安徽财经大学字

**本科毕业设计**

|  |  |
| --- | --- |
| **题 目** | 新闻文本自动分类算法研究及系统设计 |
| **学 院** | **管理科学与工程学院** |
| **专 业** | **计算机科学与技术** |
| **班 级** | **20计科4班** |
| **学 号** | **20200771** |
| **姓 名** | **陈渊琪** |
| **指导老师** | **张晓春** |

**2023 年 3 月**

安徽财经大学管理科学与工程学院

本科生毕业论文（设计）诚信承诺书

本人承诺：

1.所呈交的毕业论文（设计）《 毕业设计过程管理系统》，是在认真学习理解《安徽财经大学学位论文作假行为处理办法》和《管理科学与工程学院本科毕业论文（设计）工作管理办法》后，保质保量独立完成的，没有弄虚作假，没有抄袭别人的内容；

2.毕业论文（设计）所使用的相关资料、数据、观点等均真实可靠，文中所有引用的他人观点、材料、数据、图表均已注释说明来源；

3.毕业论文（设计）中无抄袭、剽窃或不正当引用他人学术观点、思想和学术成果，伪造、篡改数据的情况；

4.本人已被告知并清楚：学院对毕业论文（设计）中的抄袭、剽窃、弄虚作假等违反学术规范的行为将严肃处理，并可能导致毕业论文（设计）成绩不合格，无法正常毕业、取消学士学位资格或注销并追回已发放的毕业证书、学士学位证书等严重后果；

5.若在省教育厅、学校、学院组织的毕业论文（设计）检查中，被发现有抄袭、剽窃、弄虚作假等违反学术规范的行为，本人愿意接受学院按有关规定给予的处理，并承担相应责任。

学生（签名）：

年 月 日

指导老师（签名）：

年 月 日

**新闻文本自动分类算法研究及系统实现**

**摘 要**

当今社会，互联网应用在人类日常生活中无处不在，每天都有亿万条数据从互联网应用中产生，对这些数据进行人工分类是一项耗费成本巨大并且执行效率很低的任务。针对大数据中的海量文本信息，自然语言处理领域的研究实现了自动分类文本数据，高效分析文本信息。然而大多数研究都是基于英文文本数据，中文文本分类的技术研究仍然是我们关注的热点。

得益于人工智能与机器学习的快速发展，深度学习算法在文本分类领域广泛应用。文本分类方法从大类上可以分为单标签文本分类和多标签文本分类，目前单标签文本分类技术已相对成熟，然而新闻文本所包含的信息复杂多元，以单个标签对其进行归类往往是不全面的，一则新闻可提取出多个主题，每个主题对应一个标签，从而一则新闻可包含多个分类，即多标签文本分类。本文拟利用深度学习理论知识，结合新闻文本分类具体实例，对多种文本分类算法进行对比分析，尝试选取一种算法具体分析，并利用python语言基于该算法实现多标签新闻文本自动分类系统。

**关键词：**新闻 文本挖掘 文本分类 python

**Research and system implementation of news text automatic Classification Algorithm**

**Abstract**

In today's society, Internet applications are ubiquitous in People's Daily life. There are hundreds of millions of data generated from Internet applications every day. It is a costly and inefficient task to classify these data manually. In view of massive text information in big data, research in the field of natural language processing has realized automatic classification of text data and efficient analysis of text information. However, most researches are based on English text data, and the technical research of Chinese text classification is still the focus of our attention.

Thanks to the rapid development of artificial intelligence and machine learning, deep learning algorithm is widely used in the field of text classification. Text classification methods can be divided into single-label text classification and multi-label text classification. Currently, single-label text classification technology has been relatively mature. However, news texts contain complex and diversified information, and it is often incomplete to classify them with a single label. Thus, a news item can contain multiple categories, that is, multi-label text categories. This paper intends to make use of deep learning theoretical knowledge and combined with specific examples of news text classification to make a comparative analysis of various text classification algorithms, try to select an algorithm for specific analysis, and use python language to implement multi-label news text automatic classification system based on this algorithm.

**Key words:** news text mining text classification pytho

目录

[摘 要 3](#_Toc138688527)

[Abstract IV](#_Toc138688528)

[1 引言 6](#_Toc138688529)

[2 可行性研究 - 7 -](#_Toc138688530)

[2.1 需求概述 - 7 -](#_Toc138688531)

[2.2 系统可行性 - 7 -](#_Toc138688532)

[**2.2.1经济可行性** - 7 -](#_Toc138688533)

[**2.2.2技术可行性** - 9 -](#_Toc138688534)

[**2.2.3 操作可行性** - 9 -](#_Toc138688535)

[**2.2.4 法律可行性** - 9 -](#_Toc138688536)

[2.3 项目进度计划 - 10 -](#_Toc138688537)

[3 需求分析 - 12 -](#_Toc138688538)

[3.1 系统综合需求 - 12 -](#_Toc138688539)

[**3.1.1功能需求概述** - 12 -](#_Toc138688540)

[**3.1.2 系统性能需求概述** - 12 -](#_Toc138688541)

[**3.1.3 其他需求概述** - 12 -](#_Toc138688542)

[3.2 数据流图分析 - 13 -](#_Toc138688543)

[**3.2.1 顶层数据流图** - 13 -](#_Toc138688544)

[**3.2.2 功能级数据流图** - 13 -](#_Toc138688545)

[**3.2.3 数据流图细化与分解** - 14 -](#_Toc138688546)

[3.3 数据字典 - 16 -](#_Toc138688547)

[**3.3.1数据流条目定义** - 16 -](#_Toc138688548)

[**3.3.2数据存储条目定义** - 17 -](#_Toc138688549)

[**3.3.3数据处理储条目定义** - 18 -](#_Toc138688550)

[**3.3.4数据项条目定义** - 18 -](#_Toc138688551)

[3.4实体联系分析 - 19 -](#_Toc138688552)

[**3.4.1 实体提取及实体图** - 19 -](#_Toc138688553)

[**3.4.2 实体联系图** - 20 -](#_Toc138688554)

[4系统设计 - 21 -](#_Toc138688555)

[4.1总体设计 - 21 -](#_Toc138688556)

[**4.1.1 系统功能结构** - 21 -](#_Toc138688557)

[**4.1.2 系统层次图** - 23 -](#_Toc138688558)

[4.2 详细设计 - 25 -](#_Toc138688559)

[**4.2.1 代表性模块设计** - 25 -](#_Toc138688560)

[**4.2.2 系统数据库设计** - 27 -](#_Toc138688561)

[5 编码与测试 - 29 -](#_Toc138688562)

[5.1 编码 - 29 -](#_Toc138688563)

[**5.1.1 编码规则简介** - 29 -](#_Toc138688564)

[**5.1.2代表性模块示例** - 30 -](#_Toc138688565)

[5.2测试 - 36 -](#_Toc138688566)

[**5.2.1 白盒测试** - 37 -](#_Toc138688567)

[6 系统使用说明 - 42 -](#_Toc138688568)

[6.1 系统运行环境和配置 - 42 -](#_Toc138688569)

[6.2 系统操作说明 - 42 -](#_Toc138688570)

[**6.2.1 登录模块说明** - 42 -](#_Toc138688571)

[**6.2.2 新闻爬取模块说明** - 43 -](#_Toc138688572)

[**6.2.3 数据预处理模块说明** - 45 -](#_Toc138688573)

[**6.2.4 新闻分类模块说明** - 47 -](#_Toc138688574)

[**6.2.5 管理员个人中心模块说明** - 49 -](#_Toc138688575)

[**6.2.6 用户浏览模块说明** - 51 -](#_Toc138688576)

[7 总结 - 53 -](#_Toc138688577)

[参考文献 - 54 -](#_Toc138688578)

# 1 引言

在如今的网络信息时代，网络数据渗透在我们日常生活、工作和学习的方方面面，人们无时无刻地利用电子设备和网络生产数据，传播数据，其中文本数据占据大量比重。在互联网信息流动的过程中，文本数据具有在内容上复杂多样，在规模上数量庞大的特性，其中最具有代表意义的网络新闻是人们获取社会信息，表述观点诉求的有效途径。然而如何从海量的网络新闻中精准地筛选出满足用户需求的新闻信息，仍然是值得关注的热点问题。因此，在当今移动互联网日益成熟的大环境下，针对用户多样化的需求，对海量新闻文本进行有效的分类和管理具有非常重要的研究意义。

得益于人工智能与机器学习的快速发展，深度学习算法在文本分类领域广泛应用。文本分类方法从大类上可以分为单标签文本分类和多标签文本分类，目前单标签文本分类技术已相对成熟，然而新闻文本所包含的信息复杂多元，以单个标签对其进行归类往往是不全面的，一则新闻可提取出多个主题，每个主题对应一个标签，从而一则新闻可包含多个分类，即多标签文本分类。本文拟利用深度学习理论知识，结合新闻文本分类具体实例，对多种文本分类算法进行对比分析，尝试选取一种算法具体分析，并利用python语言基于该算法实现多标签新闻文本自动分类系统。

# 2 可行性研究

该章节简要阐述了新闻文本分类的应用场景与开发价值，从经济可行性、技术可行性、操作可行性和法律可行性三个角度对新闻文本自动分类系统的开发可行性进行分析，最后对系统开发周期进行预估和计划。

## 2.1 需求概述

当今社会，互联网应用在人类日常生活中无处不在，每天都有亿万条数据从互联网应用中产生，对这些数据进行人工分类是一项耗费成本巨大并且执行效率很低的任务。针对大数据中的海量文本信息，自然语言处理领域的研究实现了自动分类文本数据，高效分析文本信息。然而大多数研究都是基于英文文本数据，中文文本分类的技术研究仍然是我们关注的热点。

大数据时代，人们每天都在各式各样的互联网设备中生产数据、分析数据、利用数据，在这亿万条数据中，文本数据占据了大量的比重。在互联网信息流动的过程中，文本数据具有在内容上复杂多样，在规模上数量庞大的特性，其中最具有代表意义的网络新闻是人们获取社会信息，表述观点诉求的有效途径。截至 2021 年 6 月，我国网络新闻用户规模达 7.60亿，占网民整体的 75.2%，微博、微信、抖音等互联网热点应用无时无刻的向庞大的用户群体推送着各式各样的网络新闻，然而如何从海量的网络新闻中精准地筛选出满足用户需求的新闻信息，仍然是值得关注的热点问题。因此，在当今移动互联网日益成熟的大环境下，针对用户多样化的需求，对海量新闻文本进行有效的分类和管理具有非常重要的研究意义。

## 2.2 系统可行性

在进行软件项目开发之前，需对项目的开发价值进行评测分析，以下分别从经济可行性、技术可行性、操作可行性和法律可行性四个角度进行阐述。

### **2.2.1经济可行性**

**（1）成本估算**

表 1成本估计

|  |  |
| --- | --- |
| 项目名称 | 成本估计(单位/元) |
| 软件开发 | 5000 |
| 服务器租赁 | 500 |
| 维护工资 | 3000 |
| 人力开销 | 2000 |
| 合计 | 10050 |

**（2）效益**

表 2效益

|  |  |
| --- | --- |
| 项目 | 收益（元/年） |
| 一次性收益 | 无 |
| 经常性收益（下载文献） | 80000 |
| 不可定量收益 | 无 |
| 企业定制服务 | 20000 |
| 总计 | 100000 |

**（4）收益/投资比**

一次性支出：26000元

经常性支出：24000元/年

收益：100000元/年

收益/投资比：100000\*5/(26000+24000\*5)=3.424657

**（5）货币的时间价值**

五年预计收益：100000\*5-（26000+24000\*5）=354000（元）

文献管理系统的前期软件开发阶段，需要耗费巨大的人力，投资的成本和未来获得的效益会随着软件每年的货币时间价值有所改变。文献管理系统五年收益为354000元。实际收益需要通过货币时间价值来进行核算。用利率的形式表示货币的时间价值。假设年利率为i，如果现在存入P元，则n年后可以得到的钱数为：

H:\temp\ksohtml14132\wps1.jpg (1)

这也就是P元钱在n年后的价值。反之，如果n年后能收入F元钱，那么这些钱的现在价值是：

H:\temp\ksohtml14132\wps2.jpg (2)

假定年利率为12%，利用上面计算货币现在价值的公式可以算出系统5年预计收益的现在价值，如下表2-5所示。

表 3未来收入折算

|  |  |  |
| --- | --- | --- |
| 将来值（元） | （1+i)^n | 现在值（元） |
| 354000 | 1.7623 | 200873 |

**（6）投资回收期估算**

文献管理系统基础建设成本约为26000元，第一年内收入为100000元，软件投资的回收期约为

 （3)

因此软件的投资回收期约为0.26年，本系统开发成本较低，维修成本适中，软件投资回报期较短可以较快获得利润，用户群体丰富且需求量大，经济利益客观值得投资。

### **2.2.2技术可行性**

新闻文本自动分类系统的实现需要采用机器学习算法，主要包括文本特征提取、特征向量化、分类器训练等技术。在Python语言中，有众多的机器学习库和自然语言处理库可供选择，如scikit-learn、Tensorflow、NLTK等。这些库提供了强大的功能和易于使用的API，使得机器学习算法的实现变得更加简单和高效。此外，Python还具有丰富的第三方包和库，可用于数据处理、可视化和模型部署等方面。因此，从技术上来说，新闻文本自动分类系统的实现具有可行性。

### **2.2.3 操作可行性**

新闻文本自动分类系统的操作主要包括文本数据的预处理、模型训练和模型应用等环节。其中，文本数据的预处理需要进行分词、去停用词、词形还原等处理；模型训练需要选择合适的算法和参数，进行训练和调优；模型应用则需要将待分类文本进行特征提取、特征向量化、分类器预测等处理。这些操作需要熟悉Python编程语言和机器学习算法的相关知识和技能。同时，对于大规模的文本数据集，还需要具备数据处理和存储的能力。因此，该系统的操作可行性需要基于相关专业知识和实践经验。

此外，新闻文本自动分类系统的操作还需要考虑用户体验和易用性。系统应该提供易于操作和友好的用户界面，帮助用户快速和准确地完成分类操作。同时，系统应该具备良好的稳定性和可扩展性，以适应不断变化的用户需求和技术环境。

### **2.2.4 法律可行性**

新闻文本自动分类系统需要收集和使用大量的新闻文本数据，因此需要考虑数据隐私保护和知识产权等法律问题。在数据隐私保护方面，需要遵守相关法律法规，如个人信息保护法等，保护用户的个人信息和隐私。在知识产权方面，需要确保数据采集和使用的合法性，避免侵犯他人的知识产权。同时，还需要考虑到文本数据的版权问题，以避免侵犯新闻媒体的版权。因此，在实现新闻文本自动分类系统时，需要仔细考虑相关法律法规，遵守合法规定。同时，也需要保障用户的知情权和选择权，明确告知用户数据采集和使用的目的和方式，避免侵犯用户隐私和权益。

除此之外，还需要考虑系统的合规性和安全性。合规性方面，需要遵循相关法律法规和政策规定，确保系统的合法性和合规性。安全性方面，需要采取必要的技术措施，保障系统和数据的安全性和可靠性，防范数据泄露和黑客攻击等安全风险。

因此，在实现新闻文本自动分类系统时，需要综合考虑技术、操作和法律等多个方面的因素，确保系统的可行性和合规性。同时，还需要不断更新和改进系统，以满足不断变化的用户需求和技术环境。

## 2.3 项目进度计划

图 1项目进度甘特图

表 4项目进度表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 项目进度名称 | 开始时间 | 所需时长（天） | 结束时间 |
| 1 | 问题定义 | 2023/2/27 | 5 | 2023/3/3 |
| 2 | 可行性分析 | 2023/3/4 | 7 | 2023/3/10 |
| 3 | 需求分析 | 2023/3/11 | 8 | 2023/3/19 |
| 4 | 总体设计 | 2023/3/20 | 16 | 2023/4/4 |
| 5 | 详细设计 | 2023/4/5 | 19 | 2023/4/23 |
| 6 | 编码和单元测试 | 2023/4/24 | 11 | 2023/5/4 |
| 7 | 综合测试 | 2023/5/5 | 21 | 2023/5/25 |
| 8 | 软件维护 | 2023/5/26 |  |  |

1.问题定义：确定选题、问题分析。分析新闻文本自动分类系统将要解决的问题；确定工作目标和工程规模。

2.可行性分析：从经济、技术、操作和法律四个层面分析新闻文本自动分类系统对于问题的解决方案，是否值得对该项目进行资源投入。

3.需求分析：从用户角度对该管理系统提出需求，分别从系统功能需求、性能需求、安全性需求、可靠性需求和系统兼容性需求几个角度分析。

4.总体设计：梳理新闻文本自动分类系统的体系结构，将程序模块化。

5.详细设计：对新闻文本自动分类系统的各个小模块进行详细设计，确定模块功能所需要的算法和数据结构。

6.编码和单元测试：将设计好的各个模块以python语言实现，并对各个模块的功能进行测试。

7.综合测试：将经过单元测试检验的各个小模块装配起来，根据系统功能进行综合测试。

8.软件维护：系统正式上线后，诊断后修复使用过程中出现的软件错误，并修改软件以适应环境的变化，满足用户要求

# 3 需求分析

第二章对新闻文本自动分类系统的开发可行性进行了多方面的分析，本章主要从用户需求角度对软件的功能和性能等方面进行概述，并设计系统的顶层数据流图和各个功能模块的数据流图，根据各个功能模块设计系统数据库。

## 3.1 系统综合需求

新闻文本自动分类系统分为两大部分，分别面向用户和管理员，管理员通过该系统从网络中爬取新闻数据，由系统核心算法将这些新闻按标签分类，并呈现给用户；用户可根据分类标签选择对应的新闻。

### **3.1.1功能需求概述**

新闻文本自动分类系统通过爬虫程序从网络爬取新闻数据，利用深度学习算法对新闻数据进行分类，最后呈现给用户。主要功能可以分为以下四个部分：

1.数据爬取：系统所使用的数据均来源于网络，使用爬虫程序高效快速的爬取新闻数据。

2.文本归类：利用所爬取的新闻数据，基于深度学习算法对其进行归类。

3.用户管理：保存并管理普通用户和管理员的账号注册信息、个人信息。

4.新闻数据存储：爬取的新闻数据以及分类之后的数据保存至对应数据库存储。

### **3.1.2 系统性能需求概述**

1.系统的稳定性：本系统要求在没有人为干预的情况下可以稳定的运行，并且系统自身可以处理一些异常和错误，要求系统平均每天出错的次数不超过 3 次。

2.响应时间：每条新闻的平均爬取时间不超过5秒，在10000条数据以内文本分类时间不超过10分钟。

### **3.1.3 其他需求概述**

1.系统的可靠性：首先本系统需要要保证能够向用户展示新闻文本正确的分类结果，其次系统还需要精准捕捉用户的异常操作，以及系统内部出现的异常和错误，及时向用户以及管理员反馈错误信息并且记录到管理日志中。

2.系统的易用性：要求本系统界面简洁明了，前端交互界面能够让用户一目了然，提高用互体验好感。

3.系统可恢复性：要求本系统在出现异常错误是能够通过人工操作，甚至不需要人工操作可以恢复到正常运行的状态，管理员对数据库进行操作出现错误时，必须保证数据回滚，保护数据安全。

4.系统的安全性：首先本系统必须保证用户的传输的隐私数据的保护，其次要保证系统数据库的安全性，避免数据泄露。

## 3.2 数据流图分析

以下主要从顶层数据流图和功能级数据流图两个角度对新闻文本自动分类系统进行分析。进而对功能数据流图进行细化分析。

### **3.2.1 顶层数据流图**

以下为新闻文本自动分类系统的顶层数据流图，由用户和管理员两个实体，网络新闻数据库和系统本体构成。管理员通过系统从网络中爬取新闻数据，进而通过系统对新闻文本进行分类和存储，用户通过登录系统检索和阅览新闻。

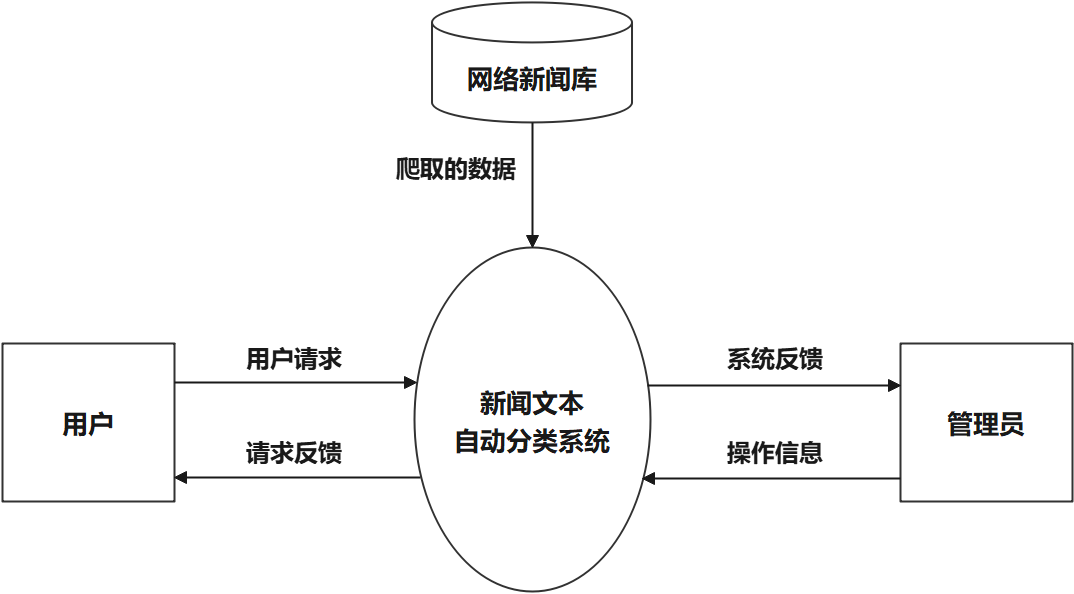


图 2顶层数据流图

### **3.2.2 功能级数据流图**

以下是系统的功能级数据流图，本系统可分为用户登录、新闻爬取、新闻分类、新闻推送与检索和用户信息管理五个主要模块（功能）。

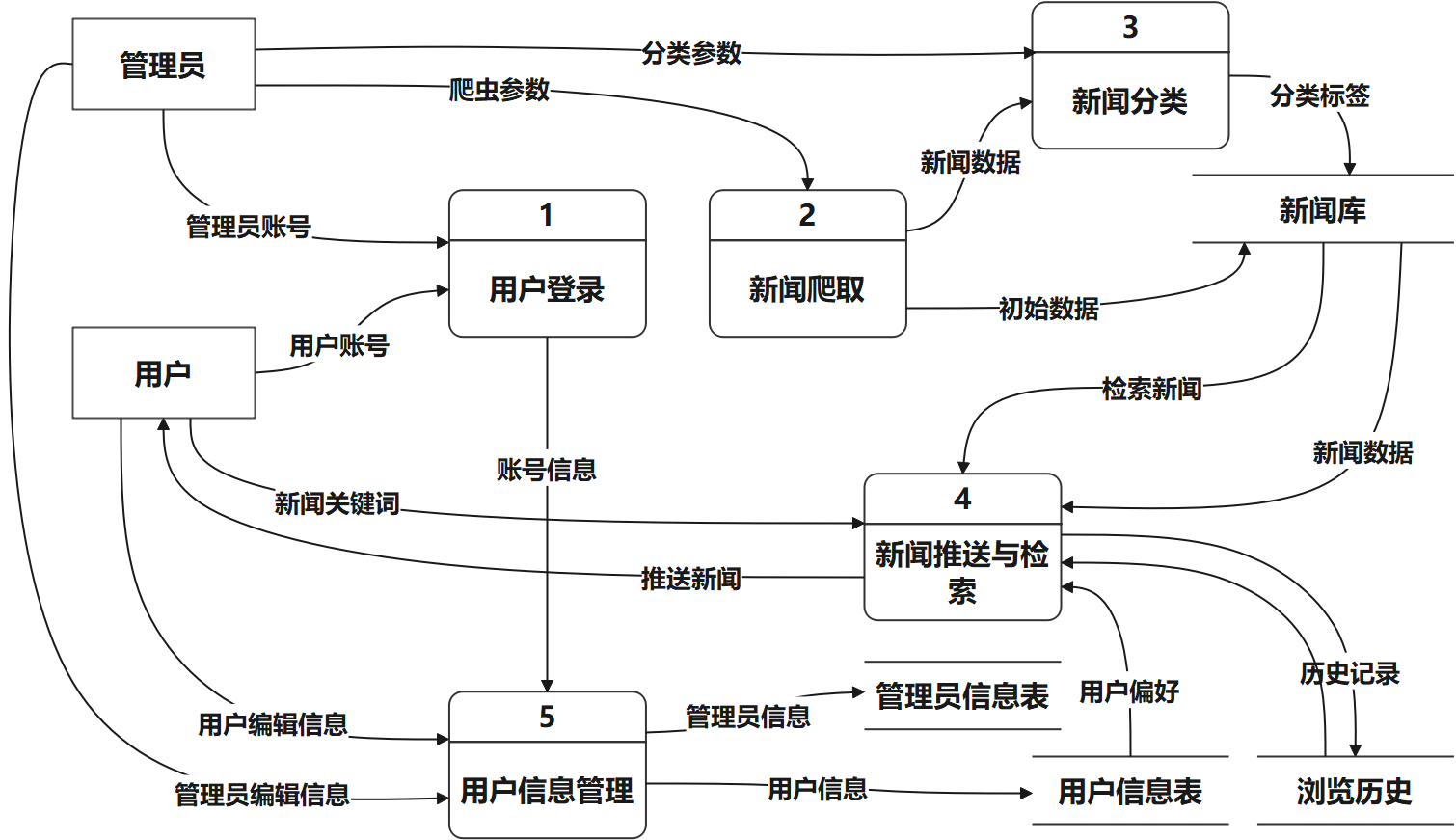


图 3功能级数据流图

### **3.2.3 数据流图细化与分解**

根据上述功能级数据流图，对各个功能进行更细化的如下分析。

**1.用户登录**

用户登录模块分为注册、登录验证和找回密码三个模块构成。

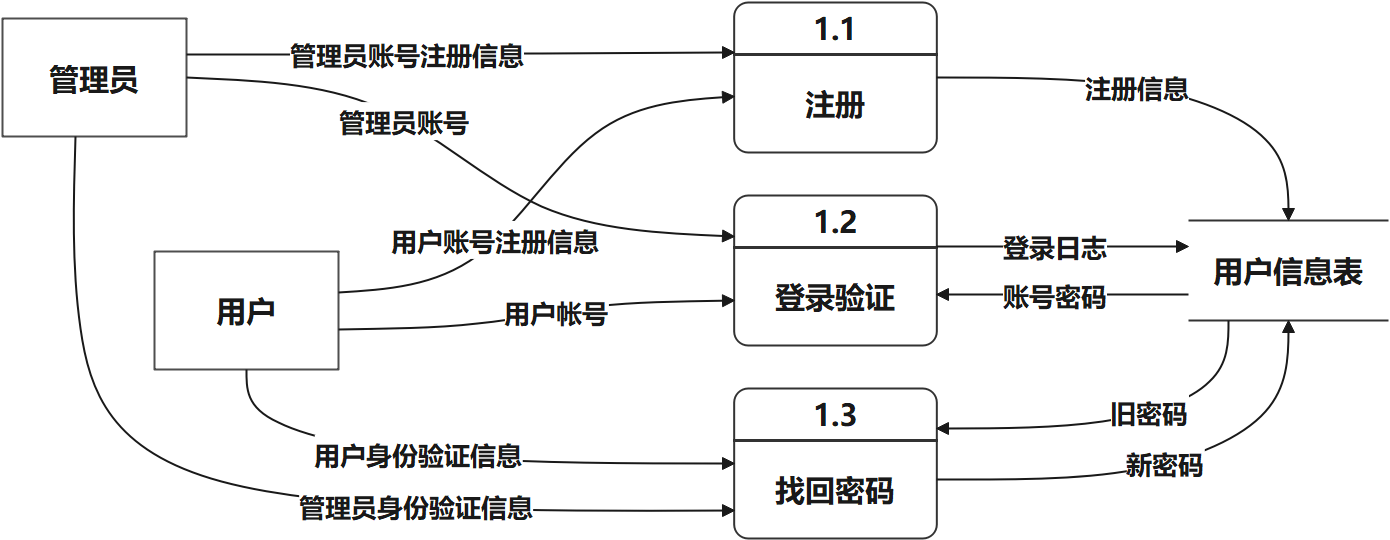
****

图 4用户登录数据流图

**2.新闻爬取**

新闻爬取主要依赖python爬虫，从网络中爬取新闻文本数据，并对爬取的文本进行清洗，去除脏数据，并对清洗后的文本数据进行初步标注。

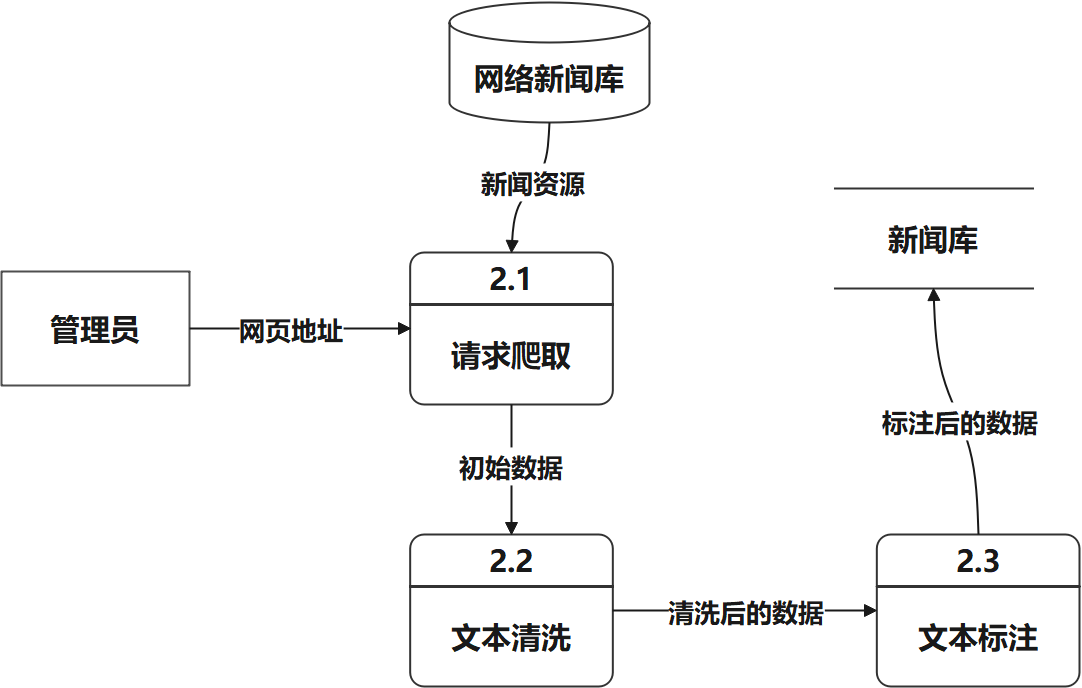
****

图 5新闻爬取数据流图

**3.新闻分类**

新闻分类是该系统的核心部分功能，全过程需要管理员的监督与操作，首先将初步整理的新闻文本数据进行预处理和特征表示，进而对数据进行均衡性分析，然后融合特征表示，构建分类网络，最后对分类后的数据进行检测，并修改错误分类。

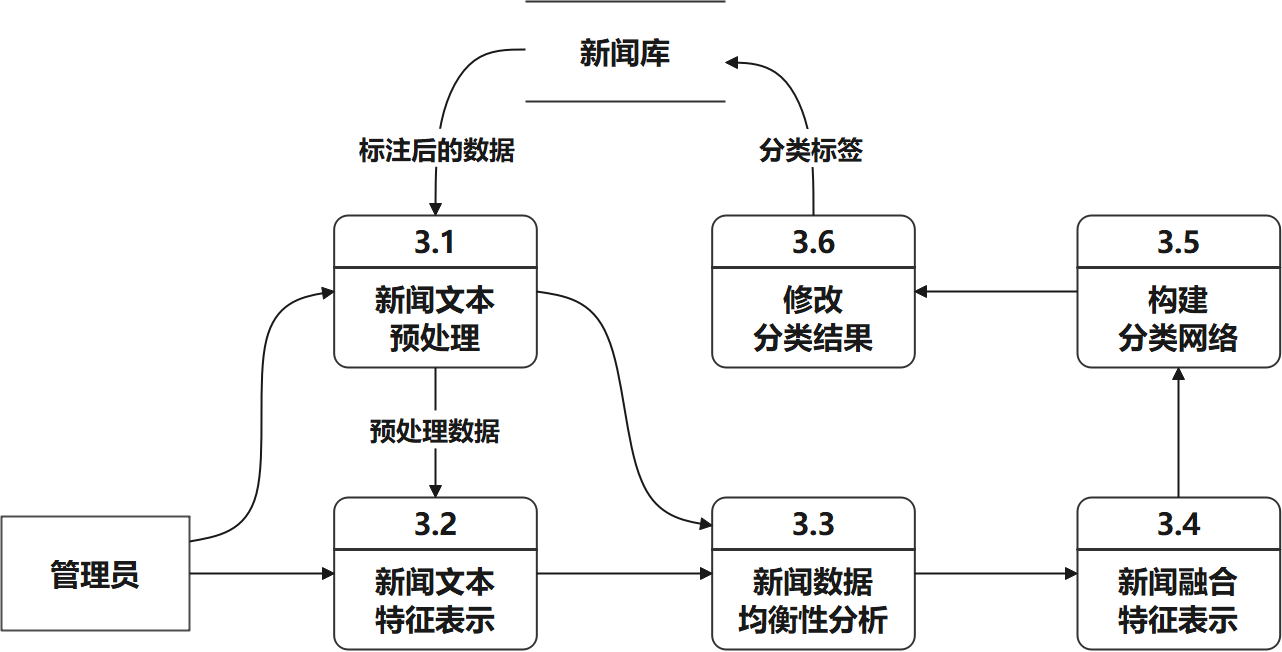
****

图 6新闻分类数据流图

**4.新闻推送与检索**

新闻推送与检索，即由新闻推送和检索两个功能组成，系统根据用户的浏览数据，获取新闻标签并计数，同时考虑用户个人信息中的“偏好”，为用户个性化推荐新闻，同时用户可根据新闻关键词或标签进行检索。

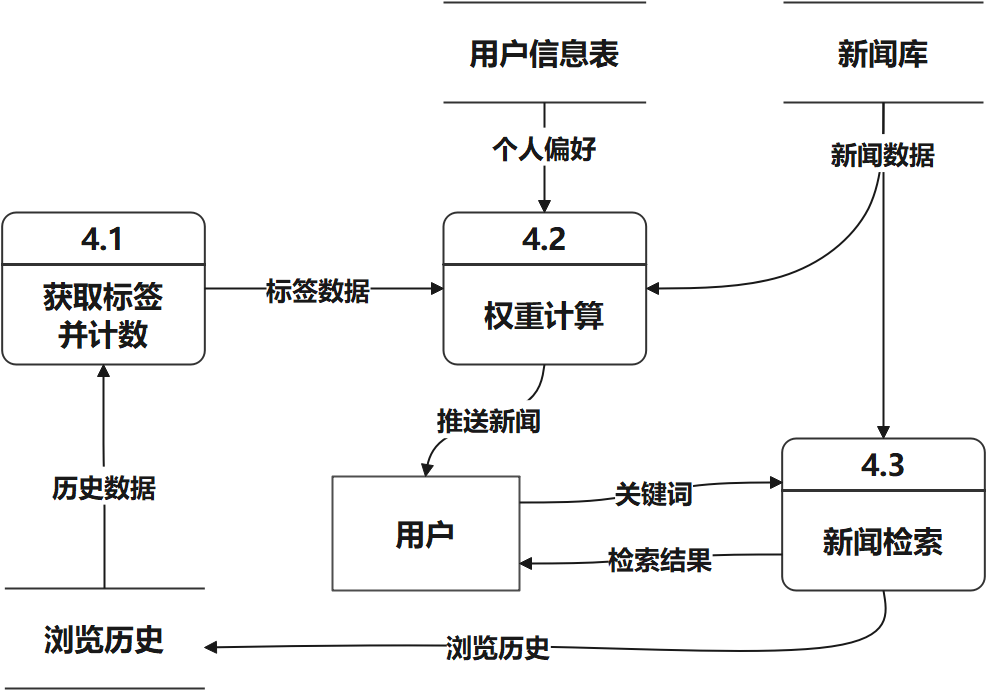
****

图 7新闻推送与检索数据流图

**5.用户信息管理**

用户细心管理，为管理员与用户提供信息编辑通道，其中，修改密码需要用户（管理员）提供旧密码。

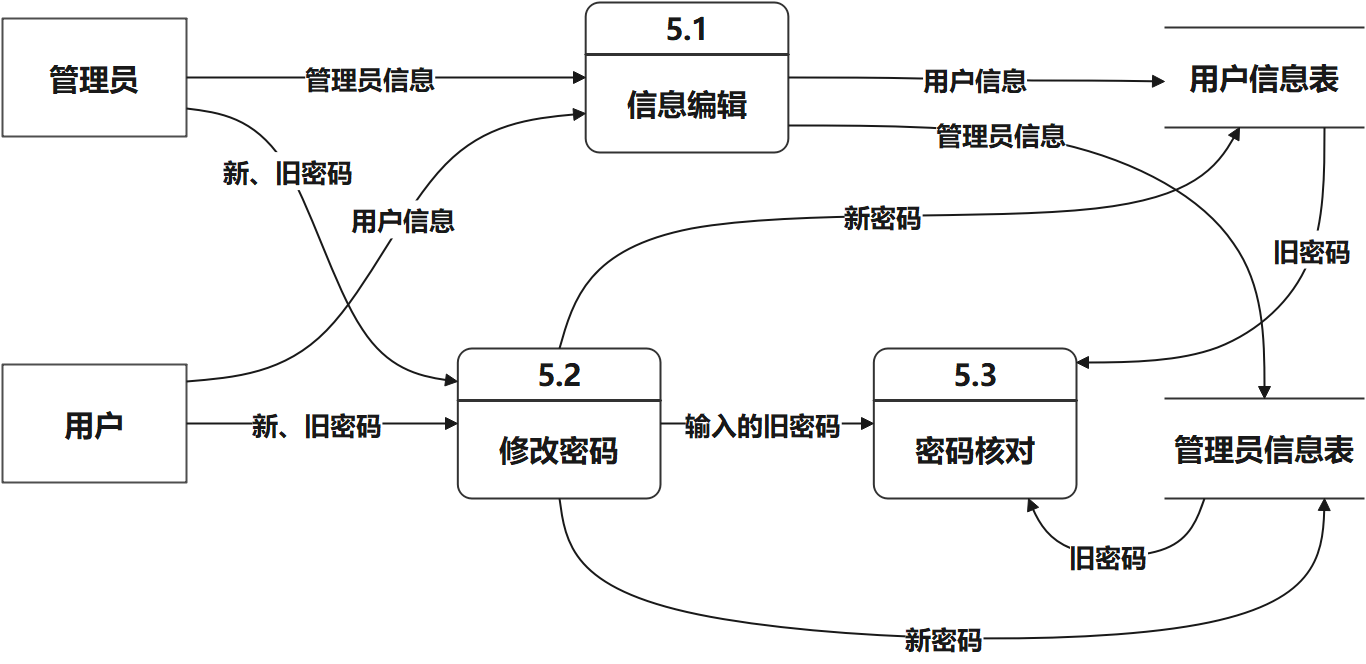
****

图 8用户信息管理数据流图

## 3.3 数据字典

### **3.3.1数据流条目定义**

表 5注册信息

|  |  |  |
| --- | --- | --- |
| **名称** | 注册信息 | |
| **别名** | 无 | |
| **描述** | 注册信息的输入 | |
| **数据流组成** | | 注册信息 = 手机号码 + 账号 + 密码 |
| **数据流来源** | | 用户、管理员 |
| **数据流去向** | | 用户信息表 |

表 6登录日志

|  |  |  |
| --- | --- | --- |
| **名称** | 登录日志 | |
| **别名** | 无 | |
| **描述** | 登录信息的输入 | |
| **数据流组成** | | 登录日志 = 账号 + 密码 + 登录时间 |
| **数据流来源** | | 用户、管理员 |
| **数据流去向** | | 用户信息表 |

### **3.3.2数据存储条目定义**

表 7用户信息表条目定义

|  |  |
| --- | --- |
| **名称** | 用户信息表 |
| **别名** | 用户信息 |
| **描述** | 记录用户账号、登录、个人偏好等的相关信息 |
| **定义** | 用户信息 = 账号 + 密码 + 登录时间 + 性别 + 年龄 + 偏好 + 浏览记录 |
| **位置** | 输入到用户信息 |

表 8新闻库条目定义

|  |  |
| --- | --- |
| **名称** | 新闻库 |
| **别名** | 新闻存储信息 |
| **描述** | 记录爬虫所爬取的所有新闻以及分类标签等相关信息 |
| **定义** | 新闻库 = 新闻 + 分类标签 + 检索次数 + 添加日期 |
| **位置** | 输入到新闻存储信息 |

### **3.3.3数据处理储条目定义**

表 9新闻分类处理

|  |  |  |  |
| --- | --- | --- | --- |
| **数据加工名称** | | | 新闻分类 |
| **说明** | 根据新闻主题为新闻数据添加分类标签 | | |
| **输入数据流** | | 爬虫爬取新闻初始数据 | |
| **输出数据流** | | 分类标签 | |
| **加工逻辑：** | | | |
| 1. 新闻文本分词  2. 新闻主题词分析  3. 根据主题词添加标签 | | | |

表 10个性推荐处理

|  |  |  |  |
| --- | --- | --- | --- |
| **数据加工名称** | | | 个性推荐 |
| **说明** | 根据用户“个人偏好”以及浏览历史为用户个性化推荐新闻 | | |
| **输入数据流** | | 用户个人信息中的“偏好”、浏览历史、新闻数据 | |
| **输出数据流** | | 推荐新闻 | |
| **加工逻辑：** | | | |
| 1. 获取个人偏好标签  2. 获取浏览历史新闻标签  3. 根据标签检索新闻数据 | | | |

### **3.3.4数据项条目定义**

表 11账号数据项

|  |  |
| --- | --- |
| **名称** | 账号 = 1位身份识别码 + 8位用户编码 |
| **别名** | 登录账号 |
| **含义** | 用户（或管理员）登录系统的唯一身份凭证 |
| **类型** | 字符串 |
| **长度** | 9 |

表 12密码数据项

|  |  |
| --- | --- |
| **名称** | 密码 |
| **别名** | 登录密码 |
| **含义** | 用户（或管理员）登录系统的必需凭证，由6-12位字母或数字组成 |
| **类型** | 字符串 |
| **长度** | 6-12 |

## 3.4实体联系分析

### **3.4.1 实体提取及实体图**

表 13实体提取表

|  |  |  |
| --- | --- | --- |
| **实体** | **主键** | **其他属性** |
| 用户 | 账号 | 密码、手机号、偏好、性别、年龄、登录时间 |
| 管理员 | 账号 | 密码、管理员等级、手机号、登录时间 |
| 新闻 | 新闻ID | 新闻标题、新闻正文、分类标签、下载日期、浏览次数、操作账号 |
| 浏览历史 | 浏览编号 | 新闻ID、浏览账号、分类标签、浏览时间 |

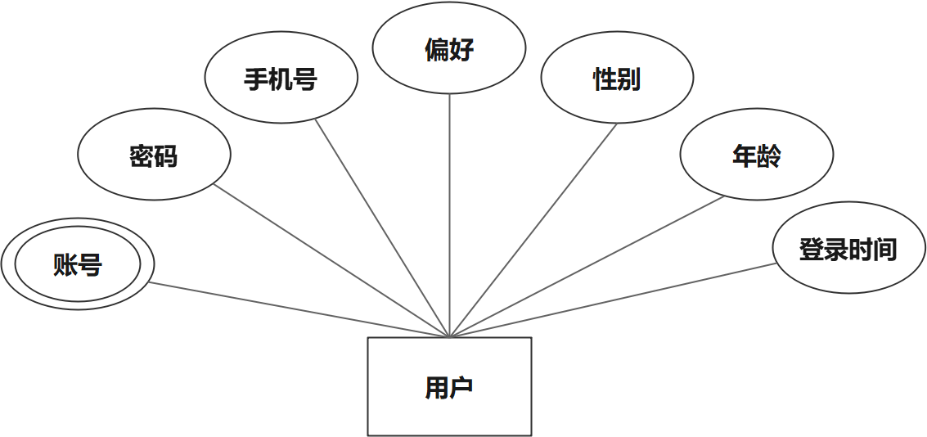


图 9用户实体图

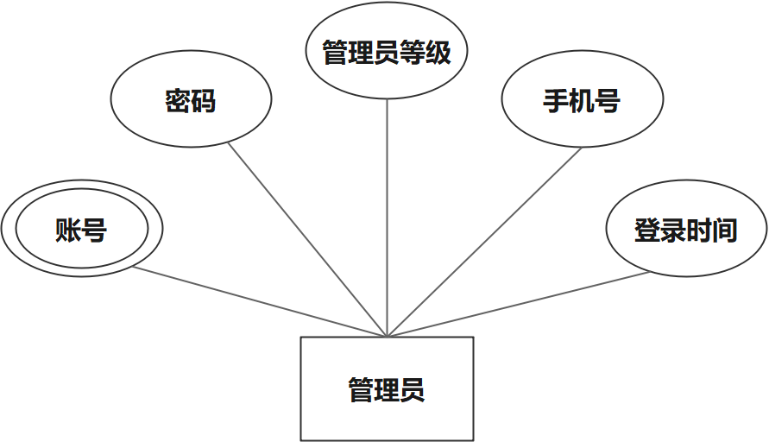


图 10管理员实体图

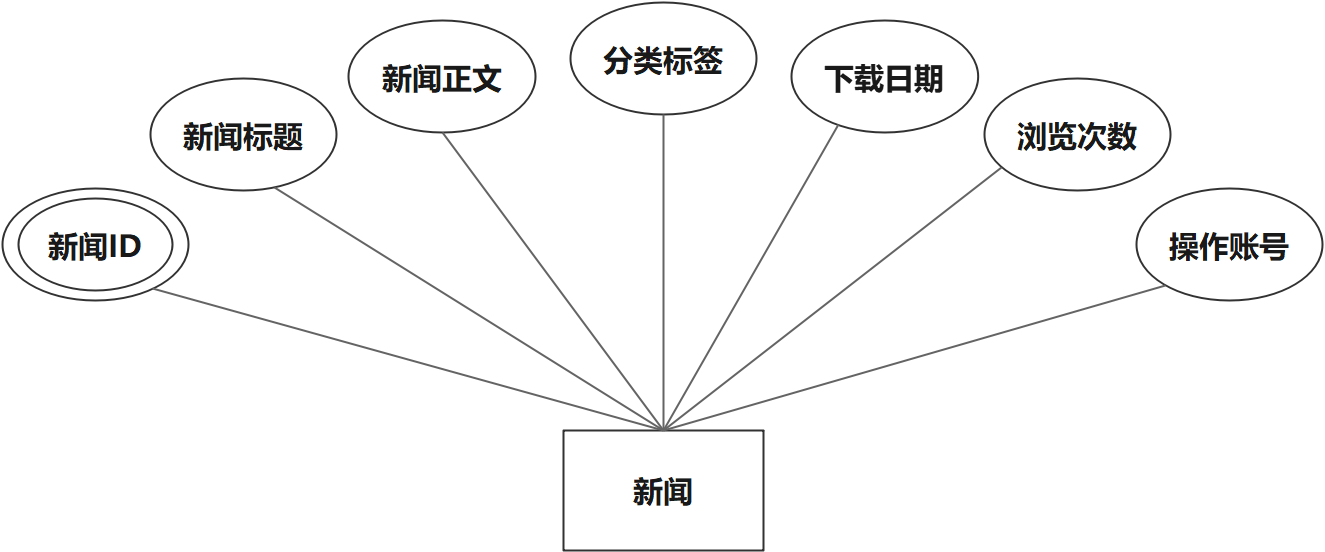


图 11新闻实体图

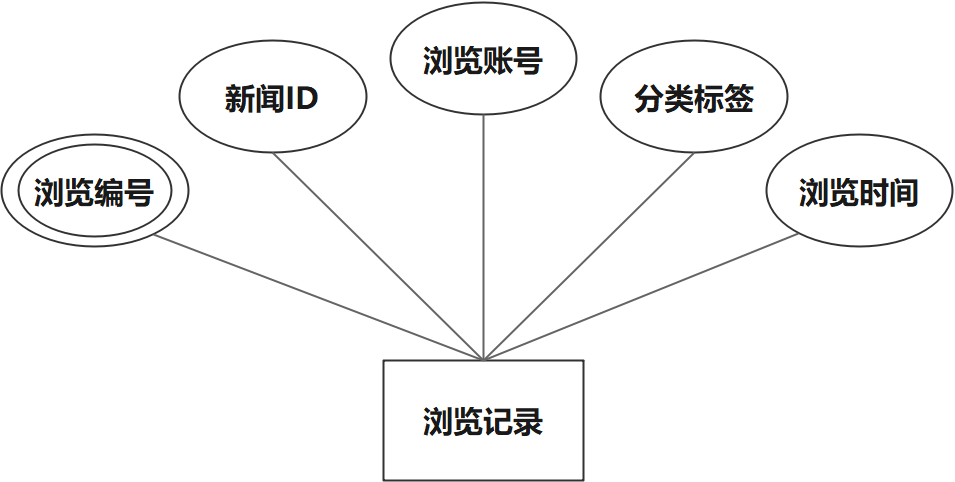


图 12浏览记录实体图

### **3.4.2 实体联系图**

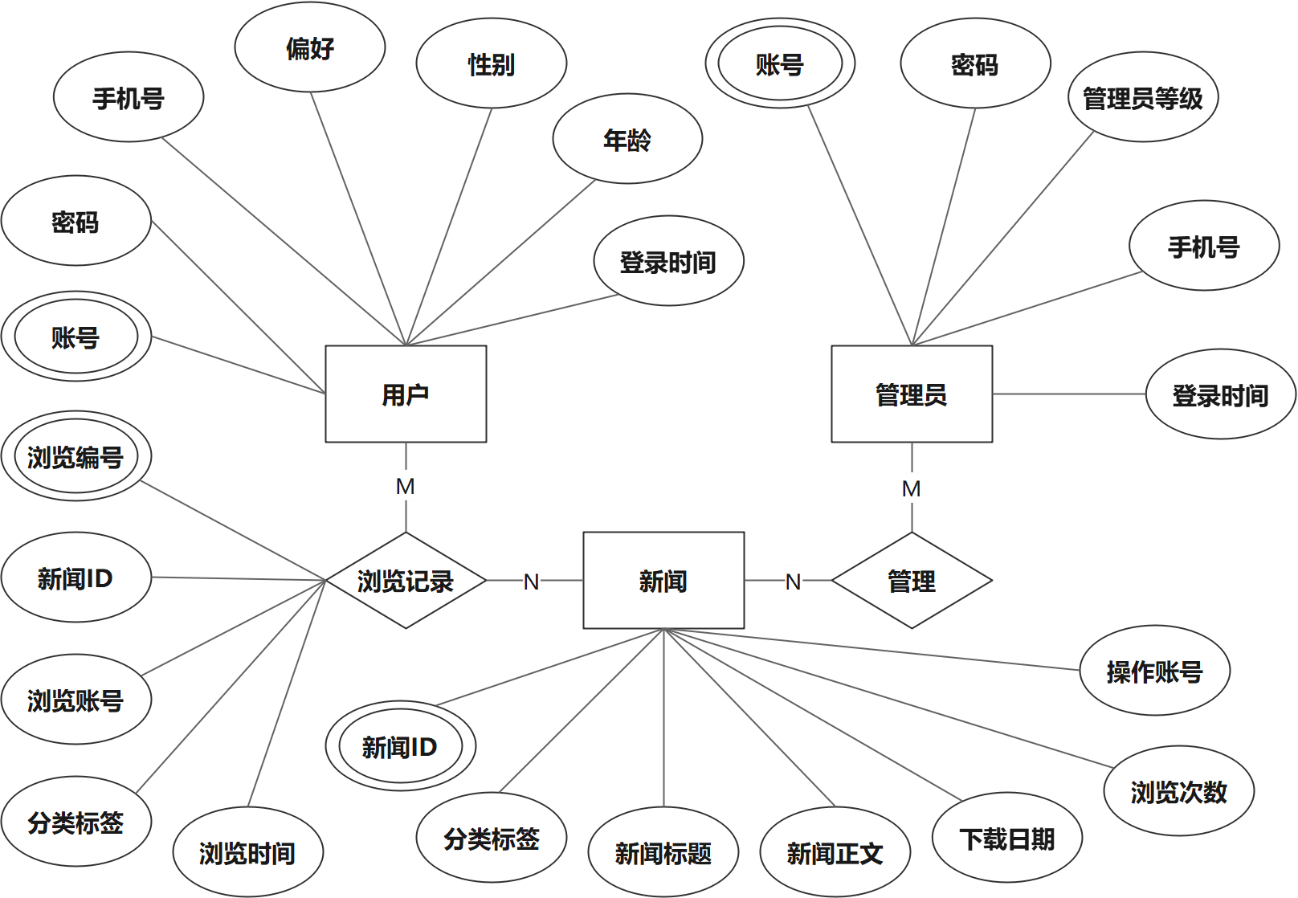


图 13实体联系图

# 4系统设计

该章节将对系统的整体架构和各个功能模块进行设计。先对系统进行总体架构的设计，设计系统功能结构图和系统层次图；进而对系统的各个功能模块进行详细设计，并对代表性模块进行着重介绍，最后对系统数据库进行设计。

## 4.1总体设计

总体设计包括系统功能结构的设计和系统层次图的设计。

### **4.1.1 系统功能结构**

**1.基于变换流的数据流图分析**

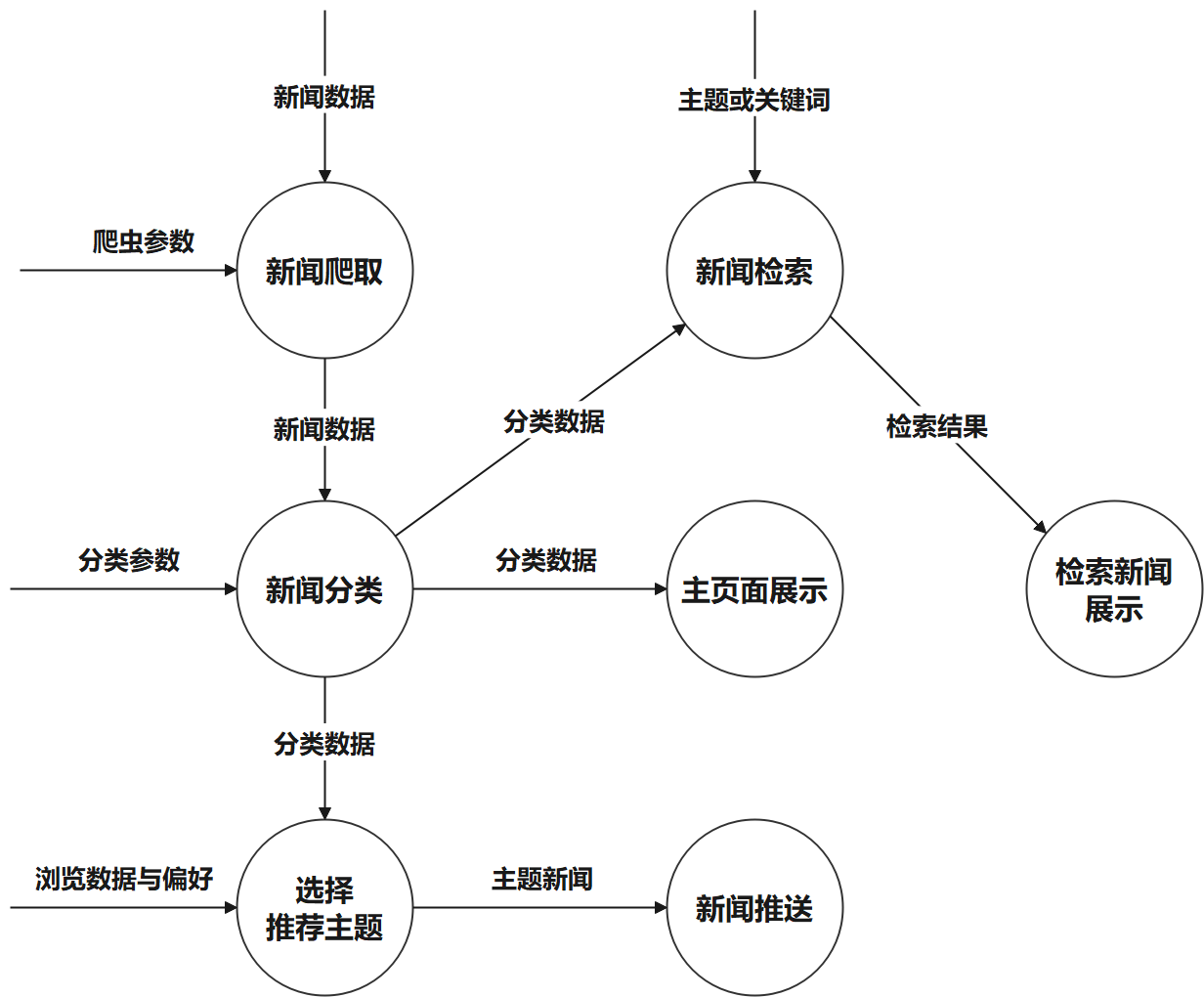


图 14基于变换流的数据流图

**2.基于变换流的软件结构分析**

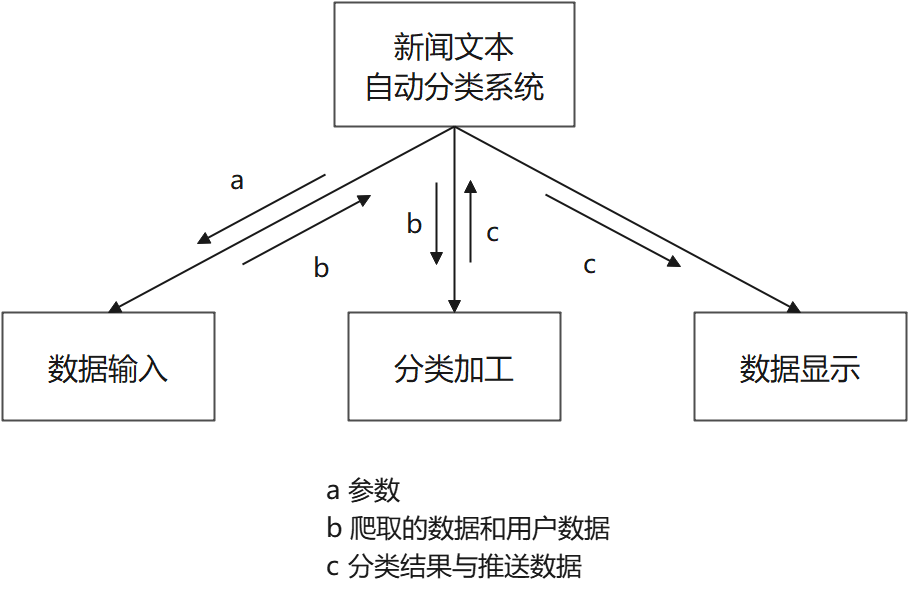


图 15顶层软件结构

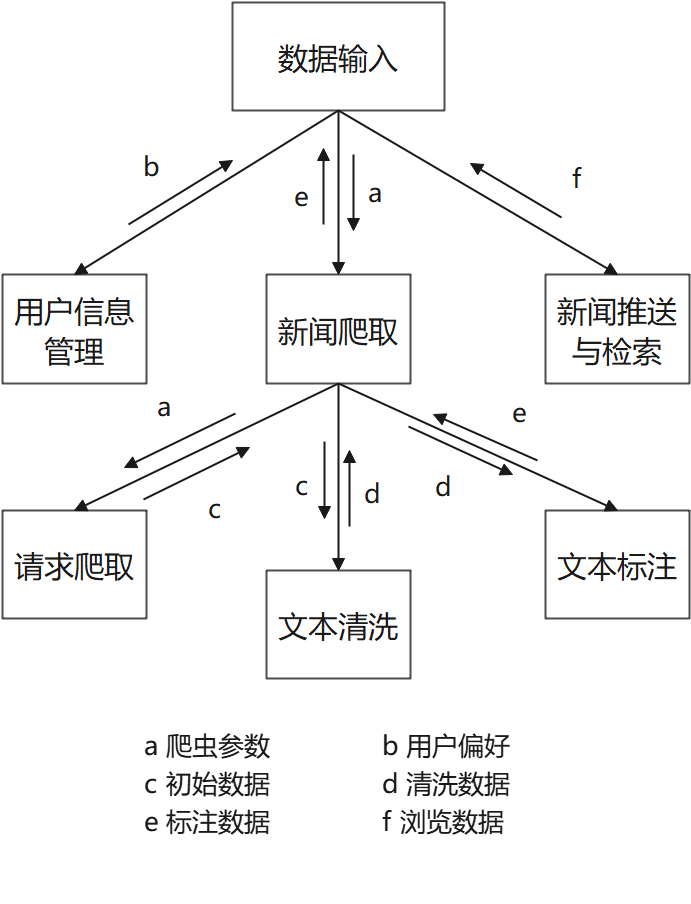


图 16数据输入结构

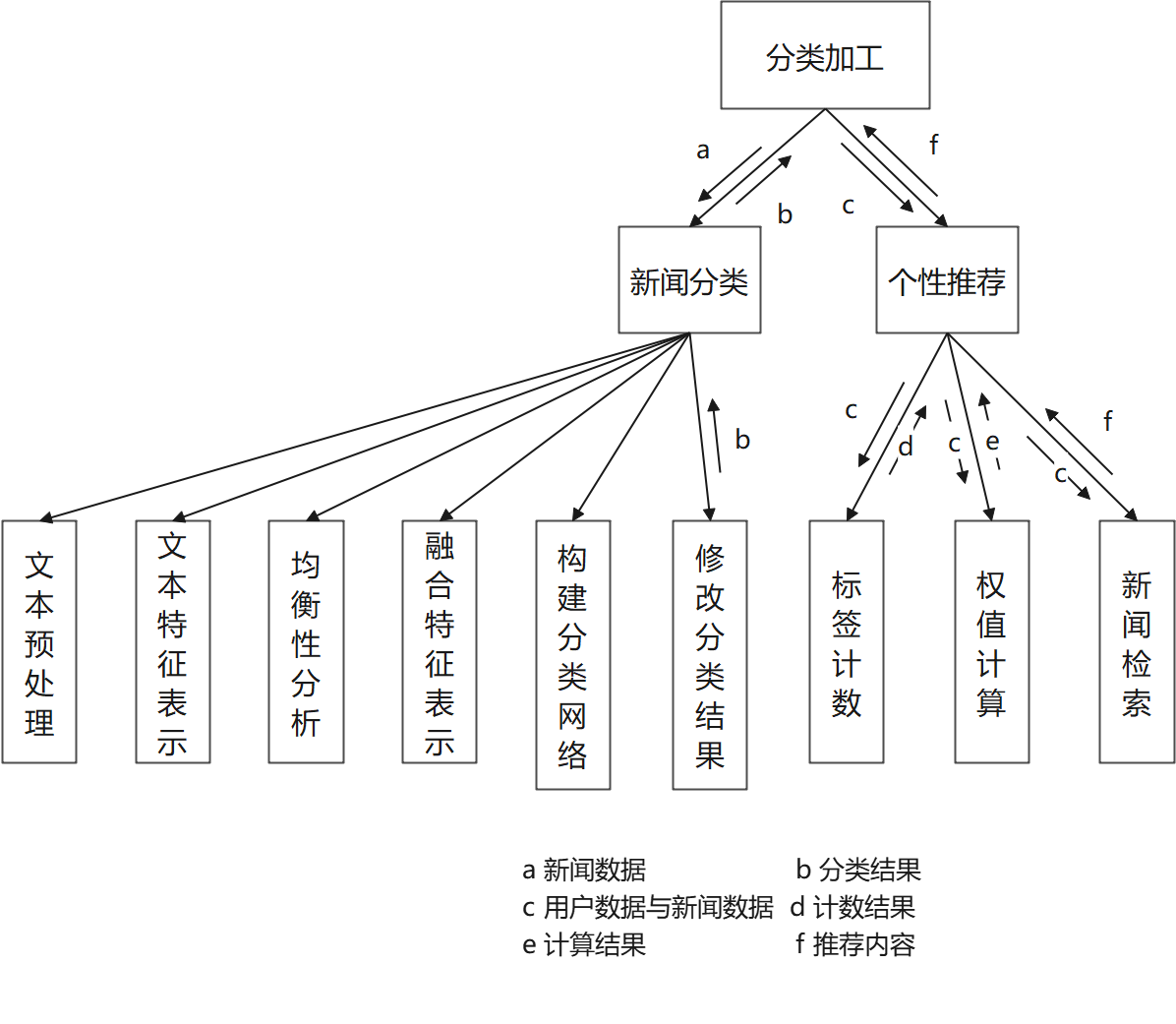


图 17分类加工结构

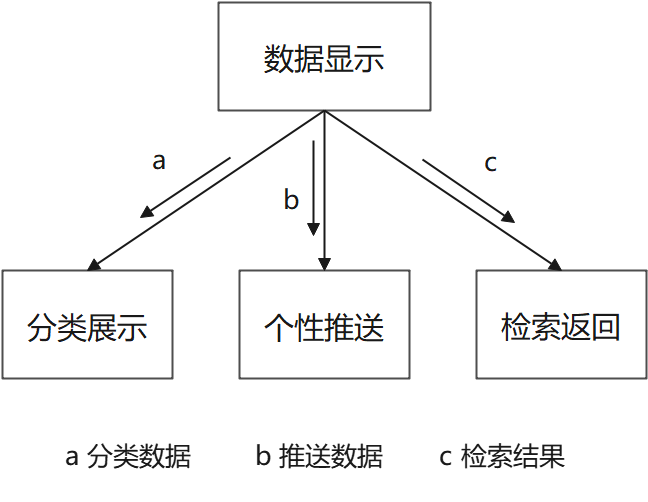


图 18数据显示结构

### **4.1.2 系统层次图**

新闻文本自动分类系统的顶层层次可以分为用户登录管理、新闻数据爬取、新闻文本分类、新闻个性推荐和个人信息管理五个模块。

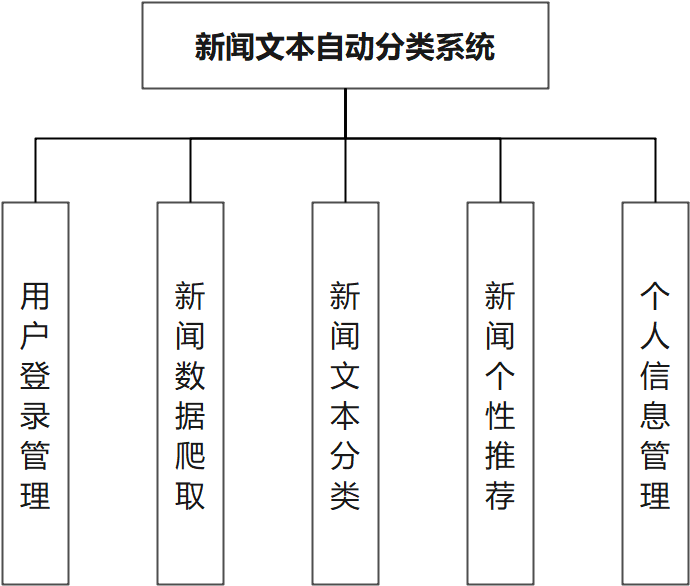


图 19系统顶层层次

其中用户登录管理可以分为账号注册、账号登录、找回密码和账号注销四个模块。

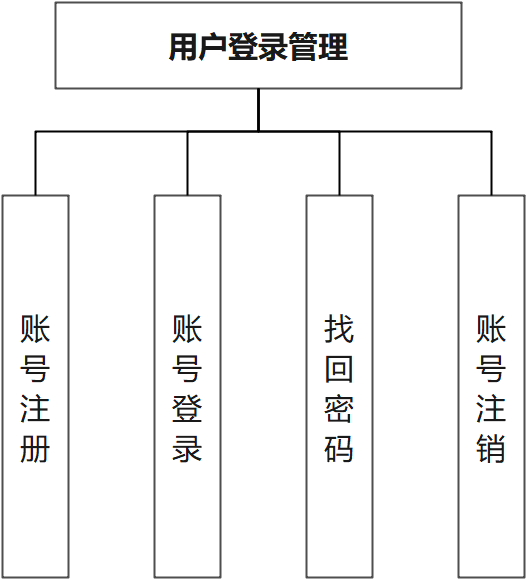


图 20用户登录管理层次

新闻数据爬取模块可以分为数据爬取、文本清洗和文本标注三个模块。文本清洗主要将缺失文本和错误数据清除。

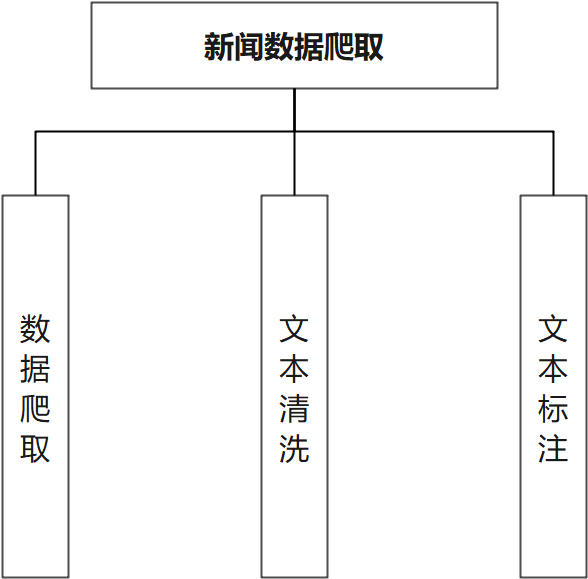


图 21新闻数据爬取层次

新闻文本分类模块可以分为新闻文本预处理、新闻文本特征表示、数据均衡性分析、新闻融合特征表示、构建分类网络和修改分类结果六个模块。

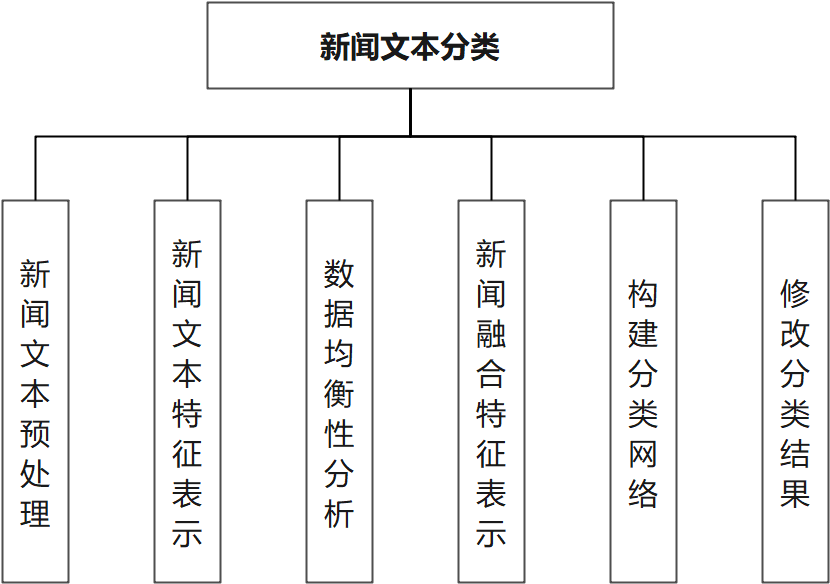


图 22新闻文本分类层次

新闻个性推荐模块可以分为标签计数、权重计算和新闻检索三个模块。

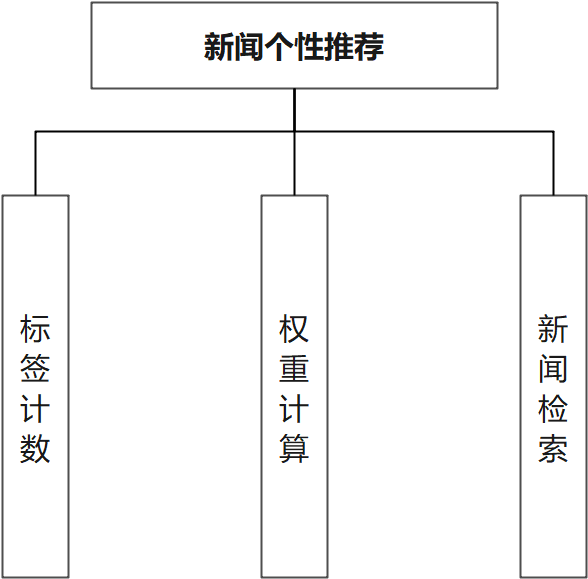


图 23新闻个性推荐层次

个人信息管理模块可以分为个人信息编辑和修改密码两个模块。

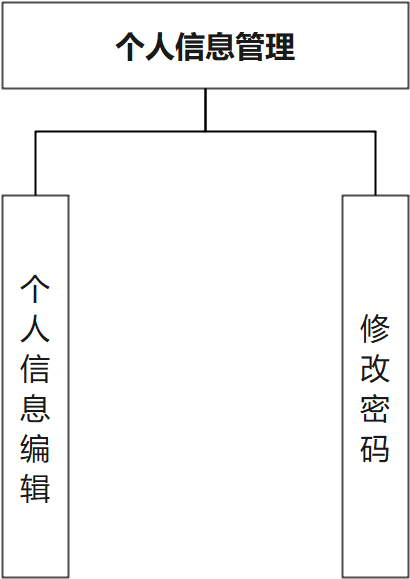


图 24个人信息管理层次

## 4.2 详细设计

新闻文本自动分类系统的详细设计主要介绍系统代表性模块和系统数据库的设计结构。代表性模块设计将介绍该模块的运作流程。

### **4.2.1 代表性模块设计**

**1. 用户登录与注册模块设计简介**

用户注册与登录是使用系统的必要一步，程序开始，用户选择“注册”或者“登录”，若选择登录，而用户并未注册，系统将引导用户进行注册，注册环节，用户需将账号与手机号绑定，发送手机验证码以完成注册。

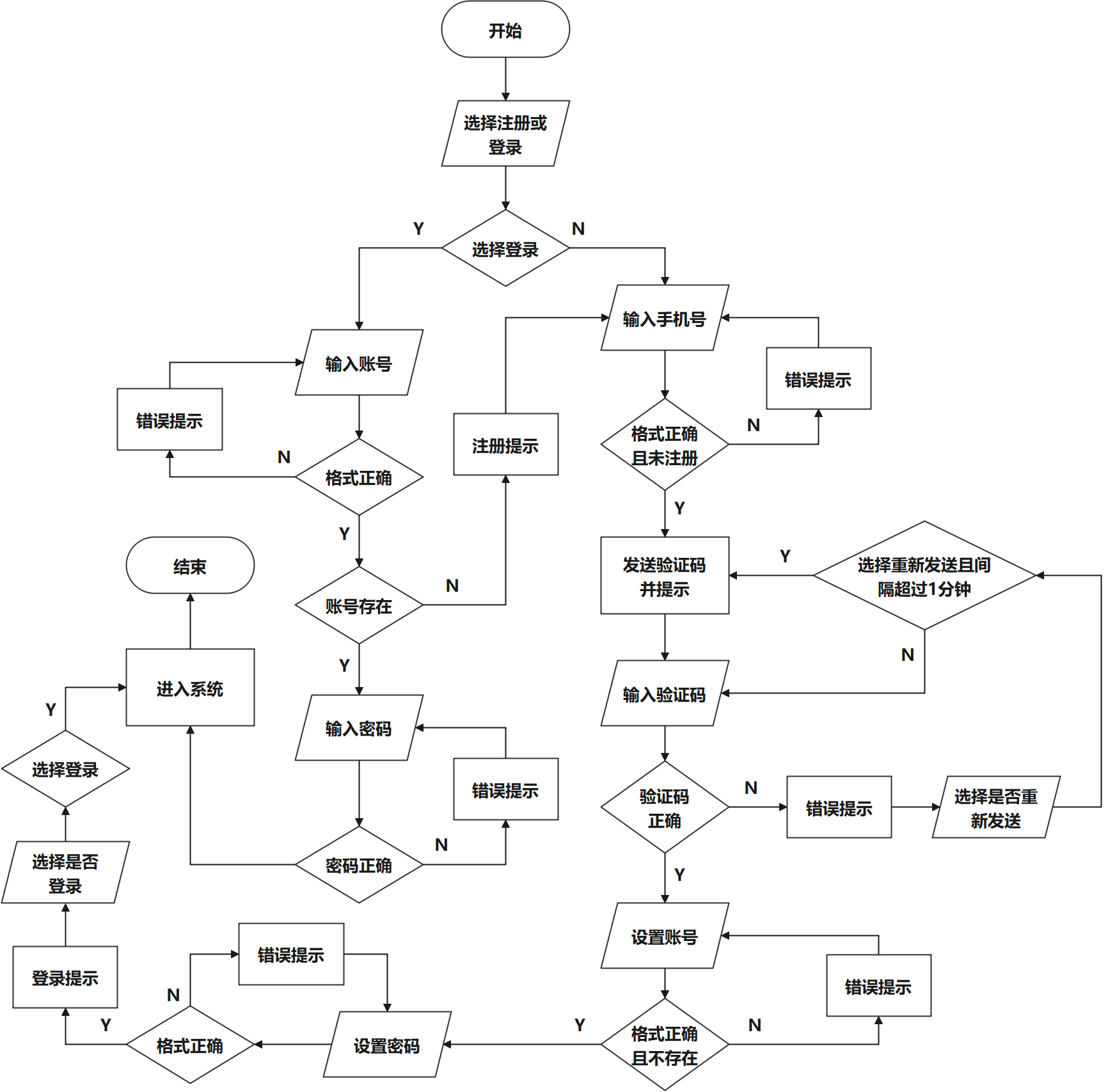
****

图 25用户登录与注册模块程序流程图

**2. 新闻个性推荐模块设计简介**

新闻个性推荐功能的设计将根据用户的偏好个性化推荐相关新闻，为用户提供更优质的使用体验。程序开始，用户填写“偏好”，在使用过程中，系统会记录用户的浏览记录。用户的偏好设置和浏览历史分别占据对应权重，若用户既未设置偏好，也没有浏览新闻，系统将随机推荐。

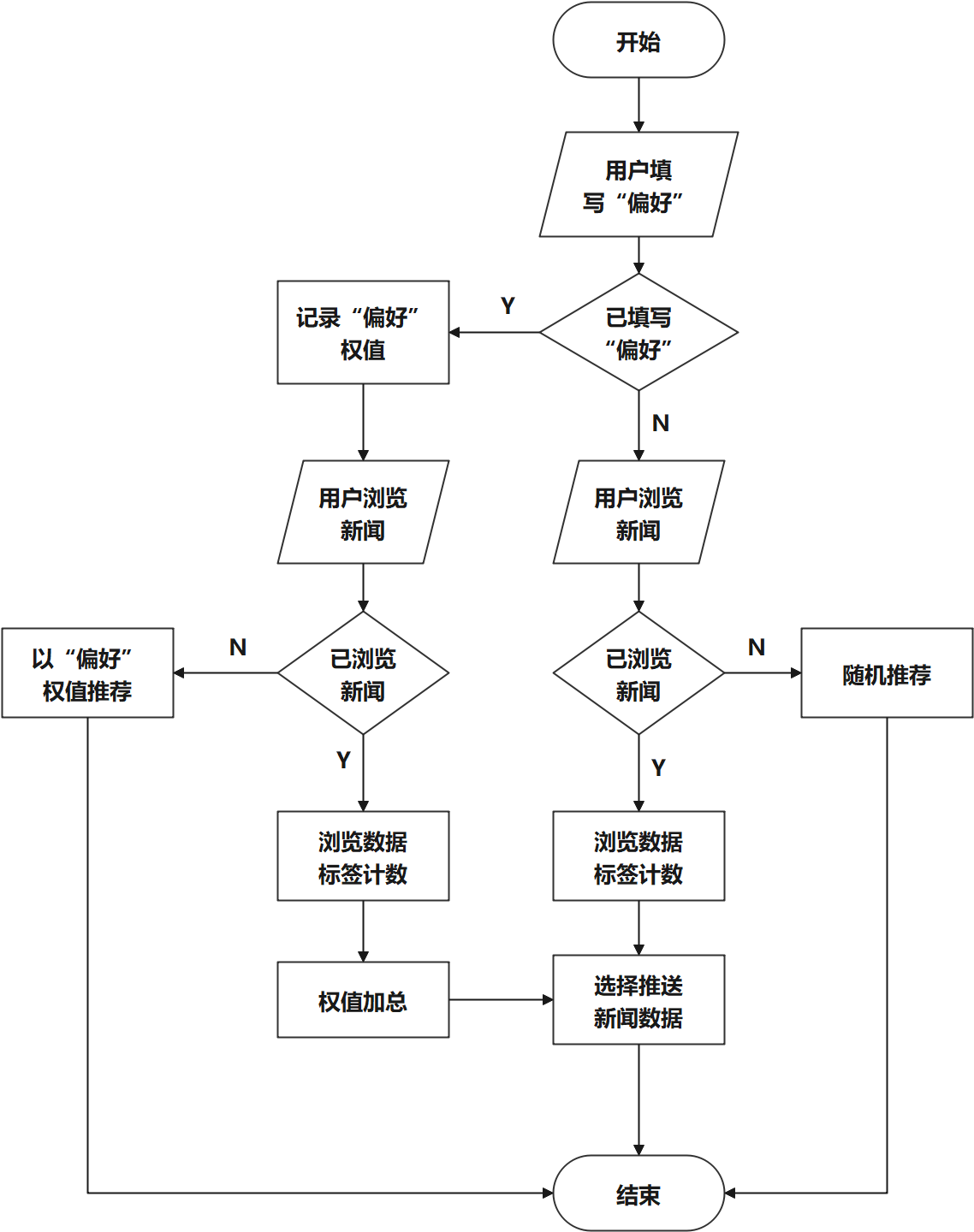
****

图 26新闻个性推荐模块程序流程图

### **4.2.2 系统数据库设计**

**1. 数据库的逻辑设计**

**（1）用户表设计**

表 14用户表数据库设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **名称** | **字段名** | **类型** | **备注** |
| **1** | 账号 | Account | char(8) | 主键 |
| **2** | 密码 | Password | Varchar(20) |  |
| **3** | 手机号 | Telephone | char(11) |  |
| **4** | 偏好 | Preference | Varchar(50) |  |
| **5** | 性别 | Sex | nchar(1) |  |
| **6** | 年龄 | Age | int(3) |  |
| **7** | 登录时间 | Log time | datetime |  |

**（2）管理员表设计**

表 15管理员表数据库设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **名称** | **字段名** | **类型** | **备注** |
| **1** | 账号 | Account | char(8) | 主键 |
| **2** | 密码 | Password | Varchar(20) |  |
| **3** | 手机号 | Telephone | char(11) |  |
| **4** | 管理员等级 | Rank | Int(1) |  |
| **5** | 登录时间 | Log time | datetime |  |

**（3）新闻表设计**

表 16新闻表数据库设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **名称** | **字段名** | **类型** | **备注** |
| **1** | 新闻ID | News ID | char(8) | 主键 |
| **2** | 新闻标题 | Title | Varchar(20) |  |
| **3** | 新闻正文 | Content | Text |  |
| **4** | 分类标签 | Label | Varchar(10) |  |
| **5** | 下载时间 | DL time | datetime |  |
| **6** | 浏览次数 | Views | Int(6) |  |
| **7** | 操作账号 | Account | char(8) | 外键 |

**（4）浏览记录表设计**

表 17浏览记录表数据库设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **名称** | **字段名** | **类型** | **备注** |
| **1** | 浏览编号 | His ID | char(8) | 主键 |
| **1** | 新闻ID | News ID | char(8) | 外键 |
| **2** | 账号 | Account | char(8) | 外键 |
| **4** | 分类标签 | Label | Varchar(10) |  |
| **5** | 浏览时间 | DL time | datetime |  |

**2. 数据库的表设计及表与表之间的关联**

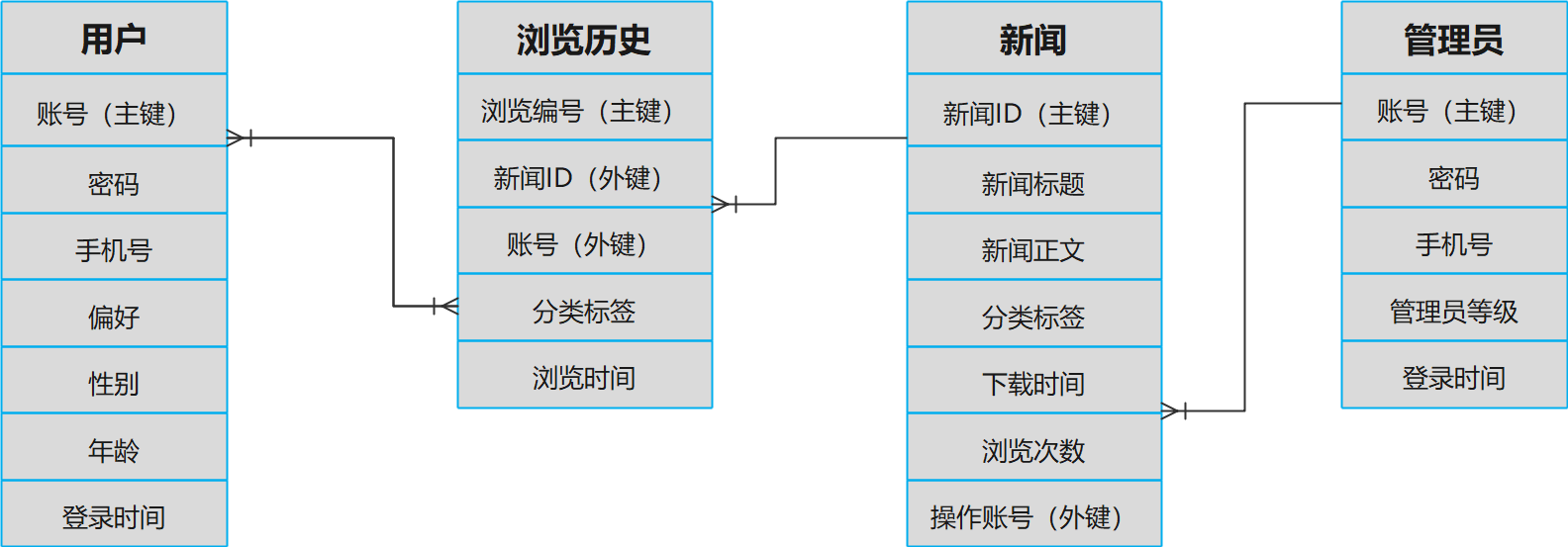


图 27数据库实体表联系

# 5 编码与测试

程序的质量主要取决于软件设计的质量。软件测试是保证软件质量的关键步骤，是对软件规格说明、设计和编码的最后复审。基于黑盒测试、白盒测试的方法，尽可能多的发现并排除软件中潜藏的错误，最终把一个高质量的软件系统交付给用户使用。

## 5.1 编码

### **5.1.1 编码规则简介**

编码规则是程序编码所要遵循的规则，要注意代码的正确性、稳定性、可读性。要避免使用不易理解的数字，用有意义的标识来替代，不要使用难懂的技巧性很高的语句。源程序中关系较为紧密的代码应尽可能相邻。

主要从排版、注释、命名、可读性、变量、函数过程、可测性、程序效率、质量保证、代码编译、代码测试等方面进行。

**1.排版**

①关键词和操作符之间加适当的空格。

②相对独立的程序块与块之间加空行

③较长的语句、表达式等要分成多行书写。

④划分出的新行要进行适应的缩进，使排版整齐，语句可读。

⑤长表达式要在低优先级操作符处划分新行，操作符放在新行之首。

**2.注释**

①注释要简单明了。

②边写代码边注释，修改代码同时修改相应的注释，以保证注释与代码的一致性。

③在必要的地方注释，注释量要适中。注释的内容要清楚、明了，含义准确，防止注释二义性。保持注释与其描述的代码相邻，即注释的就近原则。

④对代码的注释应放在其上方相邻位置，不可放在下面。

⑤对数据结构的注释应放在其上方相邻位置，不可放在下面；对结构中的每个域的注释应放在此域的右方；同一结构中不同域的注释要对齐。

⑥变量、常量的注释应放在其上方相邻位置或右方。

**3.可读性**

①避免使用不易理解的数字，用有意义的标识来替代。

②不要使用难懂的技巧性很高的语句。

③源程序中关系较为紧密的代码应尽可能相邻。

**4.程序效率**

①编程时要经常注意代码的效率。

②在保证软件系统的正确性、稳定性、可读性及可测性的前提下，提高代码效率。

③不能一味地追求代码效率，而对软件的正确性、稳定性、可读性及可测性造成影响。

④编程时，要随时留心代码效率；优化代码时，要考虑周全。

⑤要仔细地构造或直接用汇编编写调用频繁或性能要求极高的函数。

**5.代码测试**

①单元测试要求至少达到语句覆盖。

②单元测试开始要跟踪每一条语句，并观察数据流及变量的变化。

③清理、整理或优化后的代码要经过审查及测试。

### **5.1.2代表性模块示例**

**1.登录模块**

def log(self):  
 account\_input = self.ui.lineEdit\_account.text()  
 password\_input = self.ui.lineEdit\_password.text()  
 if self.ui.checkBox\_admin.isChecked():  
 user\_type = '管理员'  
 else:  
 user\_type = '用户'  
  
 if account\_input == '':  
 self.ui.label\_info.setText("请输入账号!")  
 else:  
 if password\_input == '':  
 self.ui.label\_info.setText("请输入密码!")  
 else:  
 if self.id\_exist(account\_input, user\_type):  
 if user\_type == '用户':  
 sql = 'select Password from user where Account = %s'  
 try:  
 cursor.execute(sql, (account\_input))  
 db.commit()  
 except:  
 db.rollback()  
 pw = cursor.fetchall()[0][0]  
 if pw == password\_input:  
 pass  
 else:  
 self.ui.label\_info.setText("密码错误!")  
 else:  
 sql = 'select Password from admin where Account = %s'  
 try:  
 cursor.execute(sql, (account\_input))  
 db.commit()  
 except:  
 db.rollback()  
 pw = cursor.fetchall()[0][0]  
 if pw == password\_input:  
 self.ui.close()  
 self.admin = Table\_admin()  
 self.admin.ui.setWindowTitle("TextClassifier")  
 self.admin.ui.setWindowIcon(QtGui.QIcon('icon.ico'))  
 self.admin.ui.show()  
 else:  
 self.ui.label\_info.setText("密码错误!")  
 else:  
 self.ui.label\_info.setText("账号不存在，请先注册账号!")

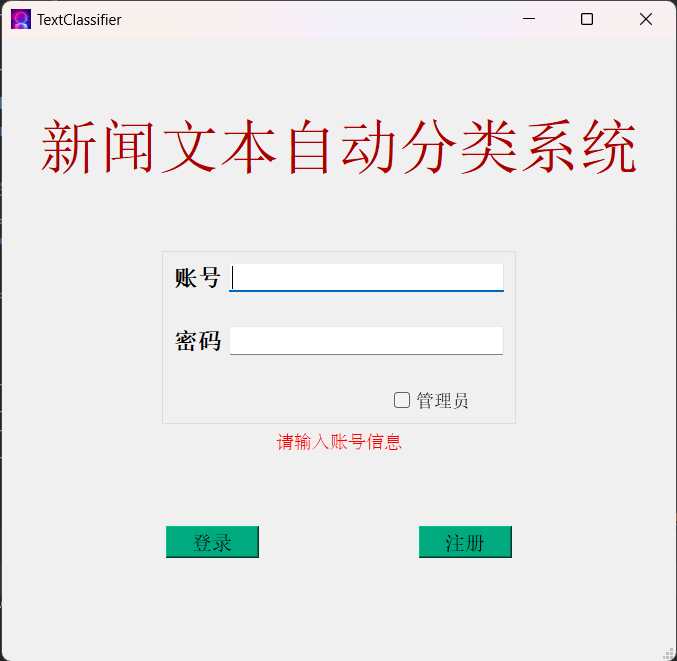


图 28登录界面

**2.预爬取模块**

def pre\_crawl(self):  
 self.ui.label\_3.setText('请稍候')  
 self.ui.textBrowser\_preview.clear()  
 news\_type = self.ui.comboBox.currentText()  
 url = self.get\_url(news\_type)  
 num = self.ui.spinBox\_n.value()  
 headers = { 'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) \  
 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/113.0.0.0 Safari/537.36'}  
 try:  
 res = requests.get(url, headers=headers)  
 soup = BeautifulSoup(res.content, 'html.parser')  
 content = soup.prettify()  
 l = self.get\_len(news\_type)  
 content = content[l:-2]  
 with open('temp.json', 'w', encoding='utf-8') as fp:  
 fp.write(content)  
 obj = json.load(open('temp.json', encoding='utf-8'))  
 title\_lst = jsonpath.jsonpath(obj, '$..title')  
 url\_lst = jsonpath.jsonpath(obj, '$..url')  
 if num > len(title\_lst):  
 num = len(title\_lst)  
 self.ui.spinBox\_n.setValue(num)  
 self.url\_lst = url\_lst[:num]  
 self.title\_lst = title\_lst[:num]  
 for i in range(num):  
 self.ui.textBrowser\_preview.append('【' + str(i+1) + '】' + title\_lst[i] + '\n')  
 self.pre\_crawl\_flag = 1  
 self.ui.label\_3.setText('预爬取成功!')  
 except Exception as a:  
 self.ui.textBrowser\_preview.setText('预爬取失败!')

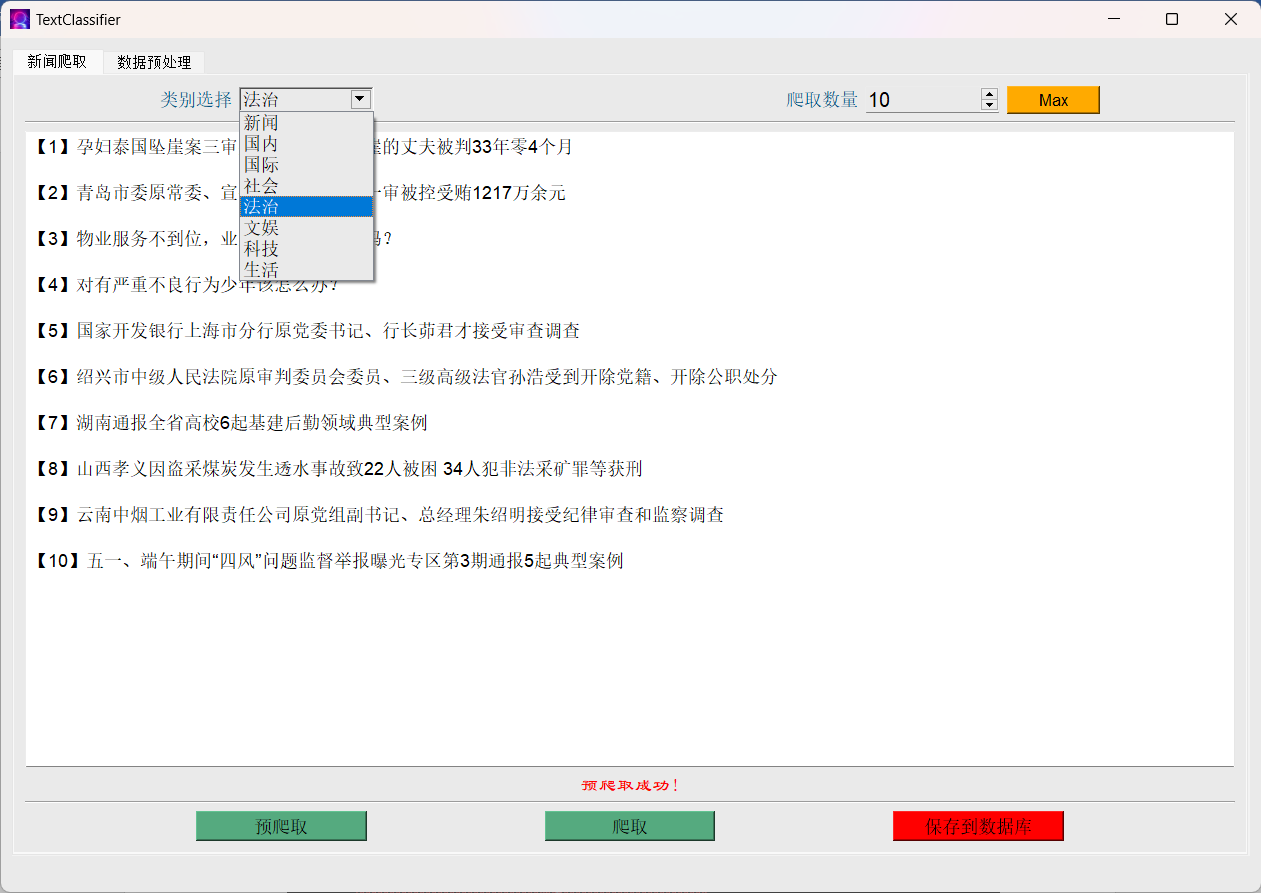


图 29新闻预爬取界面

**3.爬取模块**

def crawl(self):  
 self.now\_time = datetime.datetime.now()  
 headers = {'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) \  
 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/113.0.0.0 Safari/537.36'}  
 if self.pre\_crawl\_flag == 0:  
 self.ui.label\_3.setText('请先完成预爬取!')  
 else:  
 self.ui.label\_3.setText('爬取中，请耐心等待!')  
 self.ui.textBrowser\_preview.clear()  
 lst = self.url\_lst  
 lose\_num = 0  
 self.index\_lst = []  
 self.content\_lst = []  
 for i in range(len(lst)):  
 try:  
 res = requests.get(url=lst[i], headers=headers)  
 soup = BeautifulSoup(res.content, 'html.parser')  
 text = soup.select\_one('.content\_area').text.strip()  
 text\_ = text.split('　　')  
 text\_line = ''  
 for t in text\_:  
 text\_line += '\n　　' + t  
 self.content\_lst.append(text)  
 self.ui.textBrowser\_preview.append('>>>>>>>' + str(i+1) + text\_line +'\n \n \n')  
 self.index\_lst.append(i)  
 except:  
 lose\_num += 1  
 self.crawl\_flag = 1  
 self.ui.label\_3.setText('爬取失败' + str(lose\_num) + '条!')



图 30新闻爬取

**3.数据预处理**

def data\_request(self):  
 try:  
 cursor.execute('select title, url from news')  
 result = cursor.fetchall()  
 row = cursor.rowcount  
 vol = len(result[0])  
 self.ui.tableWidget.setRowCount(row)  
 self.ui.tableWidget.setColumnCount(vol)  
 self.ui.tableWidget.setHorizontalHeaderLabels(['title', 'url'])  
 for i in range(row):  
 for j in range(vol):  
 data = QTableWidgetItem(str(result[i][j]))  
 self.ui.tableWidget.setItem(i, j, data)  
 self.ui.tableWidget.verticalHeader().setDefaultSectionSize(20)  
 self.ui.tableWidget.horizontalHeader().setSectionResizeMode(QHeaderView.Stretch)  
 self.ui.tableWidget.setAlternatingRowColors(True)  
 except:  
 self.ui.tableWidget.clearContents()  
 self.ui.label\_4.setText('未找到数据!')  
  
def content\_view(self):  
 self.ui.textBrowser\_content.clear()  
 try:  
 select\_row = self.ui.tableWidget.currentRow()  
 select\_url = self.ui.tableWidget.item(select\_row, 1).text()  
 sql = 'select content from news where url = %s'  
 try:  
 cursor.execute(sql, select\_url)  
 db.commit()  
 except:  
 db.rollback()  
 select\_content = cursor.fetchall()[0][0]  
 select\_content = select\_content.split('　　')  
 for t in select\_content:  
 self.ui.textBrowser\_content.append('　　' + t)  
 except:  
 self.ui.label\_4.setText('请先选中新闻!')  
  
def delete\_news(self):  
 self.ui.textBrowser\_content.clear()  
 try:  
 select\_row = self.ui.tableWidget.currentRow()  
 select\_url = self.ui.tableWidget.item(select\_row, 1).text()  
 sql = 'select title from news where url = %s'  
 try:  
 cursor.execute(sql, select\_url)  
 db.commit()  
 except:  
 db.rollback()  
 select\_title = cursor.fetchall()[0][0]  
 sql = 'delete from news where title = %s'  
 try:  
 cursor.execute(sql, select\_title)  
 db.commit()  
 except:  
 db.rollback()  
 self.ui.textBrowser\_content.setText('已删除《' + select\_title + '》')  
 self.data\_request()  
 except:  
 self.ui.label\_4.setText('请先选中新闻!')

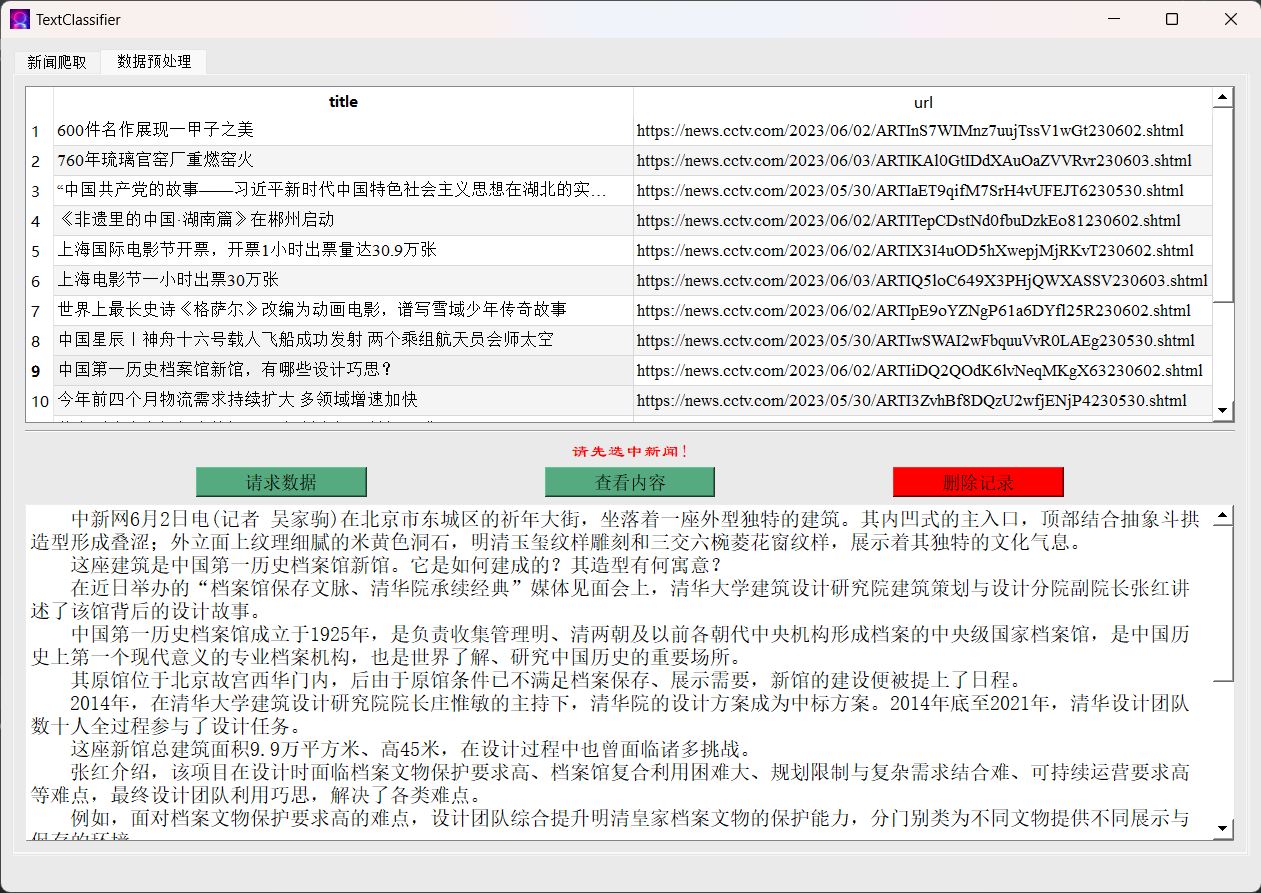


图 31数据预处理界面

## 5.2测试

在软件初步成型时应该进行软件测试。软件测试是为了发现错误而执行程序的过程。或者说，软件测试是根据软件开发各阶段的规格说明和程序的内部结构而精心设计一批测试用例（即输入数据及其预期的输出结果），并利用这些测试用例去运行程序，以发现程序错误的过程。

总的来说以最少的时间和人力，系统地找出软件中潜在的各种错误和缺陷，以考虑用户是否能接受该产品。

**软件测试规范：**

①“应早尽早”地进行软件测试。

②测试用例应由测试输入数据和与之对应的预期输出结果这两部分组成。主要是关注用户需求功能和测试使用功能是否正常。

③负责编程部分的队员应避免检查自己的程序。条件限制由非编程的队员进行测试工作；

④把Pareto原理应用到软件测试中，充分注意测试中产生的集群现象，80%的错误可能都来自20%的模块。根据这个规律，应当对错误群集的程序段进行重点测试，以提高效率。

⑤妥善保存测试计划、测试用例、出错统计和最终分析报告，为未来的维护提供方便。

**软件测试策略：**

采用静态测试与动态测试相结合，其中动态测试使用白盒法与黑盒法。

静态测试：针对团队之前拟定的目标，采用人工检测和计算机辅助静态分析对程序进行检测，发现已经编写的代码中的错误，寻找既定路线的更优解；

动态测试：通过运行程序尝试发现错误。针对产品应该具有的功能通过黑盒测试检验功能是否符合要求；针对产品的内部工作过程，通过白盒测试检验每种内部操作是否按规定要求正常进行。

### **5.2.1 白盒测试**

**1.登录模块**

**（1）登录模块独立路径分析**

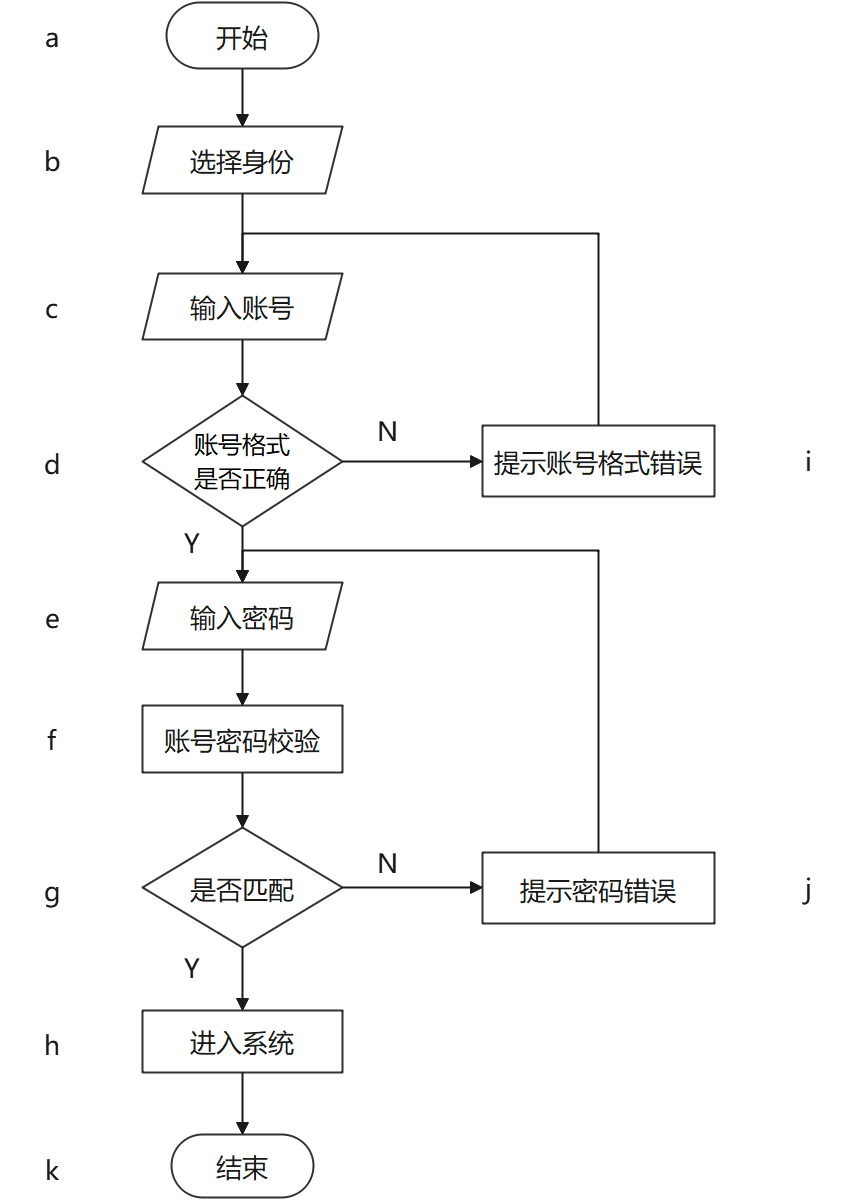


图 32登录模块路径分析

根据以上流程图可得所有的路径：

①a-b-c-d-e-f-g-h-k;

②a-b-c-d-i-c-d-e-f-g-h-k;

③a-b-c-d-e-f-g-j-e-f-g-h-k;

④a-b-c-d-i-c-d-e-f-g-j-e-f-g-h-k.

**（2）登录模块测试用例设计**

根据以上路径设计测试用例如下：

①账号格式正确；账号密码通过验证；

②账号格式错误；再次输入账号密码通过验证；

③账号格式正确；密码错误；

④账号格式错误；再次输入密码错误。

**2.爬取模块**

**（1）爬取模块独立路径分析**

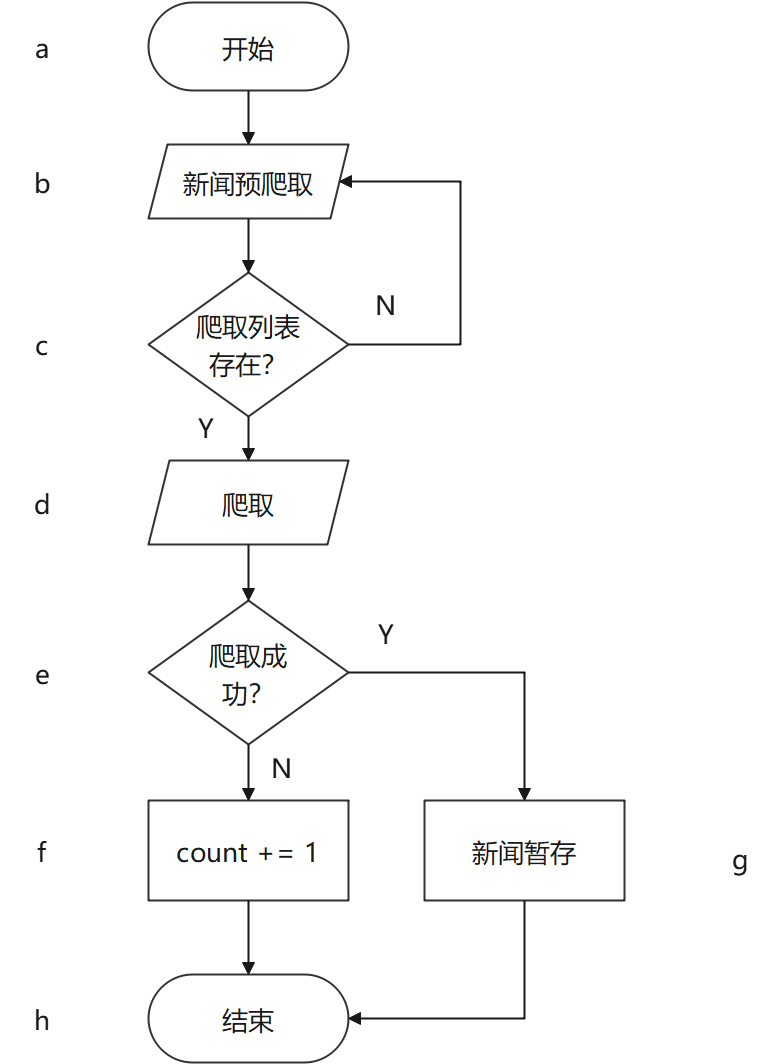


图 33新闻爬取模块路径分析

根据以上流程图可得所有的路径：

①a-b-c-d-e-f-h；

②a-b-c-d-e-f-g-h；

③a-b-c-b-c-d-e-f-g-h；

③a-b-c-b-c-d-e-f-h；

**（2）爬取模块测试用例设计**

根据以上路径设计测试用例如下：

①先进行预爬取，后爬取成功；

②不进行预爬取，直接进行爬取，且爬取成功；

③先进行预爬取，后爬取失败；

④不进行预爬取，直接进行爬取，且爬取失败。

**（3）爬取模块测试情况分析**

①爬取失败计数count += 1;

②提醒先进行预爬取，新闻暂存；

③提醒先进行预爬取，新闻暂存;

④提醒先进行预爬取，爬取失败计数count += 1;

**5.2.2 黑盒测试**

黑盒测试着重测试软件功能，黑盒测试并不能取代白盒测试，黑盒测试与白盒测试相互补充，黑盒测试可能发现白盒测试不容易发现的其他类型的错误。黑盒测试是从用户的角度出发，设计相关的测试条件，发现具体的错误。如功能不正确、界面错误、数据结构错误或外部数据库访问错误及性能错误。

**1. 边界值分析**

边界是指，对于输入和输出等价类而言，稍高和稍低于其边界值的一些特定情况。经验得知，大量的错误是发生在输入或输出范围的边界上，而不是在输入范围的内部。

边界值分析方法思想：确定边界之后，选取正好等于、刚刚大于或刚刚小于边界的值作为测试数据，而不是选取等价类中典型值或任意值作为测试数据。

（1）登录模块边界值分析

* 输入账号信息大于8位数字,密码正确

输入：

账号：123456789

密码：12345678aabb

预期输出：

拒绝登录

* 输入账号信息低于8位数字

输入：

账号：123456

密码：12345678aabb

预期输出：

拒绝登录

* 输入账号信息正确，密码仅含数字

输入：

账号：12345678

密码：123456789

预期输出：

拒绝登录

* 输入账号信息正确，密码正确

输入：

账号：12345678

密码：12345678aabb

预期输出：

登录成功

# 6 系统使用说明

本系统使用python语言开发，使用pyqt5程序包设计系统界面，利用MySQLWorkbench连接数据库，最后使用pyinstaller将程序打包。

## 6.1 系统运行环境和配置

表 18 运行配置与环境

|  |  |
| --- | --- |
| 项目 | 详情 |
| 开发硬件环境 | AMD Ryzen 5 3500U 2.10GHz 16.0GB内存 |
| 运行软件环境 | PC |
| 操作系统 | Windows11家庭中文版 |
| 开发环境与工具 | PyCharm Community Edition 2022.1.3;Qt Designer |
| 编程语言 | python |

## 6.2 系统操作说明

### **6.2.1 登录模块说明**

以下是登录注册模块截图，用户账号由8位数字组成，密码长度在8-16位字符，登录界面和账号注册界面右下角都有勾选框，用户可选择注册为普通用户或管理员，并且登录对应账号。

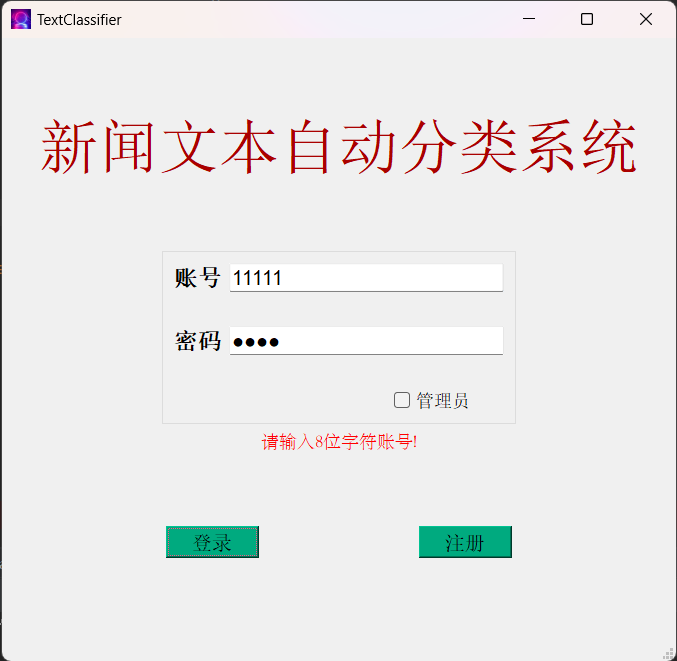


图 34 登录界面



图 35 注册界面

### **6.2.2 新闻爬取模块说明**

新闻爬取模块大致分为两块，新闻的预爬取和正式爬取，用户（管理员）可选择爬取新闻的类别和爬取数量（最大默认为80条），点击“预爬取”按钮，程序开始获取新闻标题和各条新闻的url，接着点击“爬取”按钮，程序开始逐条爬取新闻，并计数爬取失败的条数，最后点击“保存到数据库”，若数据库已经保存了某条新闻，则相同的新闻不会二次入库，并提示保存失败条数。

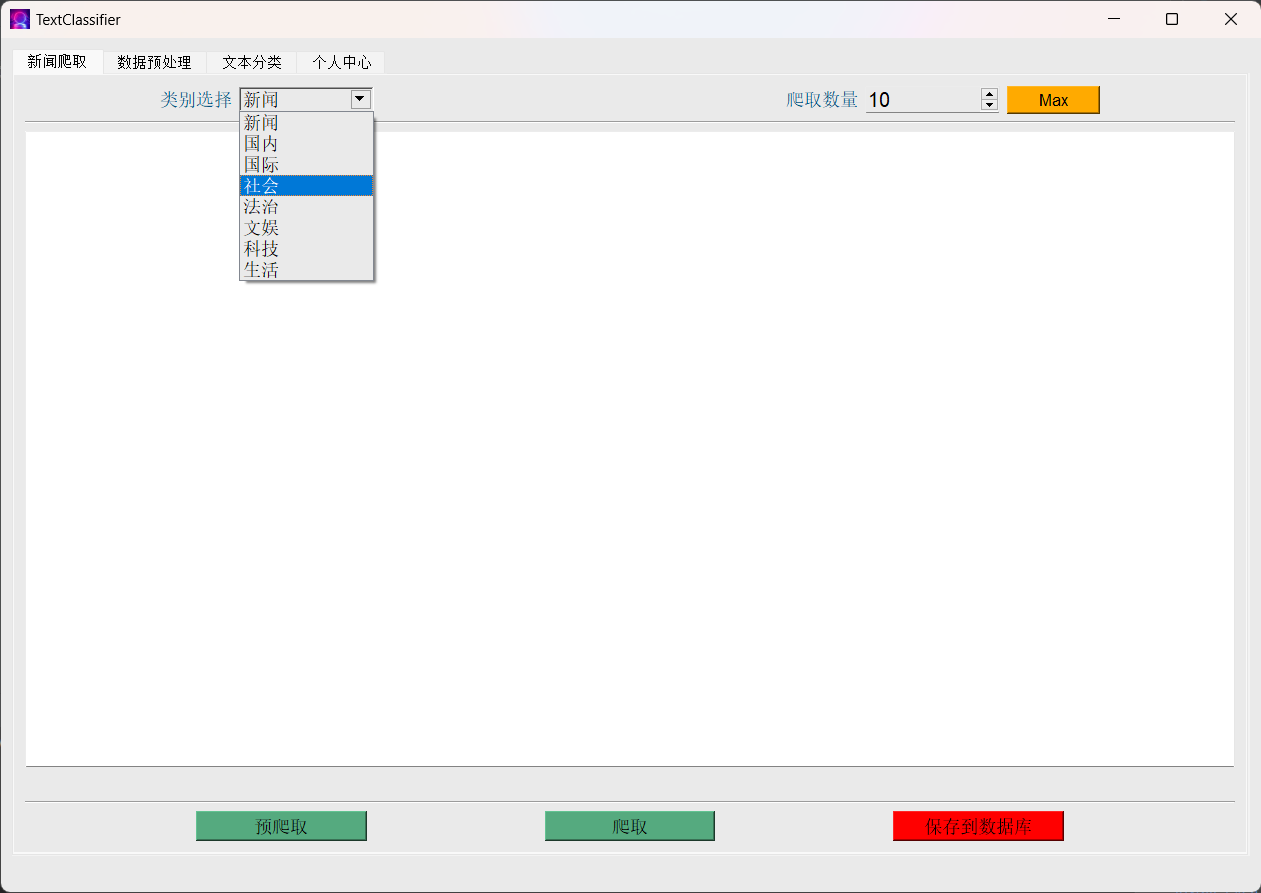


图 36新闻爬取（1）



图 37 新闻爬取（2）



图 38 新闻爬取（3）

### **6.2.3 数据预处理模块说明**

数据预处理主要用于处理乱码数据和内容缺失数据，进入“数据预处理”界面，用户点击“刷新”按钮获取最新数据，用户可查看到各新闻标题以及新闻链接，选中列表中的新闻，点击“查看内容”按钮，可在左下角文本显示框中查看新闻内容，点击“词云”按钮，可对新闻文本生成词云，对于缺失数据，用户可在选中后点击“删除记录”按钮删除对应数据。



图 39 数据预处理（1）

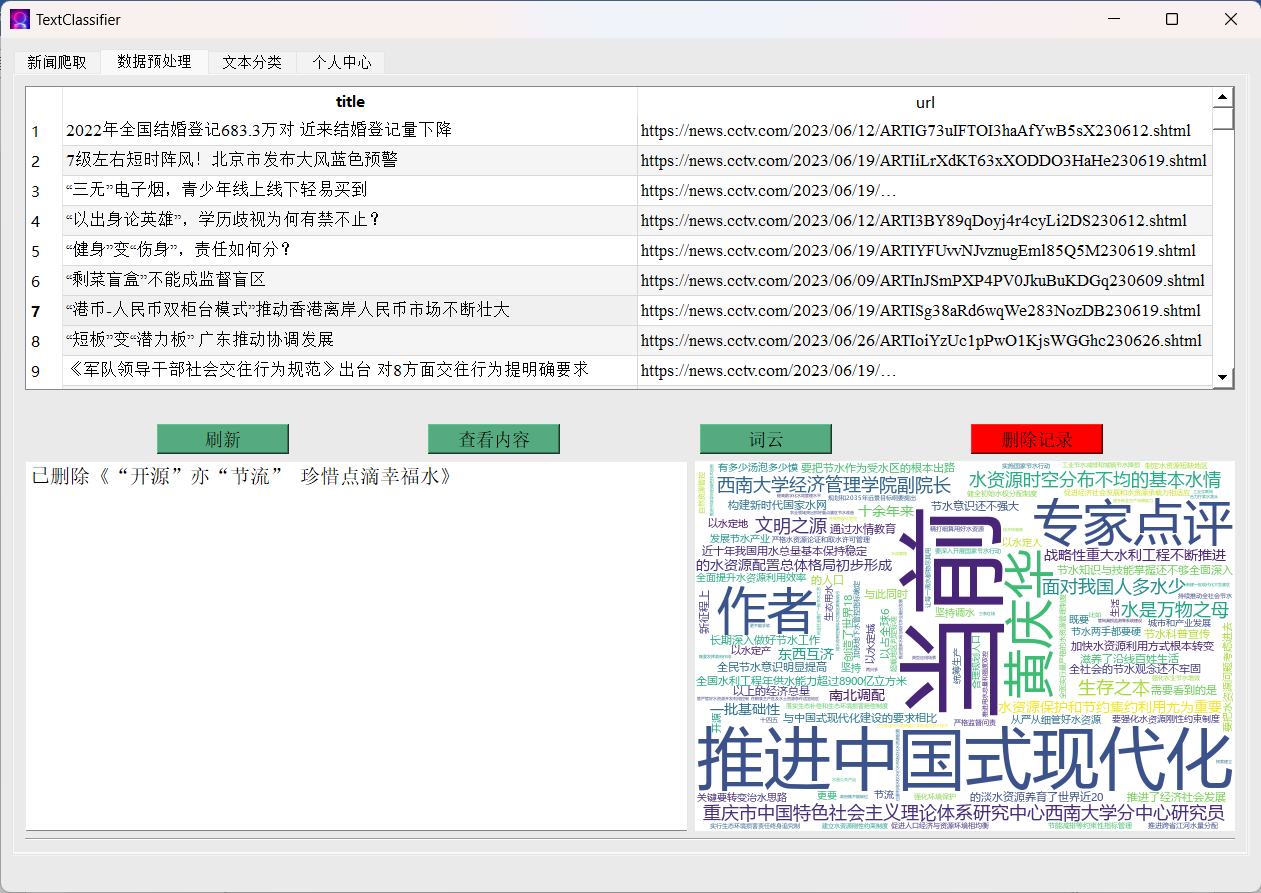


图 40 数据预处理（2）

### **6.2.4 新闻分类模块说明**

进入“文本分类”模块，用户点击“刷新”按钮，获取最新数据，点击“执行分类”按钮即可对新闻进行分类（程序已保存训练好的模型），点击“重置”按钮会将当前管理员管理的所有新闻分类信息重置，最后，若对分类有异议，可自行更改。

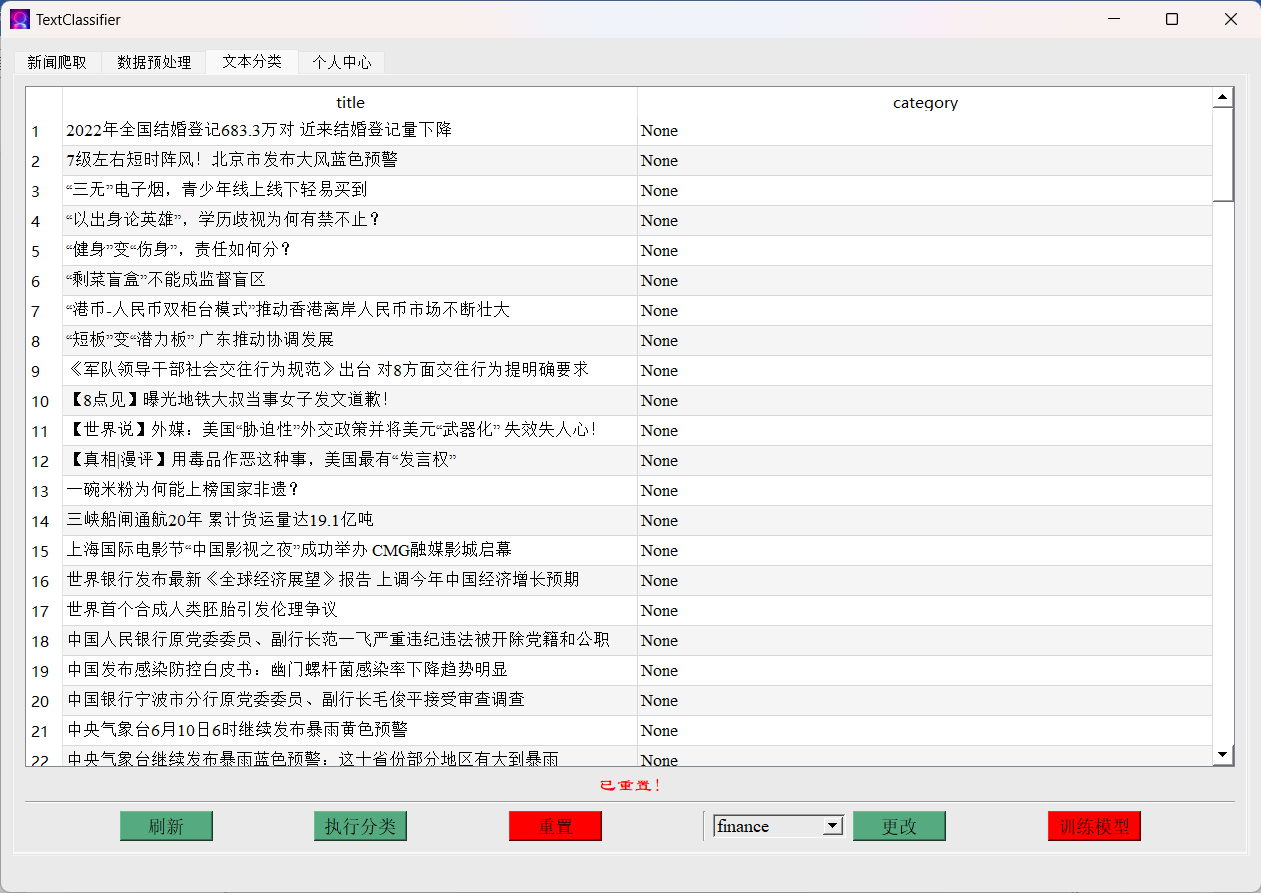


图 41新闻分类（1）



图 42 新闻分类（2）

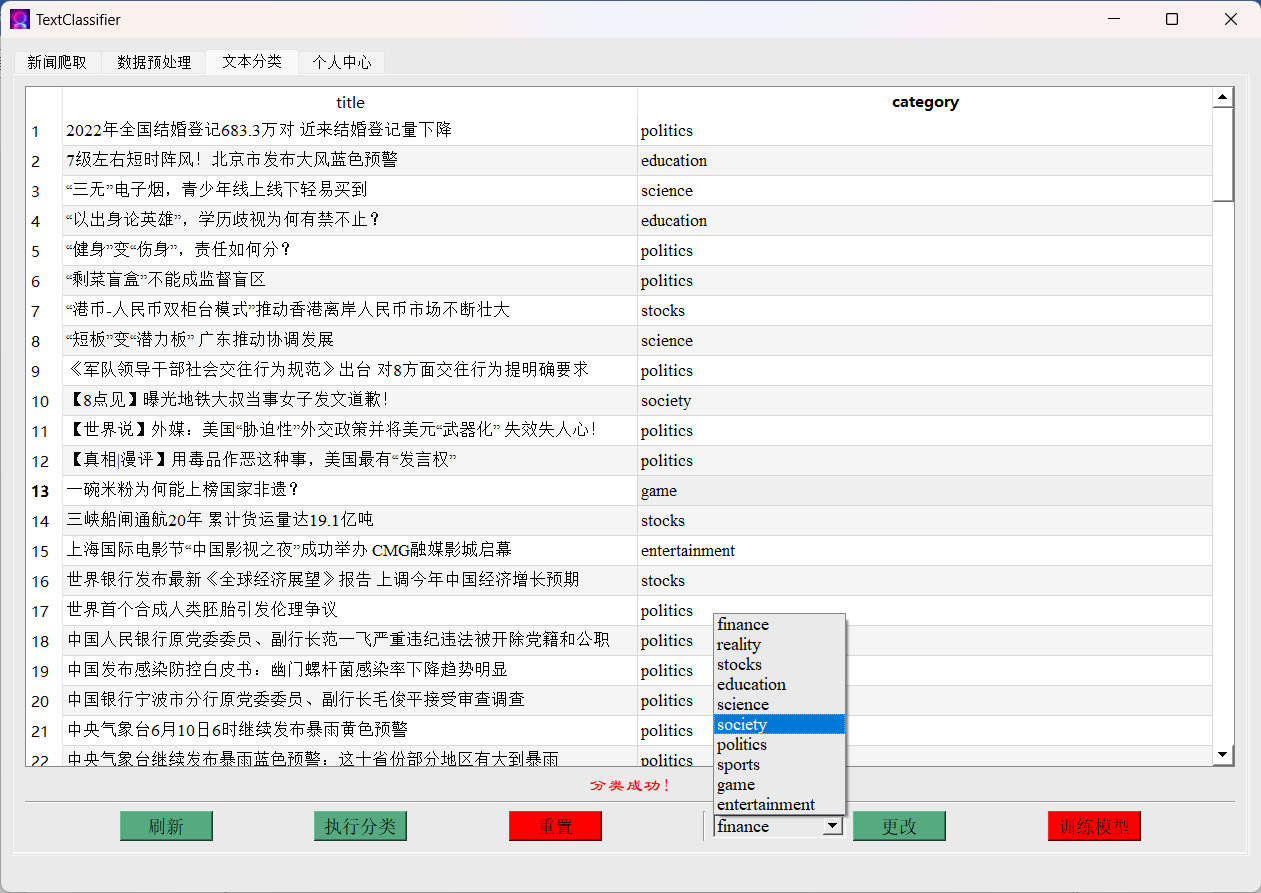


图 43 新闻分类（3）

### **6.2.5 管理员个人中心模块说明**

管理员个人中心界面所显示的“贡献值”是管理员所管理的所有新闻的计数，左下的表格是对应类别的计数。点击“编辑个人信息”按钮即可跳转信息编辑界面，提供修改昵称和个人简介以及密码修改服务。点击“登出账号”将退出当前账号，跳转至登录界面。

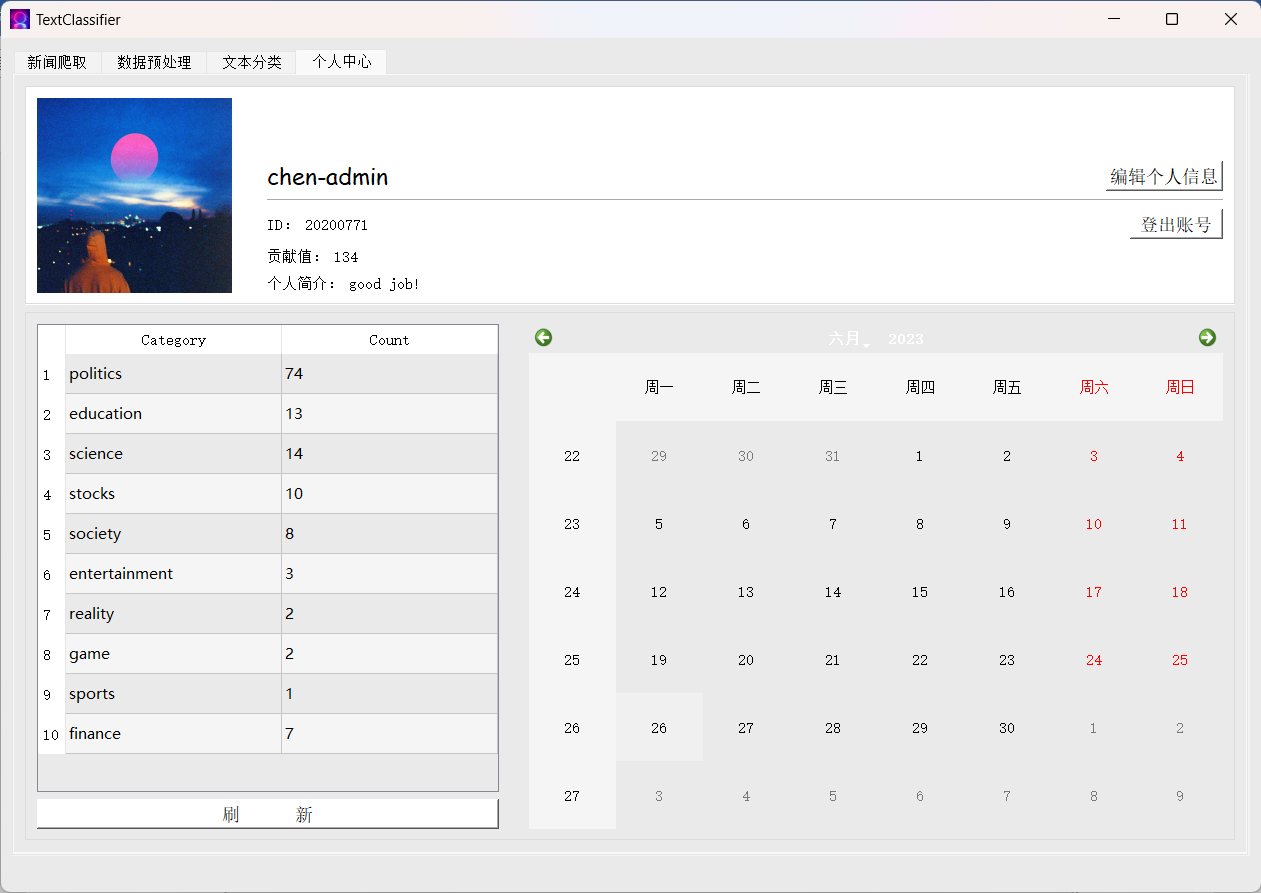


图 44 管理员个人中心

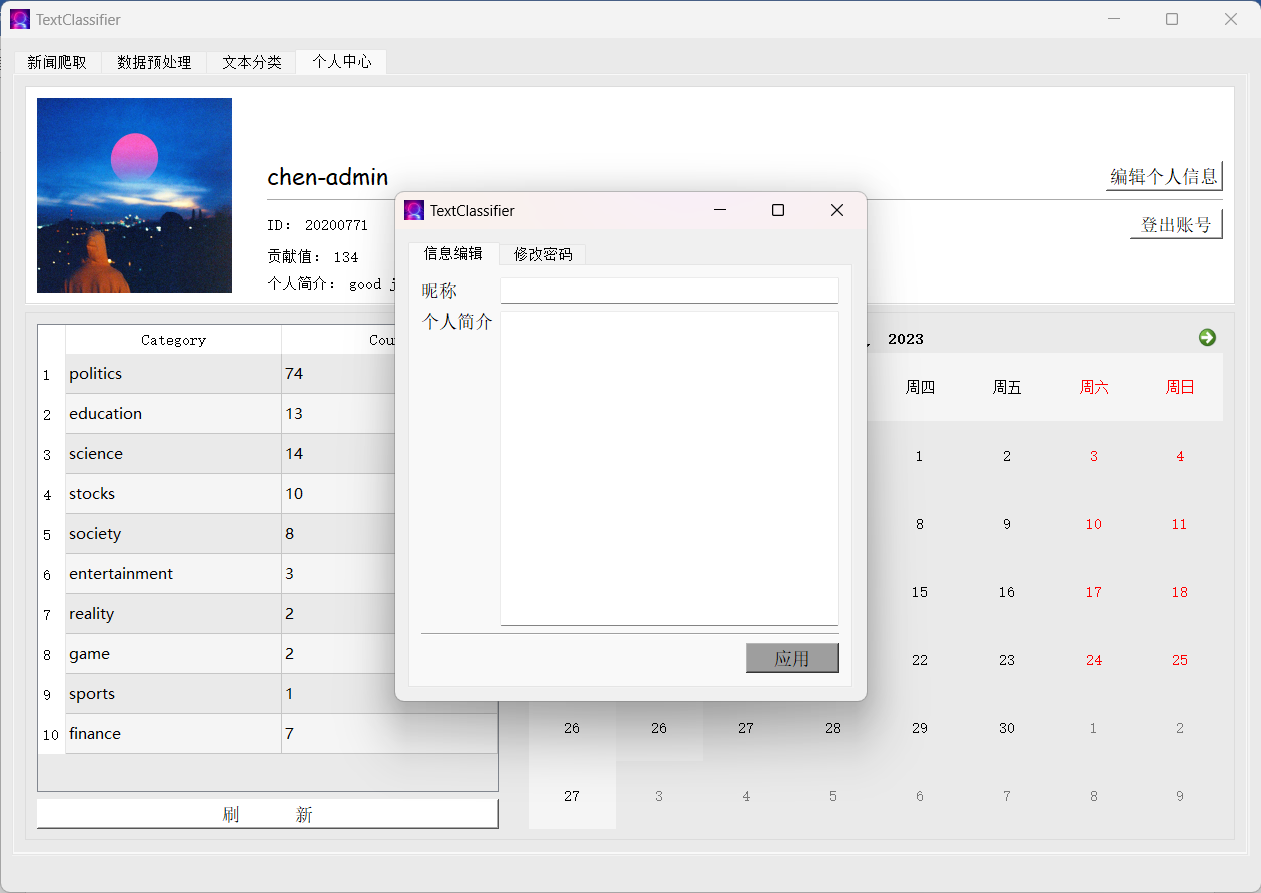


图 45 编辑信息

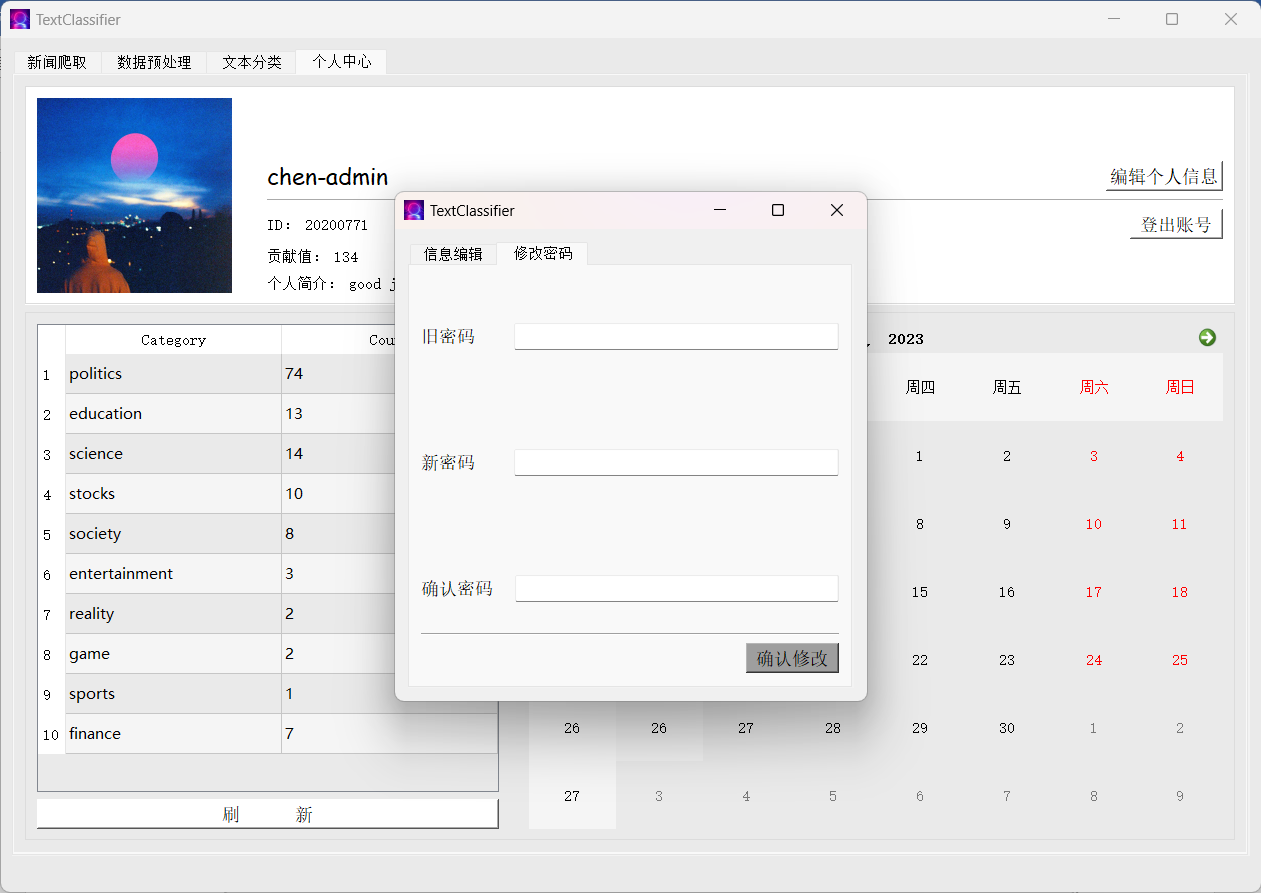


图 46 修改密码

### **6.2.6 用户浏览模块说明**

通过登录用户账号，进入用户主页面，系统自动获取当前所有新闻数据（所有管理员管理的数据），用户可通过搜索和选择类别来选择新闻。选中对应新闻，点击“查看内容“，可在右边文本框显示新闻内容。

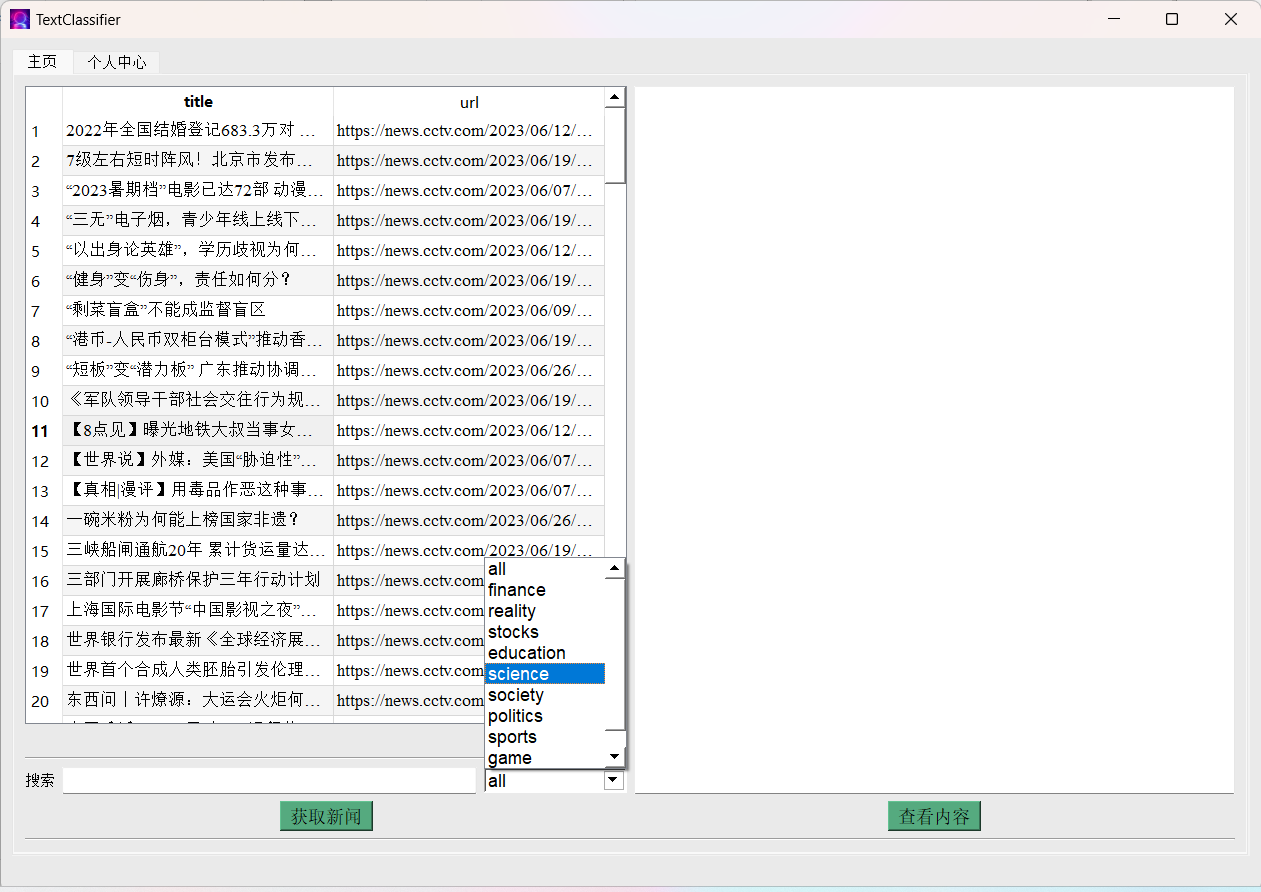


图 47用户主页面（1）

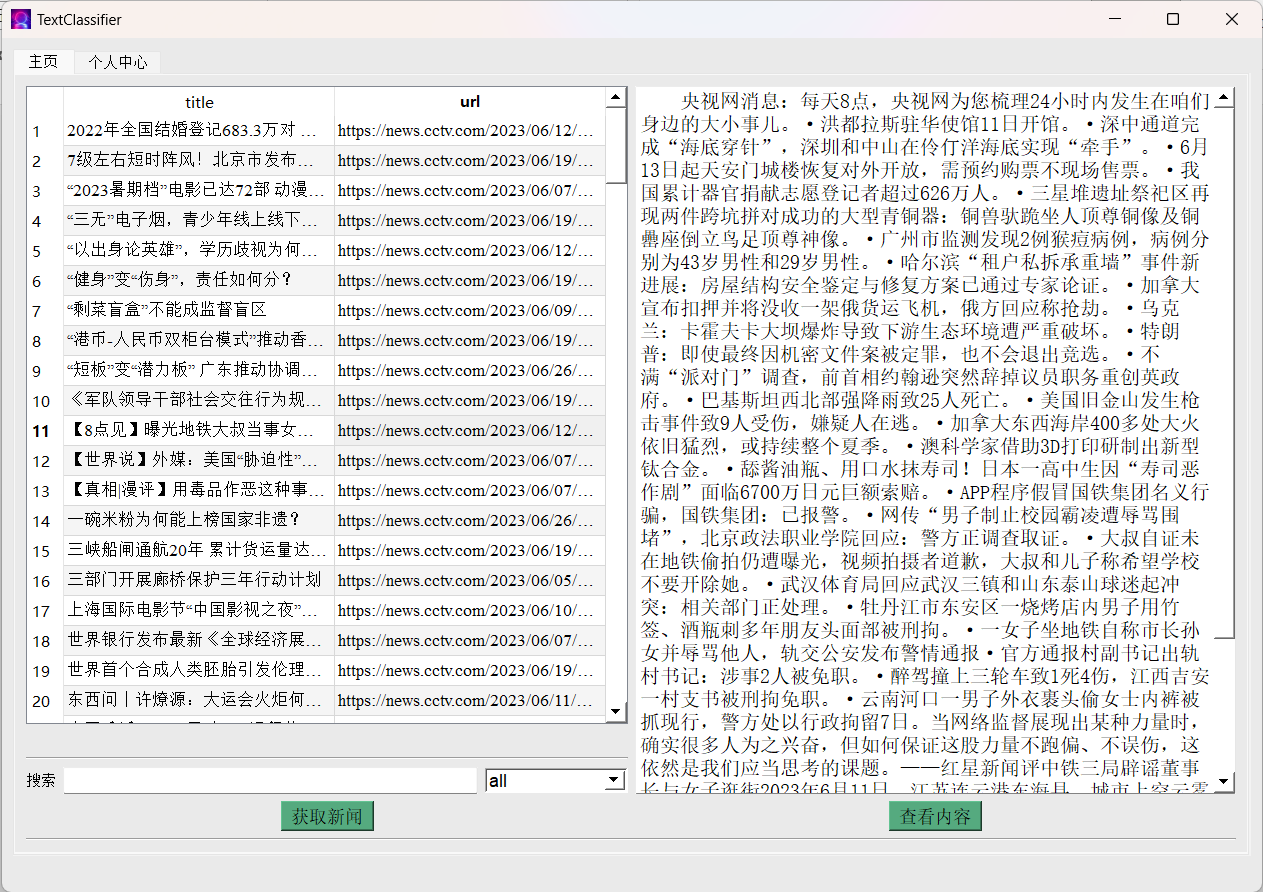


图 48 用户主页面（2）

# 7 总结

在如今的网络信息时代，网络数据渗透在我们日常生活、工作和学习的方方面面，人们无时无刻地利用电子设备和网络生产数据，传播数据，其中文本数据占据大量比重。在互联网信息流动的过程中，文本数据具有在内容上复杂多样，在规模上数量庞大的特性，其中最具有代表意义的网络新闻是人们获取社会信息，表述观点诉求的有效途径。然而如何从海量的网络新闻中精准地筛选出满足用户需求的新闻信息，仍然是值得关注的热点问题。因此，在当今移动互联网日益成熟的大环境下，针对用户多样化的需求，对海量新闻文本进行有效的分类和管理具有非常重要的研究意义。

本系统实现了基本的新闻挖掘和分类功能，对于更加个性化的功能，还有待后续改进。

# 参考文献

1. 徐建飞,吴跃成.基于BERT-BiLSTM-CNN模型的新闻文本分类研究[J].软件工程,2023,26(06):11-15.DOI:10.19644/j.cnki.issn2096-1472.2023.006.003.
2. 王乾,曾诚,何鹏,张海丰,余新言.基于Ro BERTa-RCNN和注意力池化的新闻主题文本分类[J/OL].郑州大学学报(理学版):1-8[2023-06-26].DOI:10.13705/j.issn.1671-6841.2022168.
3. 李海艳,王文东,江丛君.机器学习下的新冠疫情新闻文本分类[J].延安大学学报(自然科学版),2023,42(01):79-82.DOI:10.13876/J.cnki.ydnse.220054.
4. 郑创伟,王泳,邢谷涛,谢志成,陈义飞.基于深度神经网络的中文新闻文本分类方法[J].中国传媒科技,2023(03):147-151.DOI:10.19483/j.cnki.11-4653/n.2023.03.033.
5. 郝婷,王薇.融合Bert和BiLSTM的中文短文本分类研究[J].软件工程,2023,26(03):58-62.DOI:10.19644/j.cnki.issn2096-1472.2023.003.012.
6. 叶榕,邵剑飞,张小为,邵建龙.基于BERT-CNN的新闻文本分类的知识蒸馏方法研究[J].电子技术应用,2023,49(01):8-13.DOI:10.16157/j.issn.0258-7998.223094.
7. 阮进军,杨萍.基于Att-CN-BiLSTM模型的中文新闻文本分类[J].通化师范学院学报,2022,43(12):65-70.DOI:10.13877/j.cnki.cn22-1284.2022.12.012.
8. 周鹏. 基于词嵌入的文本分类及新闻推荐算法研究[D].南京邮电大学,2022.DOI:10.27251/d.cnki.gnjdc.2022.000885.
9. 高兴东. 基于深度学习的中文新闻标题文本分类系统研究[D].南京邮电大学,2022.DOI:10.27251/d.cnki.gnjdc.2022.001370.
10. ]温伍正宏. 基于深度学习的新闻文本分类系统[D].南京邮电大学,2022.DOI:10.27251/d.cnki.gnjdc.2022.000726.
11. [1]廖运春,舒坚.基于加权Word2Vec和TextCNN的新闻文本分类[J].长江信息通信,2022,35(09):32-35.