# 1 引言

## 1.1 系统研究概述

### 1.1.1 课题研究现状与发展趋势

网络舆情是指由各种不同的社会团体所组成的公众群体和舆论代表，在某些特定的时间和空间或网络范围中，对自己关注的或与自身个体利益紧密相关的各种社会实践政府中策或社会问题的具有一定影响力和导向性的公共群体大众的意见的集合。网络舆情监测系统则随着计算机科学及互联网的发展而普及开来，通过特定网络数据采集程序来获取网络平台的舆论情况，并将所获取的网络数据 进行存储，以为之后的网络舆情数据分析来做好准备。在数据分析环节对所获取的舆情数据进行内容分析或文本数据挖掘，以便获取舆论情况的相关特征。

伴随着对网络舆情监测方面的研究的进展逐渐加深，对于网络舆情方面的现状的研究也不断的发展。网络舆情监测系统逐渐朝着大数据、实时性方面深，以及人工智能和人工神经网络的发展，舆情监测系统对与舆论情况的把握越来越准确。舆情监测系统所表现出来的对舆论监测以及动态把握的功能和作用有日益加强，甚至能通过对当前舆论情况的分析，而预测出未来一段时间内的舆论发展方向和趋势。

### 1.1.2 国内外研究状况

国外对舆情及大数据研究主要在概念研究和技术研究上，其中重点在技术研究。同时大部分的科学研究项目都集中在在对于大数据的影响及技术挑战上，相较于数据科学，其更重视的数据处理技术在工程实践中的应用，主要分析研究的分析算法和处理方法。同时因为一些相关的法律法规的约束，在舆情监测方面的成果和工程项目较少。

通过查询资料，我国对网络舆情研究及管理主要开始于本世纪初期，在对网络舆情的研究中，主要研究网络舆情的引导、控制、预警及对应急舆情情况的处理方面上。在舆情引导方面主要是从媒体作用、职业道德和公众素养三个角度进行；在控制方式方面则加强法规制度、技术、专业人才的培养等方面与地面策应与网络处理相结合建立了立体化网络舆情监控机制。对于预警机制则主要从（1）及时掌握舆情动态，重视舆情监测采集系统的建设;（2）注重参与引导网络舆论情况；（3）对于不良网络舆论情况制定建立合理的应对方式；（4）推进网络舆情分析处理的工作的效率。

## 1.2 系统开发的意义

如今的时代由于网络化、数字化和信息化的融入，不再像以前的生活那么单一化。人才招聘系统的开发具有很强的信息指导性特征，采用ASP.NET开发此系统，具有更灵活和方便的交互性，在Internet中实现信息及数据的搜索越来越容易，及时、全面、大量的获取人力资源信息，进行发布、浏览、搜索大大的给企业和用户提供便捷的开展人事咨询及交流等工作；使用SQL 2005数据库技术，对于人才招聘系统的信息的管理提供了方便，通过对实时的求职信息的数据分析及分类，便于个人及企业去参考和决策，避免浪费精力和资源，有效的增强个人及企业使用的成功性。

# 2 系统开发技术分析

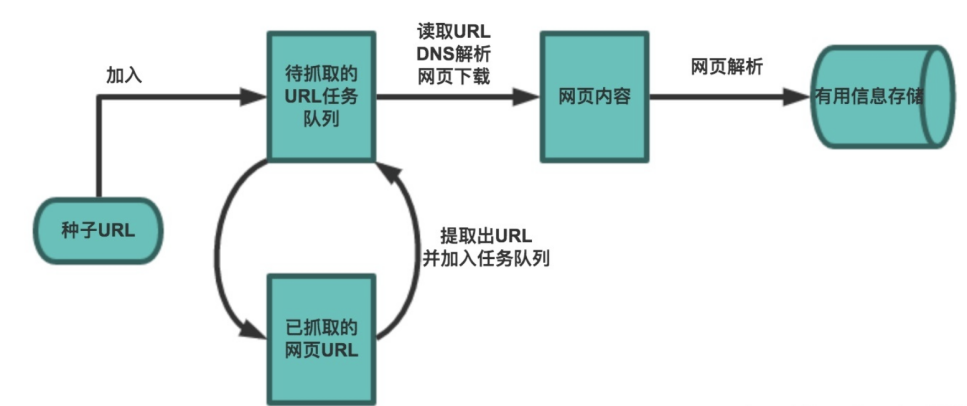
## 2.1 **社交网络数据采集**

网络数据的采集是通过网络数据采集程序模拟正常的浏览器访问来实现网络 请求的程序。可以类似的把互联网比作一张有这着各个节点相互连接的大网，而网络数据采集程序就相当于是在互联网网络上不断浏览爬行的爬虫，网页则是网上的节点。每当爬虫程序到达一个网络节点的时候就可以类比相当于浏览访问了该网络页面，并且采集到了这个网络节点上所具有数据信息。网络的数据的采集这是这样：将每个节点与节点之间的连线等效的视为互联网网络之中的每个网页与网页之间的链接关系，这样，当爬虫每访问过这张网上的一个节点之后，便可以通过网页之间的链接来探测并通过该节点上与其他节点的通路，从而顺着以该 节点的连线到达下一个相临的网络节点，即通过一个网页来获取此网页所相邻的之后网页，依次重复上述步骤并且将所有已经到达的网络节点标记，最终这样网上的所有节点都可以被爬虫全部遍历，从而获取网络上的数据。

### 2.1.1 网络爬虫原理

爬虫首要任务就是获取网页，这里的网页并不是一般我们在浏览器中所看到的网页视图，而是指的是网页未经渲染的源代码。网页源代码中包含着网页中的而大多数信息，所以只要可以获得网页的源代码，即可从中提取到所需信息。

在互联网上，一般的公开数据信息（各种网页）都是以http/https网络协议进行传输的。其主流的网络爬虫程序基本思路是：通过具有特定关键字和规则的URL把所有相关的html页面全部获取下来,然后通过解析html文本，将所获得的数据信息收集起来，再存储到数据库中。图1为本文所编写的网络爬虫工作流程结构图。



**图1 网络爬虫工作流程结构图**

### 2.1.2 爬虫的逻辑结构

一个最基本的网络爬虫程序的具体工作流程如下：

1. 通过模拟HTTP的请求操作来向目标服务器发送一个类似与浏览器的请求（requests），目标服务器所返回的响应（respond）便是网页的源代码。
2. 提取网页源码中的信息：在获取网页页面之后，接下的任务便是通过一定的html规则来提取目标数据。最常用的提取方法有：正则表达式、Beautiful Soup、CSS选择器以及Xpath等。
3. 保存数据：在从网页源码中提取数据之后，将数据存储到数据库之中。一般对于网页数据的特点可以采用非关系型数据库来增强数据的可调用星。将数存储到数据库中，以便为之后的数据分析工作做好准备。

### 2.1.3 网络爬虫技术

1）广度优先策略

广度优先策略(Breadth First Search Strategy, BFS)是根据目标对象层次目录的层次级数来进行搜索的策略，首先从当前目录的某一特定的网页开始寻找目录 级数较少的网页，当处于同一层次级数的所有网页寻找完成之后，在向层次更高 的目录进行搜索，直到所有层次遍历完成为止，广度优先策略可以确保能够搜寻 到当前覆盖网络的所有节点，即能网页覆盖范围的广度大；此外使用广度优先策略算法也可以控制对目标搜索的层次深度，以防止当网络深度过深时造成程序进入死循环状态。

2）正则表达式

正则表达式，又称规则表达式或正则集合。通过直观简洁的正则表达式，便 能够非常快速的实现一个复杂度高的项目业务逻辑需求，检索其目标文本之中的 特定信息，并且正则表达式的具有非常优秀的时间复杂度O(n)。其工作原理是通过有限状态自动机来实现匹配功能的。

3）Ajax页面解析技术

Ajax，全称为Asynchronous JavaScript and XML，是一种异步的JavaScript和XML网页刷新技术，是通过脚本语言JavaScript保证网页页面能够在不被整体刷新的同时保证网页链接地址无需更改或重新发送相关网络数据请求，而直接与目标服务器之间更新传递数据并渲染改变部分网页样式的动态刷新技术。将传统更新页面的方式由页面全局刷新变为利用JavaScript改变网页从而局部刷新，以更新网页的方式。

Ajax工作过程分为以下几个阶段：1) 发送请求。2) 解析内容。本地浏览器对服务器所返回的JSON数据内容进行解析。3) 渲染网页。经过JavaScript来对已解析完成的数据内容和网页样式来进行HTML页面构造。

### 2.1.3 非关系型数据库

非关系型数据库（NoSQL）是指无需遵守关系代数模式的数据库。非关系型数据库是指在数据存储的过程中无需按照ACID特性存储数据。非关系型数据库通常应用在高并发、数据存储提取量大的情况，同时非关系型数据库具有高拓展性和可用性的特征，为了解决对大量信息数据高并发进行读写速度方面的问题，而采用非关系型存储方式。

通常的非关系型数据库采用的数据模型有：1) 行列模式：将存储的数据按照类似于Excel表格中的行列排列，以一行或一列作为一个数据记录。常见的数据库有: Hbase。 2) 键值对模式：存储的数据以键值对的形式存放在数据库中。常见的数据库有Redis。3) 文档类模型：以文档集合的形式来存储数据，与“键值对”模式较为类似，但数据结构中存在集合这一结构。常见的有：MongoDB。

## 2.2 数据分析

数据分析是指通过使用适当的数据统计分析方法来对采集来的大量数据进行分析，剔除噪声数据并减少数据嫡增，提取有价值信息和探寻结论从而对所获取的数据加以详细分析和概括总结研究的过程。

### 2.2.1 词云介绍

词云是对网络数据中的文本信息进行数据分析的一种方法。通过对数据文本进行分词处理，并对其中出现的频率次数大的关键词予以视觉方面的突出，形成关键词图片,从而过滤掉大量的低频的文本信息词汇，是得可以仅通过词云来反映文本的关键主旨。由词语出现的频率来决定其在词云之中的位置、大小。

数据词云制作的过程大致可以分为一下儿步：

1. 明确词云主题和意义：判断词云的主题以及所希望表达的内容，为之后的词云制作明确方向。
2. 语料收集：收集要进行分析的文本数据。
3. 文本分词和关键字提取：将文本数据中的词语提取出来，并统计不同词语的出现频率

### 2.2.2 数据清洗

在正式的工程实践中，通过网络爬虫程序所获取的网页数据可能会包含一定的缺失值及噪声数据，此外也可能包含一定量的无效数据以及因为人工录入而导致的问题值，因此在数据分析的过程中首先就需要数据清洗。数据清洗是指对数据进行审查校验的过程，其清洗的目的在于数据去重、纠正错误数据以及数据低 端化处理，从而确保数据的完整性与一致性。数据清洗的流程大致如图2。



**图2 数据清洗流程**

2.3 数据可视化

可视化，就是将结构化的或者非结构化的数据转换成适当的可视化图表，以便将数据的特征可以直观地展现于人们面前。数据可视化普遍应用于各种报表或者说明展示性的文件中，借助使用直观的数据图表表现出数据背后所蕴含的信息, 显得简洁、直观。

数据可视化的表现形式的种类很多，其中各个可视化所侧重表达的重点各不相同。除了传统的饼图、柱状图、折线图等常见图形，还有气泡图、面积图等视 觉特效更好的图表。

### 2.3.1 Echarts 技术

Echarts，是一个由网页脚本语言JavaScript实现的可视化开源图形库,具有较强的兼容性和稳定性，可以流畅的运行在当今的绝大多数设备上，同时兼容当前多数网页浏览器，可以提供简洁直观、交互度丰富以及可以个性化制作的可视化图表。 除此之外，Echarts通过其内置的dataset数据属性可以支持直接读取使用包括二维表、元组以及key-value对等多种数据格式，此外通过设置encode属性(编码属性)就可以完成从数据到可视化图表的映射，化简了大部分应用场景下的数据转换 的步骤，而且具有极强的兼容性，此外具有较强的拓展性，可以借助多组件生成复杂的图表类型。

对于大数据可视化方面，Echarts有其特定的从底层描述二进制数据的类似数组的数据格式，以使在渲染大数据量的数据图表时可以占用更少的内存，从而大幅度的提升可视化应用的性能和速度。此外Echarts采用以数据驱动类型为其设计理念，图表可以随着数据的改变驱动图表展现的改变，无需在编码方面进行调整。因此可以非常简单快捷的实现数据动态化，只需要获取数据，输入数据，便可以生成数据图表.

### 2.3.2 Flask技术

Flask是一个基于Python的Web开发微框架。借助Flask框架可以更简单高效的完成网络Web应用的软件框架。Flask框架大致可以分为前端和后端两个部分。 Flask前端部分是通过Bootstrap框架与Jinjia模板语言来实现数据展示的，Flask中的app模块则负责后端功能，根据前端部分发送的数据请求，而从后端获取数据之后 通过路由（route）映射到前端。

2.4 系统运行、开发环境

本系统主要是针对网易新闻进行数据采集，通过网络爬虫来获取数据。具体的开发环境如下：

1）系统环境：Windows 10系统；

2）开发语言：Python 3.8，实现网络爬虫、数据分析及Web框架搭建；

3）开发平台：Pycharm，Python开发环境；

4）数据库：MongoDB，用于存储爬取的数据；

5）前端开发：JavaScript，用来开发并改进网页；

6）Web框架：Flask框架、Echart，实现数据图表化。

# 3 系统需求分析

## 3.1 需求分析概述

系统通过自动获取网页内容的程序来实现对网页信息的采集；通过改写网页数据请求头(request head)中的User-Agent、Referer及Cookie来实现网络数据的连续采集；并通过正则表达式和将HTML文档转换成一个树形结构的方法来实现网页数据的定位及获取。同时研究了XML及JSON页面分析技术和非关系型数据库(MongoDB)等数据采集存储技术。

再次，通过对获取的文本信息进行分词处理，提取高频词汇及其他信息来实现对数据的分析；通过利用文本信息时间来实现对舆情状况的中长期分析。

最后，通过搭建web网络框架，来建立网络服务器。除此之外通过html、Echart及JQuery等网络前端技术来实现数据的可视化展示。

## 3.2功能分析

网络舆情监测系统主要分为数据采集模块、数据分析以及可视化模块三个主要部分。每个模块的功能需求也各不相同，其中以数据采集模块的功能要求最多。下面是对各个功能模块的需求的具体分析：

1）数据采集模块

数据采集模块的功能是从互联网上获取数据信息。为了实现舆情监测的目的，数据采集将会对网易新闻的数据进行采集。采集的数据类型主要有两种：

a. 网易新闻首页各个发言贴的url、发帖人ID、发帖标题以及帖子留言数等信息进行采集。

b. 通过发言贴的url对各发言贴内部的发言内容和留言内容、留言图、留言人ID、留言时间以及留言设备型号进行采集。

在数据采集完成之后，将数据的格式转化为Json格式，同时存储到 MongoDB数据库中。

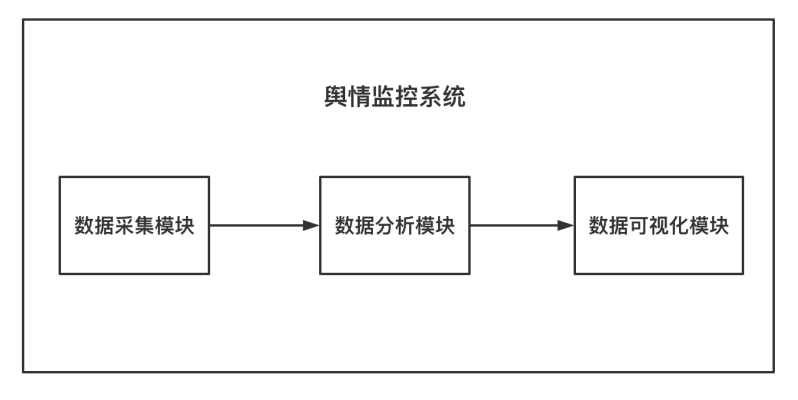
2) 数据分析模块

数据分析模块的主要功能是对已经采集的数据进行校验审查以及数据分析，该模块的功能需求主要可以分为数据清洗和数据分析两部分。数据清洗部分主要的需求就是通过清洗来实现数据去重、缺值补全以及剔除错误值。

数据分析则可以通过对数据消息文本进行分词、提取关键词来实预测分析；通过时间等数据，来统计舆论热度。此外，通过分析其他特定数据也可分析出其他信息。

3) 可视化模块

可视化模块的主要功能是将数据可视化，并且提供一个网络框架。为实现数据可视化，通过调用Echarts将数据以图表的形式展示出来。网络框架则通过Flask搭建一个供数据展示的Web网络，实现数据可视化以及前后端交互。



**图3 舆情监测系统结构图**

### 3.2.1 经济可行性分析

新系统的经济可行性分析的含义是，估计新系统开发和运行所需的费用，以及新系统的效益，将投资和效益进行比较，说明在经济上是合算的。

本系统投资低，只需配置一台稍好的电脑及相关的应用软件，此外本系统能够提高数据处理的及时性和准确性；降低了管理人员的劳动强度，提高了劳动效率；使用信息更加方便，从而使决策更及时、准确；取得了其他经济和社会效益。

本环节通过分析新系统的成本效益，其中包括估计项目开发的成本，开发费用和今后的运行、维护费用以及新系统将获得的效益，估算开发成本是否高于项目预期的全部经费。下面具体列出了成本和效益的分析：

1) 支出

现在电脑的应用已经得到了普及是基础的信息化设施，目前使用计算机进行软件开发，网络环境和基础设施能满足本系统的实施要求，所以无须另外的资金投入。

其它一次性支出：

由于本系统的开发是作为课程实践来进行的，无须资金投入，而且本系统所用到的软件成本投入也不高。因此软件设计和开发费可以承受。

经常性支出：

系统运行后的日常维护费用少。仅需简单的管理中心，该中心的工作人员可以负责本系统的维护工作。

2 ) 效益

本系统的效益可以从经济效益和社会效益两方面考虑。其社会效益包括系统投入运行后可以使管理实现科学化、规范化，增强用户对知识信息的管理能力，从而提高工作效率，从长远来看，也是信息化的必经之路。

由此可见系统的建设还是挺有发展前景的。

### 3.2.2 技术能力分析

根据本项目开发的功能需求，采用Flask与MongoDB数据库进行开发，因此在技术层面上，是完全可行的。

### 3.2.3 运行可行性分析

本系统的客户端采用的是图形界面方式，由于日常所使用的操作系统都是Windows系统，所以经过简单的培训就可以胜任新系统的操作工作。新系统的目标就是帮助用户实现资料的管理，并且新系统的运行不会对组织机构和管理方式产生大的影响，所以不会受到任何阻碍。

系统的设计上需要注重安全性和用户操作的流畅性。因此要在对用户的管理和资格审核环节做出限定，只有经过审核确认的用户才有权使用，而且需要对管理员进行不同的权限分配。

### 3.2.4 法律可行性分析

法律可行性主要是指待开发的软件是否存在知识产权等与法律有关的问题，如有这类问题，该软件即使开发成功，也难以成为产品销售。开发该系统的所有软件都使用正版，且属于自主设计，无剽窃他人设计行为。

因此，开发本系统在法律上是可行的。

# 4 系统总体设计

## 4.1 系统功能模块的设计分析

数据采集模块以广度搜索策略为设计思想，分别构造出网易新闻爬虫和留言爬虫。网易新闻爬虫的主要任务是是获取目标网易新闻中的所有发言贴的url、发帖人ID、发帖标题以及帖子留言数。留言贴爬虫的任务则是获取各发言贴内部的发言内容和留言内容、留言图片、留言人ID、留言时间以及留言设备型号进行采集。

数据清洗模块则主要通过正则匹配比对来实现数据缺值查找以及剔除错误值。并且在遍历数据的时候，及时调整数据编码方式，以防止可视化模块在数据展是出现编码错误。

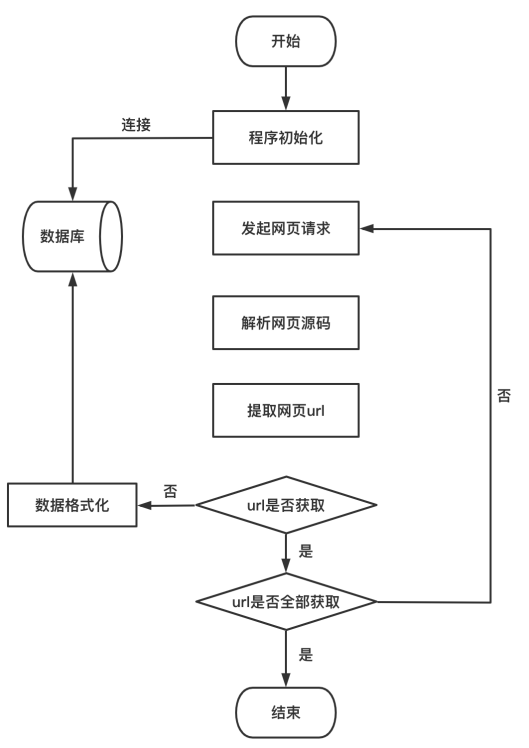
数据分析模块通过提取已收集的文本数据，从关键词、时间以及发帖频率等多 个维度进行数据分析，并对分析数据进行格式化处理。

数据可视化模块通过Echart来实现数据的图表化，以便更直观的了解数据特征。

## 4.2 爬虫模块设计

### 4.2.1 网易新闻爬虫设计

网易新闻爬虫程序的主要功能是通过遍历来实现对目标网易新闻的全部发言贴的url进行获取。图4为网易新闻爬虫数据获取的流程图。



# 

**图4 网易新闻爬虫数据获取流程**

首先初始化爬虫程序，在链接信息数据库的同时，向目标网易新闻服务器发送模拟 浏览器的网页请求，当接收到服务器返回的网页源代码时，便完成了数据请求功能。当网页源码数据接收在本地时，首先通过BS4将网页源码转化为格式化的剖析树, 并对网页信息进行初步大略的定位并将其提取出来。将提取的网络信息由html格式转化为单纯的字符串，之后通过re正则表达式实现目标信息的准确定位留言贴url。

当获取留言贴url之后，通过数据库中以存储的数据判断之前是否以及获取过 该url若当前url已经存在与数据库中，则舍弃当前url,重新进行网页请求；反之，则将其保存至数据库中。当出现连续的url都已获取，则判断是否全部获取所有url当全部url已经获取时，选择终止程序。

## 4.3 留言贴爬虫设计

首先初始化留言贴爬虫程序，同时链接url数据库提取数据库中储存的url向目 标网易新闻服务器发送包含发言贴url的网页请求，接收服务器返回的网页源代码，从而完成了数据请求功能。当网页源码数据接收在本地时，同样的首先使用BS4将网页源码转化为格式化的剖析树，并对网页信息进行初步大略的定位并将其提取出来。将提取的网络信息由html格式转化为单纯的字符串，之后通过re正则表达式实现目标信息的准确定位留言内容、留言时间以及留言图片。将获取的内容、时间以及图片转化为Json格式的数据，进而存储到分析内容数据库。

当获取一个页面的数据信息之后，通过判断是否获取全部页面信息来确定是否 已经对该留言贴的全部内容进行了采集。如果采集完毕，则从url数据库中提取新的留言贴url进行数据采集。

## 4.4 数据库设计

通过对系统的需求可知道数据库所要存储的数据主要为网易新闻数据以及留言贴 数据。通过分析，所采集的数据详细信息主要由两部分组成：

1）网易新闻数据：主要包括留言贴标题、回复数、留言贴href、留言贴发帖时间 及发帖人等信息。

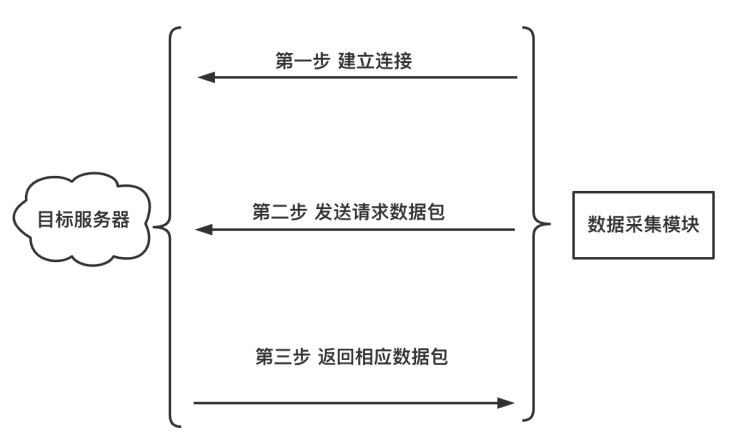
2）留言贴数据：主要包括留言者名称、ID、使用设备、留言时间、留言文本 内容以及留言图片。

因为社交网络的数据更新频率快、数据量大以及数据采集过程中所可能产生 的缺失值、误差值等原因，数据储存选用非关系型数据库MongoDB来进行存储。 根据对舆情监测系统的数据存储分析，将系统数据分别设计成两个相互独立的数据存储表，存储网易新闻数据以及留言贴数据。

# 5 系统设计与实现

## 5.1 数据采集模块实现

对于数据采集模块，能否和目标服务器进行网络连接交互，从而实现对网络数据的请求与接收十分重要。对此，数据采集模块与网络目标服务器之间的交互是通 过HTTP网络协议来实现其功能的。网络采集模块与网络Web服务器的数据交互 过程大致分为以下儿个步骤：第一步，网络采集模块通过伪装成浏览器建立与目标 服务器的网络交互连接；第二步，网络采集模块通过对目标网页的href进行特定的编码,同时向目标服务器发送该编码请求数据包；第三步，当目标服务器对数据采集模块发送的请求数据包进行解析，当数据包验证解析正常时，向数据采集模块发送相应数据包。



**图5 数据交互流程**

Python编程语言为在开发环境中为开发者提供了两种常用的可以发送HTTP 请求的库，这两种HTTP库分别时Python自带的Urllib和由kennethreitz提供的 Requests。Urllib是Python自带的标准HTTP库,在使用的时候可以无需安装，直接调用。Urllib提供了网页请求、服务器相应、代理以及URL解析功能。Request则是一种相较与Urllib库更为强大的数据库，提供了比Urllib更为多样化的功能，同时也进一步简化了调用并且其运行速度有了很大的提升。

虽然Python自带的Urllib已经能够满足进行网络HTTP数据传输交互的需求, 但是应为其调用方式较为复杂,无法适应当前程序开发的敏捷性。因此，选用 Requests作为网络数据通信库来执行HTTP网络数据传输。

在Python中调用Requests十分方便，只需在项目程序文件中引起Requests，便可以调用该HTTP库。Requests库提供了非常简单方便快捷的HTTP Get和HTTP Post 请求方法，可以快速的调用。

图6为目标数据获取情况。



**图6 目标数据获取情况**

## 5.2 数据存储功能实现

因为通过数据采集模块所获取的数据具有极强的对应结构，因此采用 MongoDB作为数据库。MongoDB数据库所存储的数据类型为键值对(Key-Value), 所以需要对已经获取的数据进行格式转换，将其格式转化为键值对。格式转化伪代码如下：

data = {  
 'title' : title,  
 'hum' : int(nun^rep),  
 'href' : https://news.163.com/  
 'date' : date,  
 'poster' : poster,  
 'flag' : 0  
}

在将数据存入数据库之前需要连接数据库，并确定服务器对应端口。同时因为MongoDB数据库的特性，数据只能存储在数据库中对应的集合中。伪代码如下：

database=pymongo.MongoClient(host='localhost', port=27017)  
// 连接到MongoDB数据库，并打开对应端口  
workdata = database['网易新闻']  
// 连接或者创建一个数据库  
collection = workdataf['data']  
// 创建数据集合，数据将会存储在集合之中  
collection.insert(data)  
// 将数据插入数据库中

## 5.3 数据分析模块实现

### 5.3.1 词云实现

数据分析模块主要任务是实现对文本数据的热词统计、热度统计。热词统计是通过词云来实现的，制作词云大致需要一下几个步骤：

1）文本内容收集：将所有文本内容集中统计。

2）文本分词：将按照汉语词语规则将文本中的词语分隔出来，并对其出现次数进行统计。

3）生成词云：确定生成词云图片的规则，并根据词语及其个数来绘制词云。

对于分文分词选择调用Python的Jieba库来完成其功能。代码如下：

def fetch\_keywords(new\_title):  
 """新闻关键词抽取，保留表征能力强名词和动词"""  
 seg\_words, pos\_words, key\_words = extractor.extract\_sentence(new\_title, keyword\_ratios=0.8)  
 seg\_key\_words = []  
 for word, pos, is\_key in zip(seg\_words, pos\_words, key\_words):  
 if pos in {'n', 'nt', 'nd', 'nl', 'nh', 'ns', 'nn', 'ni', 'nz', 'v', 'vd', 'vl', 'vu', 'a'} and is\_key:  
 if word not in STOPWORDS:  
 seg\_key\_words.append(word)  
 return seg\_key\_words

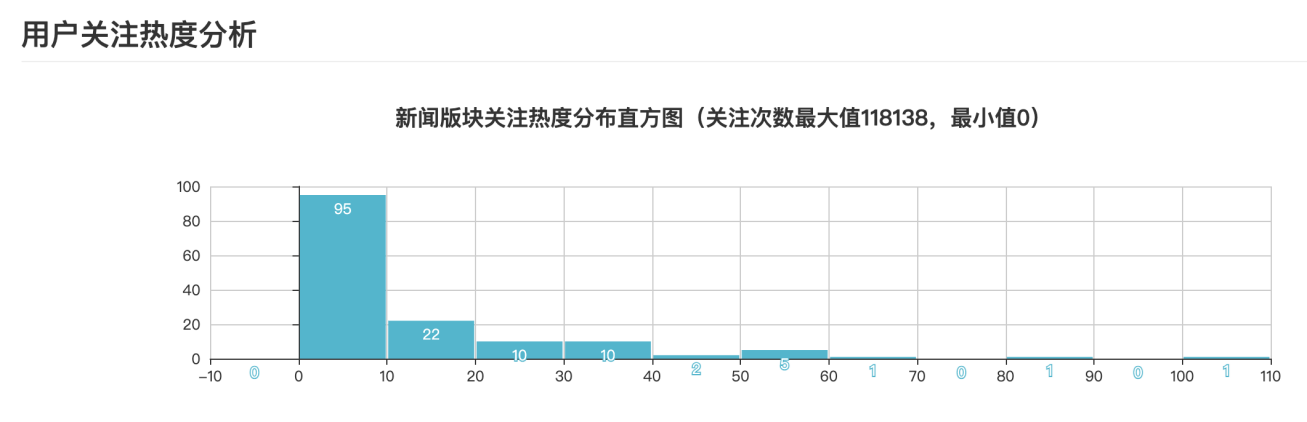


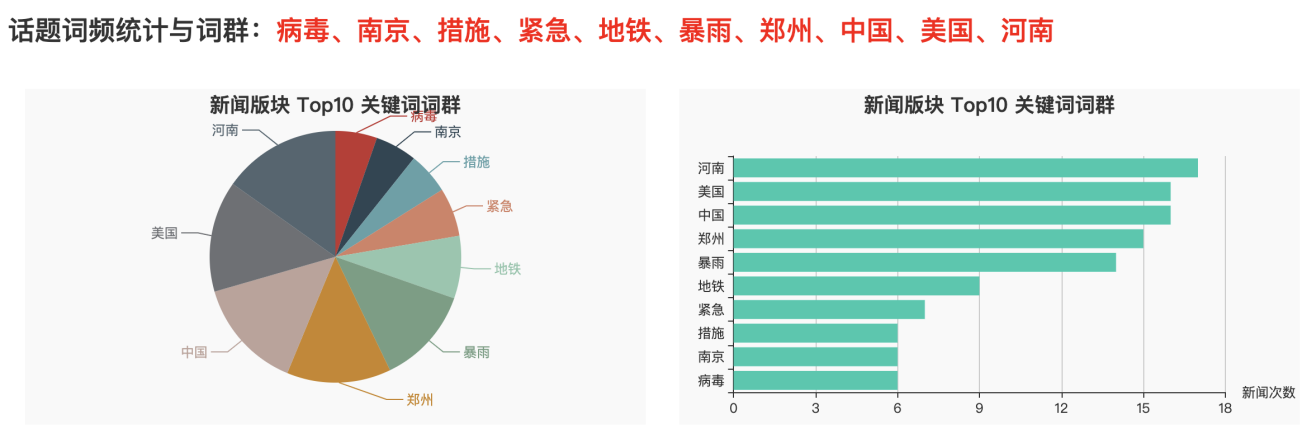
**图7 词云图片实例**

### 5.3.2 热度统计实现

通过对过往一段时期内的数据进行统计，来实现对舆情热度的准确统计。实现热度统计需要根据网易新闻的发布时间来进行计数。伪代码如下：

@app.route('/news\_words\_analysis/<category>')  
def news\_words\_analysis(category):  
 cate\_df = news\_df[news\_df['category'] == category]  
  
 word\_count = {}  
 for key\_words in cate\_df['title\_cut']:  
 for word in key\_words:  
 if word in word\_count:  
 word\_count[word] += 1  
 else:  
 word\_count[word] = 1  
  
 wordclout\_dict = sorted(word\_count.items(), key=lambda d: d[1], reverse=True)  
 wordclout\_dict = [{"name": k[0], "value": k[1]} for k in wordclout\_dict]  
  
 # 选取 top10 的词作为话题词群  
 top\_keywords = [w['name'] for w in wordclout\_dict[:10]][::-1]  
 top\_keyword\_counts = [w['value'] for w in wordclout\_dict[:10]][::-1]  
  
 click\_counts = cate\_df['click\_count'].values  
 max\_click\_count = max(click\_counts.tolist())  
 min\_click\_count = min(click\_counts.tolist())  
 click\_counts = (click\_counts - min(click\_counts)) / (max(click\_counts) - min(click\_counts)) \* 100  
  
 cate\_mean\_click\_count\_df = news\_df[['category', 'click\_count']].groupby(by='category').mean().reset\_index()  
 all\_category = cate\_mean\_click\_count\_df['category'].values.tolist()  
 mean\_click\_count = cate\_mean\_click\_count\_df['click\_count'].values.tolist()  
  
 return jsonify({'词云数据': wordclout\_dict, '词群': top\_keywords, '词群个数': top\_keyword\_counts,  
 'top10': cate\_df.values.tolist()[:10], 'click\_counts': click\_counts.tolist(),  
 'max\_click\_count': max\_click\_count, 'min\_click\_count': min\_click\_count,  
 'all\_category': all\_category, 'mean\_click\_count': mean\_click\_count  
 })



**图7 用户关注热度分析**

**图8 话题词频率统计与词群**

## 5.4 数据可视化模块实现

数据可视化模块按功能可以分为前端模块和后端模块两个部分。后端模块的主要功能是作为Web服务器响应前端请求，在本系统中选用Flask搭建后端Web 框架。前端模块则是通过Echart来实现数据的图表化，同时前端网页选用Jinja来搭建前端网页。

### 5.4.1 后端部分实现

通过Flask搭建一个应用，大致需要一下几个步骤：

1）创建实例应用。当Web 服务器收到来自浏览器请求时，便会调用请求所对应的实例应用；当应用调用完成时便会向请求方返回相应的响应数据；

2）创建路由和视图函数，路由对应的url, 当url满足对应路由时则会调用相关的视图函数，同时视图函数会调用实例应用；

3）调用前端模板。当实例应用运行完成之后，会得到相应的数据，Flask通过调用渲染前端模板，从而得到网页源代码，便会发送相应数据。

app = Flask(\_name\_)  
// 创建实例应用  
app.config.from\_object('config')  
// 配置实例应用  
@app.route('/')  
// 创建路由  
def hello\_world():  
 return render\_template('index. html')  
// 返回视图函数

### 5.4.2 前端模块

通过在浏览器端调用Echart来进行图表可视化。伪代码如下：

function Clip(options) {  
 this.\_target = options.target;  
 // 生命周期  
 this.\_life = options.life || 1000;  
 // 延时  
 this.\_delay = options.delay || 0;  
 // 开始时间  
 // this.\_startTime = new Date().getTime() + this.\_delay;// 单位毫秒  
 this.\_initialized = false;  
 // 是否循环  
 this.loop = options.loop == null ? false : options.loop;  
 this.gap = options.gap || 0;  
 this.easing = options.easing || 'Linear';  
 this.onframe = options.onframe;  
 this.ondestroy = options.ondestroy;  
 this.onrestart = options.onrestart;  
 this.\_pausedTime = 0;  
 this.\_paused = false;  
}

# 

# 6 结论

这次的毕业设计，我受益匪浅，不仅学到了很多新的知识，而且通过毕业论文的撰写，我的思维方式更加缜密，能多个角度的看待、处理问题，科学地整理和分析所学的专业知识，并将之提炼成一种思想。通过此次论文的撰写，我对此系统的开发和完善有了更科学系统的详细总结，我相信这对我日后的学习和工作将带来更有利的影响。由于时间和知识水平等客观因素，课题所完成的只是我所理解的程度，还有诸多方面仍可进行扩展及完善。 本文基于Python编程语言经过设计和实现了一个舆情监测系统，可以提供对 当前舆论情况的动态监测。该舆情系统从结构上可以分为三个模块：数据采集模块、数据分析模块以及数据可视化模块。通过数据采集模块可以实现网络数据的动态增量采集，同时将获取的数据存储到MongoDB数据库中，实现数据的备份并为之后的数据分析工作提供了便利。数据采集模块则实现了对所采集的数据进行数据清洗和数据分析等功能，为之后的数据可视化模块提供了可用数据。数据可视化模块则通过Echart技术实现数据图表化，同时通过Flask搭建Web网络框架 来实现其网站功能。

该系统在功能以及健壮性上能够满足对舆情监测需求的公司社团组织的需求，具有一定的工程价值。在技术上，本系统采用Python为主要编程语言，来实现绝大数的功能。本文的一些具体工作内容如下：

1）研究了Python相关知识，通过Python编程语言实现了舆情监测系统的实际与开发。

2）研究了HTTP网络协议，在数据采集模块通过模仿HTTP协议来实现与网络服务器的数据传输。

3）研究了非关系型数据库MongoDB，通过MongoDB数据库实现对数据的存储、调用以及数据备份。

4）研究了Flask相关技术，通过Flask搭建了Web服务器，来实现网站的前后端的数据交互方法及原理。

5）研究了Echart相关技术，通过借助JavaScript的Echart框架，实现了在前端页面完成数据图表化的功能。

# 参考文献

[1]赵俊. 社交网络的数据采集与分析方法研究[D]. 郑州大学.

[2]王青, 成颖, 巢乃鹏. 网络舆情监测及预警指标体系构建研究[J]. 图书情报工作, 2019, 55(8):5.

[3]高承实, 荣星, 陈越. 微博舆情监测指标体系研究[J]. 情报杂志, 2018, 30(9):5.

[4]孙飞显, 程世辉, 倪天林,等. 基于新浪微博的负面网络舆情监测研究——针对政府的负面网络舆情研究系列之一[J]. 情报杂志, 2017, 34(4):5.

[5]王艺. 重大突发公共事件的微博舆情监测与引导初探[J]. 贵州民族学院学报:哲学社会科学版, 2011.

[6]何佳, 周长胜, 石显锋. 网络舆情监控系统的实现方法[J]. 郑州大学学报：理学版, 2016, 42(1):4.

[7]丁杰, 徐俊刚. IPSMS:一个网络舆情监控系统的设计与实现[J]. 计算机应用与软件, 2017, 27(4):3.

[8]曹树金, 周小又, 陈桂鸿. 网络舆情监控系统中的主题帖自动标引及情感倾向分析研究[J]. 图书情报知识, 2018(1):8.

[9]叶昭晖, 曾琼, 李强. 基于搜索引擎的网络舆情监控系统设计与实现[J]. 广西大学学报：自然科学版, 2019, 36(A01):7.

[10]孙培梁, 林枫, 昌志泷. 大数据时代的政府网络舆情监测系统研究[J]. 数字技术与应用, 2017(3):2.

[11]唐勇. 互联网舆情监控系统的设计与实现[D]. 北京邮电大学, 2019.

**Design and Implementation of Network Public Opinion Monitoring System**

Yang Xuan

(School of Computer and Information Engineering, Anyang Normal University, Anyang 455002, Henan, China)

**Abstract:** With the increasing development and promotion of various network social media and the development of cloud computing and big data technology, it has become the focus of the current academic and industrial circles to detect current public opinion trends and events in a timely manner by mining and obtaining social big data and conducting data analysis. One of the technologies of concern.

Data collection, data analysis and visual display are all key links in the public opinion monitoring system. This paper realizes public opinion monitoring by implementing data collection module, data analysis module and data visualization module, and realizes public opinion monitoring of social network.

**Key words:** data collection; non-relational database; regular expression; Flask; HTML

# 致 谢

转身回眸之间，我在安阳师范学院已经度过了四年的大学生涯，临近尾声之际，我要诚恳的感谢我的导师，从论文选题、材料收集、写作方法等方面我的老师都给予了我莫大的帮助。

感谢我的父母，在我准备毕业设计论文的撰写的过程中，给予了我很大的信心。他们在我心累时，与我交谈，让我能够走出颓废。在此，我想对他们说一声，谢谢，我爱你们！

感谢我的室友，在毕业设计过程中，给予了我很大帮助，有许多问题在与他们讨论过程中得以解决。在此，由衷的表示对他们的感谢。

最后，感谢生活四年的母校，希望以后有机会能够回来看看。