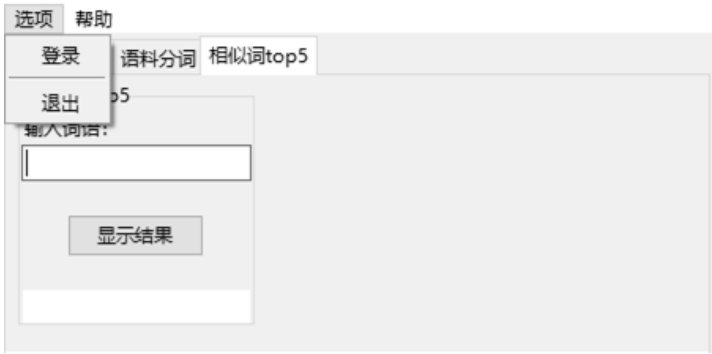


图 4.7 选项按钮下拉页

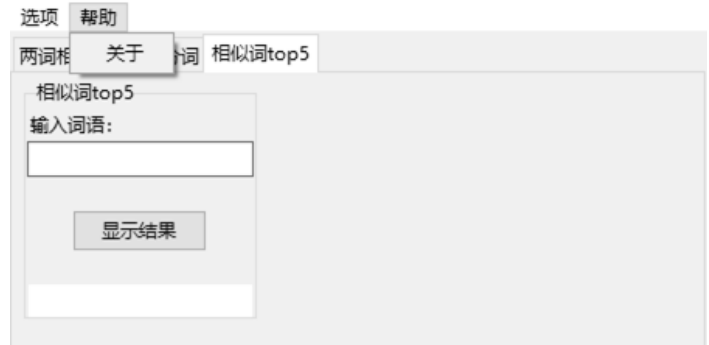


图 4.8 帮助按钮下拉页

以上为网络舆情分析系统可视化界面软件的全部内容，使用这个软件可以让用户能够方便的使用该系统。

### 4.3 系统测试与结果分析

本节将会对系统进行测试并分析系统的性能，能否到达使用者预期所需的各项功能。系统将会测试两词相似度、语料分词和相似词 top5 三个标签页下的功能。

首先是两词相似度功能，实际使用如图 4.9，图 4.10，图 4.11 所示：



图 4.9 两词相似度 1

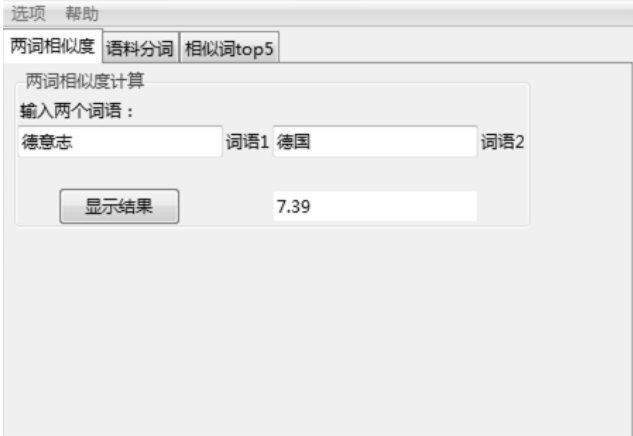


图 4.10 两词相似度 2



图 4.11 两词相似度 3

通过对以上这些词对的计算可以发现，网络舆情分析系统有比较好的计算效果，不仅是对规范表达，对各种网络中出现的非规范表达都有较好的覆盖率，可以给使用者有效的结果反馈。

对于语料分词功能，实际使用如图 4.12 和图 4.13 所示：



图 4.12 语料分词 1



图 4.13 语料分词 2

通过对示例语料的分词处理,两种分词模式都可以很好的完成分词任务,分词结果符合语言习惯,用户也可以按需求选择不同的分词模式来满足分词任务。

对于相似词 top5 功能,实际使用如图 4.14,图 4.15 和图 4.16 所示:



图 4.14 相似词 top5-1



图 4.15 相似词 top5-2



图 4.16 相似词 top5-3

通过对示例词的计算可以发现，网络舆情分析系统计算效果良好，无论是常规词还是网络中的新词都能够有效的支持，当使用者不了解某个词时，通过这一功能将与该词相似的词显示出来，在计算结果中出现易于理解的同义词，使用者就可以了解该词的含义。

测试过网络舆情分析系统软件的各项功能后发现可以满足系统的需求，使用者可以通过网络舆情分析系统完成相应的任务。

#### 4.4 本章小结

本章详细介绍了网络舆情分析系统的搭建过程，首先分析了系统的需求，然后对系统的各项功能按需求与设计进行实现并搭建一个完整的系统，最后对系统进行测试，满足了系统的需求。

## 第5章 总结与展望

### 5.1 全文总结

本文主要开发了一个基于深度学习的网络舆情分析系统，该系统包括实现网络语料数据收集，语料数据库搭建，语料数据处理和用户可视化界面软件编程等任务，其中本文重点研究了语料数据处理中的基于深度学习的自然语言处理，提出了动态权重多模型融合的方法。本文将该算法作为网络舆情分析系统的核心算法运用于词相似性的计算中，实现了网络舆情分析系统的自适应深度学习。本文还为系统开发了一套完整的可视化界面供使用者使用。

论文研究的主要工作如下：

#### (1) 网络语料数据的爬取

通过使用基于 Python 的 Scrapy 模块，针对各大社交平台编写相应的网络爬虫程序，实现语料数据在无人值守的情况下的自动爬取，为语料库的建立打下了基础。

#### (2) 数据库的搭建

使用 MySQL 数据库来存储实时的网络语料数据，这些数据按照一定的规则和模式存储在服务器中，这些数据可以有序的在后续使用中获取到，可以增加整个系统的效率。

#### (3) 词相似性的计算

词相似性的计算使用了动态权重多模型融合的方法，该方法是文本的重点研究内容，该算法将自然语言处理中常用的几种算法结合起来，并引入深度学习来应对语料数据大量增加的问题，然后在各模型的计算结果间采用动态权重的方法，根据语料的特点选取不同的权重来得到最终的结果。最后根据各模型的特点，采用不同的方法来量化计算结果，将无法描述的两词相似程度变为直观的数字，使得词相似性可以被用户理解。

#### (4) 可视化界面的设计

为了使该系统能够让用户直观的使用所以开发了相关的可视化界面。可视化界面使主要的三个功能都可以通过简单的操作实现，增加了系统的易用性，并直观的显示计算结果。对系统的测试后，系统的各项功能都可以实现，说明该系统可以满足本论文的各种需求。

## 5.2 研究展望

存在的问题及展望：

(1) 本系统的数据获取方式可以进行改进，通过对各个网络平台的数据结构研究，编写合理的网络爬虫程序，能够进一步提高网络爬虫自动化爬取网络数据的效率，使整个系统可以更高效的获取实时数据；

(2) 本文的各模型之间动态权重的策略还有进一步优化的空间，选取更合理的权重可以进一步提高词相似性的准确率。并且在 Word2Vec 模型的参数选择上也有进一步改进的空间。

(3) 本系统的交互界面还可以进一步的美化，使其具有更好的交互性。并且本系统还可以将更多的可更改选项提供给用户，使其可以自行调整如数据库，自然语言处理算法中的各种参数，使用户可以结合自己的硬件环境来选择适合的参数以达到最佳的效果。

## 参考文献

- [1] Harris Z S. Mathematical structures of language[M]. Florida: Krieger Publishing Company,1968.
- [2] 吴军. 数学之美[M]. 北京: 人民邮电出版社,2014.11.
- [3] Curran J R. Ensemble Methods for Automatic Thesaurus Extraction[C]//Conference on Empirical Methods in Natural Language Processing, 2002:222-229.
- [4] Agirre E, Alfonseca E, Hall K, et al. A study on similarity and relatedness using distributional and ordnet-based approaches[C]//Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics,2009:19-27.
- [5] Turney P D. Domain and function: A dual-space model of semantic relations and compositions[J]. Journal of Artificial Intelligence Research,2012,44:533-585.
- [6] Finkelstein L, Gabrilovich E, Matias Y, et al. Placing search in context: The concept revisited[J]. ACM Transactions on Information Systems,2002,20(1):116-131.
- [7] Landauer T K, Dumais S T. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge[J]. Psychological Review,1997, 104(2):211-240.
- [8] Bruni E, Boleda G, Baroni M, et al. Distributional semantics in technicolor[C]//Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. Association for Computational Linguistics,2012:136-145.
- [9] Jia Y, Li Y, Zan H. Acquiring Selectional Preferences for Knowledge Base Construction [C]//Workshop on Chinese Lexical Semantics. Springer, Cham,2017:275-283.
- [10] Zhang Y, Clark S. Syntactic processing using the generalized perceptron and beam search[J]. Computational linguistics,2011,37(1):105-151.
- [11] Richardson S D, Dolan W B, Vanderwende L. MindNet: acquiring and structuring semantic information from text[C]//Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2. Association for Computational Linguistics, 1998:1098-1102.
- [12] 孙景广,蔡东风,吕德新,等. 基于知网的中文问题自动分类[J]. 中文信息学报,2007,(1):90-95.
- [13] 王松松,高伟勋,徐逸凡. 结合路径与词林编码的词语相似度计算方法[J]. 计算机工程, 2017.
- [14] 唐怡,周昌乐,练睿婷. 基于 HowNet 的中文语义依存分析[J]. 心智与计算,2010, (2) :109-116.
- [15] 赵倩倩. 词语相似度计算及其在语义选择限制知识获取中的应用研究[D]. 郑州大学,2018.
- [16] 彭琦,朱新华,陈意山,等. 基于信息内容的词林词语相似度计算[J]. 计算机应用研究,2017.
- [17] 贾玉祥,王浩石,咎红英,等. 汉语语义选择限制知识的自动获取研究[J]. 中文信息学报,201



- 4,28(5):66-73.
- [18] Li W, Liu T, Zhang Y, et al. Automated generalization of phrasal paraphrases from the web[C]//Proceedings of the 3rd International Workshop on Paraphrasing(IWP2005),2005:49-56.
- [19] 买志玉,金澎,曾赛.基于大规模语料库的汉语词相似计算[J]. 中原工学院学报,2010, 21(3): 45-50.
- [20] Le Q, Mikolov T. Distributed representations of sentences and documents[C]//International Conference on Machine Learning,2014:1188-1196.
- [21] Rong X. word2vec parameter learning explained[J]. arXiv preprint arXiv,2014:1411.2738.
- [22] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[C]//Proceedings of the International Conference on Learning Representations, 2013.
- [23] Zeng X, Yang C, Tu C, et al. Chinese LIWC lexicon Expansion via Hierarchical classification of word embeddings with sememe Attention[C]//Proceedings of AAAI 2018,2018.
- [24] Heylen K, Peirsman Y, Geeraerts D, et al. Modeling word similarity: An evaluation of automatic synonym extraction algorithms[C]//Proceedings of the 6th International Language Resources and Evaluation,2008,3243-3249.
- [25] Turney P D. Similarity of semantic relations[J]. Computational Linguistics,2006,32(3):379-416.
- [26] Perozzi, Bryan, Al-Rfou, Rami, Skiena, Steven. DeepWalk: Online Learning of Social Representations[C]//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM,2014:701-710.
- [27] Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model[J]. The Journal of Machine Learning Research,2003(3):1137-1155.
- [28] 石静,吴云芳,邱立坤,等. 基于大规模语料库的汉语词义相似度计算方法[J]. 中文信息学报, 2013,27(1):1-6.
- [29] Jin P, Carroll J, Wu Y, et al. Distributional Similarity for Chinese: Exploiting Characters and Radicals[J]. Mathematical Problems in Engineering,2012,(2012-08-15),2012, 2012 (6):2301-2314.
- [30] Guo S R, Guan Y, Li R. Chinese word similarity computing base in combination strategy[C]//Proceedings of NLPCC 2016, Lecture Notes in Artificial Intelligence,2016, 10102: 744-752.
- [31] 梅家驹,竺一鸣,高蕴琦,等. 同义词词林[M]. 上海: 上海辞书出版社,1983.
- [32] 董振东,董强. 知网[DB/OL]. <http://www.keenage.com>.
- [33] 马永起,韩德培,蒙立荣,等. 基于 How-net 的词语语义相似度算法[J]. 计算机工程,2018,44 (06):151-155.
- [34] 朱靖雯,杨玉基,许斌,等. 基于 HowNet 的语义表示学习[J]. 中文信息学报,2019,33(03):33-41.
- [35] 刘群,李素建. 基于《知网》的词汇语义相似度计算[J]. 中文计算机语言学,2002,7(2):59-76.
- [36] 吕立辉,梁维薇,冉蜀阳. 基于词林的词语相似度的度量[J]. 现代计算机(专业版),2013(01):3-6+9.

- [37] 哈工大社会计算与信息检索研究中心. 同义词词林扩展版[EB/OL]. [http://www. datatang.com/data/42306/](http://www.datatang.com/data/42306/).
- [38] Faruqui M, Dodge J, Jauhar S K, et al. Retrofitting word vectors to semantic lexicons [C]//Proceedings of the 2015Annual Conference of the North American Chapter of the ACL (NAACL 2015),2015:1606-1615.
- [39] 陈慧,田大钢,冯成刚. 多种算法对不同中文文本分类效果比较研究[J]. 软件导刊,2019,18(05):73-78.
- [40] 向春丞,穗志方,詹卫东. HowNet 与 CCD 映射方法研究[J]. 中文信息学报,2015,29(3):44-15.
- [41] 陈宏朝,李飞,朱新华,等. 基于路径与深度的同义词词林词语相似度计算[J]. 中文信息学报, 2016, 30(5):80-88.
- [42] Wu Y F, Li W. Overview of the NLPCC-ICCPOL 2016 shared task:Chinese word similarity measurement[J]. Lecture Notes in Artificial Intelligence,2016,10102:828-839.
- [43] Zou Y F, Ouyang C P, Liu Y B, et al. A Similarity Algorithm Based on the Generality and Individuality of Words[C]//Proceedings of NLPCC 2016,Lecture Notes in Artificial Intelligence,2016,10102:549-588.
- [44] Liu J, Xu J, Zhang Y. An approach of hybrid hierarchical structure for word similarity computing by HowNet[C]//Proceedings of the 6th International Joint Conference on Natural Language Processing.
- [45] Iman R L, Conover W J. A distribution-free approach to inducing rank correlation among input variables[J]. Communications in Statistics-Simulation and Computation,1982,11(3):311-334.
- [46] Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and their Compositionality[C]//Proceedings of the 27<sup>th</sup> Annual Conference on Neural Information processing Systems,2013b: 3111-3119.
- [47] Kiela D, Clark S. A systematic study of semantic vector space model parameters [C]//Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC),2014:21-30.
- [48] 刘青磊,顾小丰. 基于《知网》的词语相似度算法研究[J]. 中文信息学报,2010,24(6):31-36.
- [49] 董振东,董强. 知网和汉语研究[J]. 当代语言学,2001,3(1):33-44.

## 致 谢

本论文在导师万相奎教授的悉心指导下完成各项工作。

首先要感谢我的导师万相奎教授。他在生活中具有严以律己、宽以待人的崇高风范，在科研工作中具有精益求精的科学态度，严谨的作风，在教学工作中具有诲人不倦的高尚师德，他的人格魅力深深地感染了我。在研究生的学习生活中，万老师创造了优良的学习和工作环境，使得我们一心一意投入学习工作中。最后再次对这几年来万老师的培养、教育和支持表示感谢！

在我学习和生活上还要感谢李凤从老师、丰励老师的辛勤教导，感谢危竞、吴海波、杨辉、帅亮、陈瑞、刘翔宇、徐俊、严岳文、魏佳昕等同学的帮助以及感谢方丽雯的支持。最后对在生活中给予的关心的室友们表示感谢。

最后感谢的是我的父母。在十几年的求学中，父母给予了我他们最无私的关爱和支持。在生活和学习中的奋斗是对父母最好的回报，希望将来能获得更高的成就让父母为之骄傲。

感谢各位专家、教授百忙之中评阅我的学位论文以及聆听我的答辩！