**知识智能感知系统实训报告**

学院名称： 计算机学院

小组成员：王海、吕鑫、胡曦伟、马茼茼、谢欣旖

指导教师： 刘小明

时间：2019年1月16日

**目录**

[第1章、引言 3](#_Toc8380)

[1.1目的 3](#_Toc19849)

[1.2范围 4](#_Toc14537)

[1.3名词解释 4](#_Toc1537)

[1.4参考资料 5](#_Toc1444)

[第2章、项目背景及系统概述 6](#_Toc21580)

[2.1项目背景 6](#_Toc13645)

[2.2项目意义 6](#_Toc19017)

[2.3系统概述 7](#_Toc18056)

[2.3.1 项目概述 7](#_Toc13988)

[2.3.2 项目创新点 7](#_Toc13434)

[2.3.3 项目技术 8](#_Toc6761)

[2.3.4 系统设计 8](#_Toc11381)

[2.3.5 系统开发环境 8](#_Toc17466)

[2.3.6 小组分工 9](#_Toc1760)

[2.3.7 系统整体架构图 10](#_Toc1535)

[第3章、产品特点 10](#_Toc17564)

[3.1产品定位 10](#_Toc24908)

[3.2技术特点 11](#_Toc7253)

[3.2.1 先进性 11](#_Toc10200)

[3.2.2 可扩展性 11](#_Toc16293)

[3.2.3 跨平台性 11](#_Toc14513)

[3.2.4 易用性 11](#_Toc32234)

[3.3系统特点 11](#_Toc23365)

[第4章 ElasticSearch检索 12](#_Toc20247)

[4.1 Elasticsearch简介 12](#_Toc9246)

[4.2 Elasticsearch优点 12](#_Toc27012)

[4.3开发环境 13](#_Toc2579)

[4.4 Elasticsearch的增删改查内部实现原理 13](#_Toc23440)

[4.5 Elasticsearch保存的数据展示 14](#_Toc32081)

[4.6 Elasticsearch成果及总结 15](#_Toc1623)

[第5章 Lda在文本主题分类中的应用 15](#_Toc19863)

[5.1．学习统计基本框架 15](#_Toc2381)

[5.2Lda原理 16](#_Toc14459)

[5.2.1 LDA贝叶斯模型 16](#_Toc32758)

[5.2.2 二项分布与Beta分布 17](#_Toc30372)

[5.2.3 多项分布与Dirichlet 分布 20](#_Toc1704)

[5.3．Lda实验 22](#_Toc10269)

[5.3.1实验目的 22](#_Toc1580)

[5.3.2实验要求 23](#_Toc16360)

[5.3.4任务分配 23](#_Toc1384)

[5.3.5实验框架 24](#_Toc31114)

[5.3.6模块运行截图 25](#_Toc4161)

[5.4最终实验结果 26](#_Toc31742)

[5.4.1主题分布 26](#_Toc3338)

[5.4.2提取摘要 27](#_Toc9811)

[5.5 LDA算法成果及总结 27](#_Toc22939)

[第6章 svm在文本主题分类中的应用 28](#_Toc21766)

[6.1 学习统计基本框架 28](#_Toc10994)

[6.1.1核心思想 28](#_Toc30936)

[6.1.2核函数分析 28](#_Toc26517)

[6.1.3分类器评价 29](#_Toc31683)

[6.2 SVM实验 30](#_Toc8443)

[6.2.1 实验目的 30](#_Toc17207)

[6.2.2 实验要求 31](#_Toc16712)

[6.2.3 任务分配 31](#_Toc2423)

[6.2.4实验框架 31](#_Toc22451)

[6.2.5.模块运行截图 33](#_Toc28375)

[6.3 实验结果 35](#_Toc336)

[6.4 SVM实验成果及总结 36](#_Toc27108)

[第7章 软件测试 36](#_Toc6432)

[第8章、总结 38](#_Toc13749)

# 

# 第1章、引言

## 1.1目的

本文档适用于软件工程实训平台开发小组成员，报告融合了软件工程规范要求的项目背景、需求分析、总体设计、详细设计等文档，并附有主要功能的详细设计说明。

根据实训平台项目开发需求与实训管理要求，编写项目软件需求规格书。本文档的目的是确定系统开发的范围；反映出系统的结构，为开发人员、维护人员、需求人员间提供共同的协议而创立基础，对软件功能的实现作使命描述，作为软件人员进行设计和编码的基础；作为实训小组成员的共同文档，为各小组成员完成任务提供依据；确定系统测试及验收内容。

## 1.2范围

本报告用于描述知识智能感知系统，包括整个系统需要达到的目标，需求分析，概要设计，详细设计，测试结果以及最终的成果。反映出系统的结构、思想，为维护人员、需求人员提供一个可供参考的说明，并影响系统的需求分析、设计以及开发工作。

## 1.3名词解释

表格1.1名词解释

|  |  |
| --- | --- |
| 术语名称 | 描述 |
| 情感分析 | 情感分析，即对一篇文章的情感的进行分析，分析出文章是正面情感还是负面情感。 |
| 主题分布 | 主题发布，即对一篇文章进行全面分析，得出文章的所有主题和其分布概率，然后计算出文章的最主要的主题，呈现给大家。 |
| 关键句提取 | 关键句提取，可以分析断句，结合主题分布，关键词提取出最能表现文章主题的三句话，通过这三句话可以清楚地了解文章的大意。 |
| 关键词提取 | 关键词提取，提取文章中主要的词汇，展示出效果，并用于关键句提取 |
| 选择性搜索 | 对搜索的内容可以限定时间间隔，是否相关等选择，通过选择找到对应内容 |

## 1.4参考资料

1. 软件工程实训报告模板；
2. 北京邮电大学课题实验报告；
3. 中华人民共和国国家标准，GB9385-88，计算机软件需求说明

编制指南，中华人民共和国电子工业部，1988-04-06批准。

# 第2章、项目背景及系统概述

## 2.1项目背景

随着网络技术的不断发展，数字化、信息化以及网络化越来越成为 21 世纪的主要特征，其中人工智能的发展成为以后智能化社会的基石，知识智能感知系统希望通过分析大量的文章数据并进行处理，首先对当你想要了解的内容进行查找，显示结果，对结果处理对现阶段发生的事件进行分类，分析，提取主题，让人们在最快的时间了解自己想要的东西。为以后人工智能语言系统的发展做出一定的小小贡献。

## 2.2项目意义

随着网络快速发展，信息量指数化上升以及人民生活节奏加快，愈来愈多的人对信息获取的内容量，速度，准确度有了极高的要求，

人们在一天繁忙的工作结束后往往没有太多的时间去休息娱乐，所以获取信息快速准确成了一种趋势。一般的搜索既不准确也不快速，同时还有大量广告惨杂其中，有些网站为了长度和点击量，其中文章字数繁多，话语啰嗦，导致简单的信息无法让用户直接提取并了解，想要了解文章主题和灵魂只能读完全篇文章。以上这些问题都造成了大量时间的浪费和对用户显示的不友好性，人们往往用了太多时间去搜索，排查有用的信息。而真正想要了解的却少之又少。

所以一个对信息高效快速检索同时又兼备对信息智能提取和分类的感知系统的开发是大势所趋，必不可少的。

## 2.3系统概述

### 2.3.1 项目概述

为了实现高效快速准确的搜索，我们通过ElasticSearch搜索服务器进行分布式的全文搜索引擎。并进行关键词和时间范围等的大数据搜索。

本系统采用 SpringBoot 作为系统框架，并且采用了内置Tomact作为服务器，在进行开发时，为了提高开发效率和系统后期可维护性，这里采用了分布式开发模式。前后端分离进行跨域请求完成交互，本系统在进行设计和开发时，在显示层采用了最先进的vue.js框架实现，后端用SpringBoot框架完成开发环境搭建，功能实现，并通过python的flask接口封装了SVM、LDA、TextRank的自然语言处理方法。

总的来说，本系统主要功能是实现了对大数据的搜索、对文章情感的分析、同时实现对文章主题的分析关键词关键句的提取，基本上能满足了人们对信息快速准确的了解的需求。

### 2.3.2 项目创新点

* 应用创新：第一次把SVM和lda等模型结合运算，完成了文章全面分析；
* 技术创新：本系统结合了大数据的相关分析，使用了非关系型数据库ElasticSearch和大量主题模型方法；

### 2.3.3 项目技术

* 非关系型数据库ElasticSearch的使用。
* 支持大数据分析处理。
* 搜索速度快速准确率提高，可以做到模糊查找。
* Springboot建立的微服务，框架简单方便。
* 实现了lda结合textrank算法，可以提取文章中权重最高的三句话组成文章中心思想。

### 2.3.4 系统设计

本系统主有三重功能，功能分别是：

1. 选择性查找：可以对要查找的事件进行查找情况的选择
2. 文章的分析：对一篇文章进行分析包括分析文章感情，分析文章的主题

3、文章的提取：对文章的内容进行提取，包括提起文章的关键词，关键句。

### 2.3.5 系统开发环境

#### 硬件环境

表格 2.1硬件环境

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **名 称** | **类 型** | **用 途** |
| 1 | Centos7 | Web服务器、数据库服务器，elasticsearch数据库 | Web服务器、数据库服务器 |
| 2 | 华硕电脑 | 客户端 | 客户端 |

#### 支持软件

表格2.2支持软件一览表

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **名 称** | **类 型** | **用 途** |
| 1 | IDEA2018.1 | 应用软件 | 软件开发环境 |
| 2 | Pycharm2018.1 | 应用软件 | 算法运行环境 |
| 3 | WebStorm2018.2 | 应用软件 | 前端开发环境 |
| 3 | Axure RP8 | 应用软件 | 界面设计 |
| 4 | Tomcat 7.0 | Web服务器 | 网站架设环境 |
| 5 | MySQL5.6 | 数据库管理软件 | 管理系统数据 |
| 6 | Windows 7 及以上 | 操作系统 | 系统运行环境 |
| 7 | IE8.0以上/FireFox/Chrome | 浏览器 | 用户使用环境 |

### 2.3.6 小组分工

表格2.3项目分工表

|  |  |  |
| --- | --- | --- |
| 组员名称 | 担任角色 | 主要分工 |
| 王海 | 项目组长、算法工程师、配置管理员、编码人员、 | 1. 负责小组内部日常管理工作 2. 组织其他组员完成需求分析 3. 与其他组员一同完成系统设计 4. 完成系统架构与svm算法模块 5. 完成系统测试和系统文档报告 6. 数据采集 |
| 吕鑫 | 算法工程师、需求分析员、编码人员、 | 1、协助组长完成需求分析  2、完成elasticsearch模块与svm算法模块  3、协助完成系统测试  4、协助完成系统文档报告  5、数据采集 |
| 马茼茼 | 算法工程师、编码人员 | 1. 协助组长完成需求分析 2. 完成LDA算法模块与flask接口模块 3. 协助完成系统测试 4. 协助完成系统文档报告 5. 数据采集 |
| 胡曦伟 | 算法工程师、编码人员、 | 1、协助组长完成需求分析  2、完成LDA算法模块  3、协助完成系统测试  4、协助完成系统文档报告  5、数据采集 |
| 谢欣旖 | 需求分析员、界面设计员、 | 1、协助组长完成需求分析  2、完成前端页面展示模块  3、协助完成系统测试  4、协助完成系统文档报告  5、数据采集 |

### 2.3.7 系统整体架构图

知识智能感知系统是一个采用springboot的技术，包括选择性查找、文章主题摘要、文章关键词关键句摘要等功能。elasticsearch是系统数据库，起到支持大数据搜索的作用，用户首先进行条件性选择，界面分页显示全部显示结果，点击文章，下方显示文章，并给出要进行的操作；

# 第3章、产品特点

## 3.1产品定位

该系统的目标客户是所有网络用户，为网民朋友提供一套全面的智能搜索分类提取系统，让网民朋友能通过条件限定式搜索了解自己想要的信息并通过提取在短时间快速了解。由于系统采用的模块化设计方法，客户可以选择自己读取文章还是进行分析，分析结果数据图形化展示。

## 3.2技术特点

### 3.2.1 先进性

系统遵循技术发展主流的国际标准，以SpringBoot体系为骨架,elasticsearch6.2.4版本数据，后端采用建maven1.4工程搭建后端项目，前端采用了Vue+ElementUI,flask等最新技术。

### 3.2.2 可扩展性

模块化技术，分布式开发，前后端分离，即保证了对各种技术的兼容性, 提高了技术合作的可选择性、灵活性。每个模块之间的开发相互独立，耦合性大大降低，扩展要求只要单个扩展。

### 3.2.3 跨平台性

在设计上采用先进的springboot技术，使系统具有跨平台的特性。

### 3.2.4 易用性

### 用户界面全部采用可视化界面。只需简单的点击，并在相关的输入框中输入相关信息即可完成，以最大程度的简化手续，方便使用者。同时数据展示采用图形化格式化的形式，清晰明了，比较感突出。

## 3.3系统特点

**1.**基于springboot技术实现，具有良好的伸缩性、开放性和可扩展性。

**2.**提供了模块化的代码编写方式，解决了软件耦合性问题。

**3.**采用非关系型数据库数据库elasticsearch，建立倒排索引，大数据搜索准确快速 。

**4.**采用flask接口实现python算法与springboot进行数据交互。

**5.**前后端分离式开发模式，可以让软件开发快捷有序，维护管理轻易快捷。

# **ElasticSearch检索**

## 4.1 Elasticsearch简介

ElasticSearch是一个基于Lucene的搜索服务器。它提供了一个分布式多用户能力的全文搜索引擎，基于RESTful web接口。Elasticsearch是用Java开发的，并作为Apache许可条款下的开放源码发布，是当前流行的企业级搜索引擎。设计用于[云计算](https://baike.baidu.com/item/%E4%BA%91%E8%AE%A1%E7%AE%97/9969353" \t "https://baike.baidu.com/item/elasticsearch/_blank)中，能够达到实时搜索，稳定，可靠，快速，安装使用方便。

## 4.2 Elasticsearch优点

Elasticsearch刚好可以满足我们的以下需求:

1. 分布式的搜索引擎和数据分析引擎
2. 全文检索，结构化检索，数据分析
3. 对海量数据进行近实时的处理

## **4.3开发环境**

1. IDEA2018.1
2. ElasticSearch 6.2.4

## 4.4 Elasticsearch的增删改查内部实现原理

添加、创建：

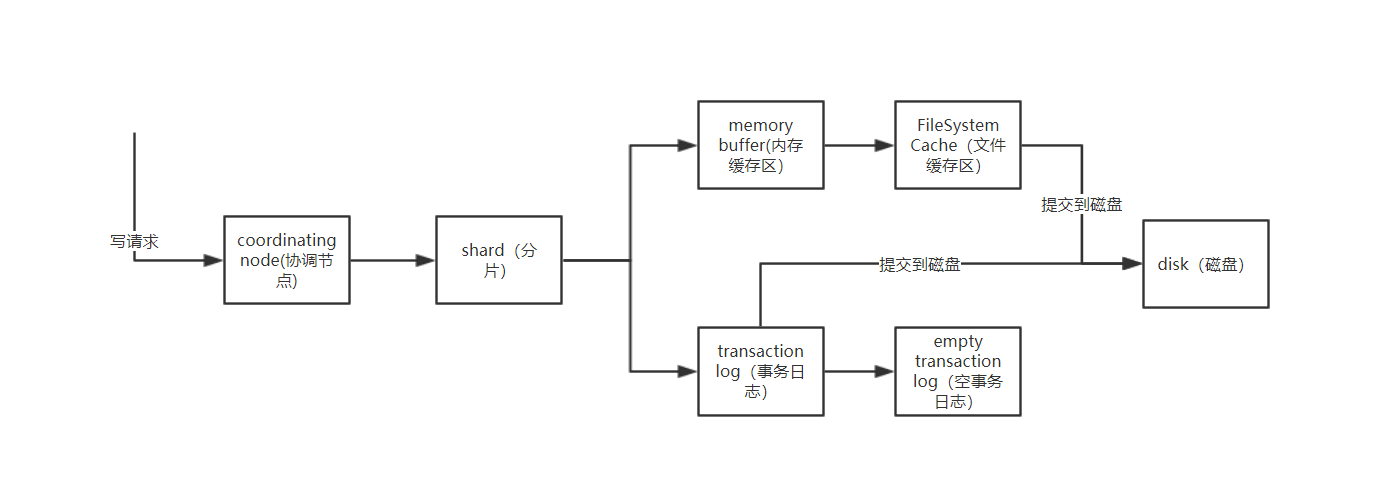
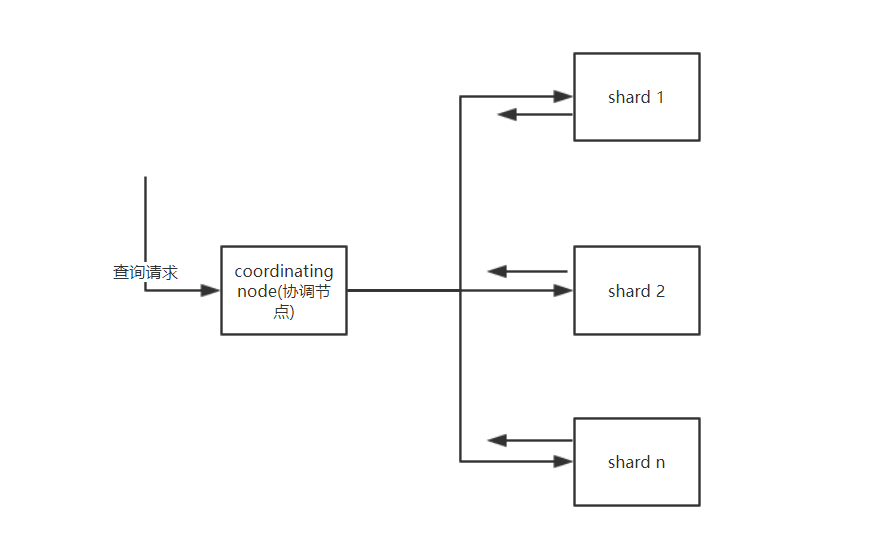


图4.4-1 ElasticSearch中crud原理图

查询、更新：

4.4-2数据查询原理图

## 4.5 Elasticsearch保存的数据展示

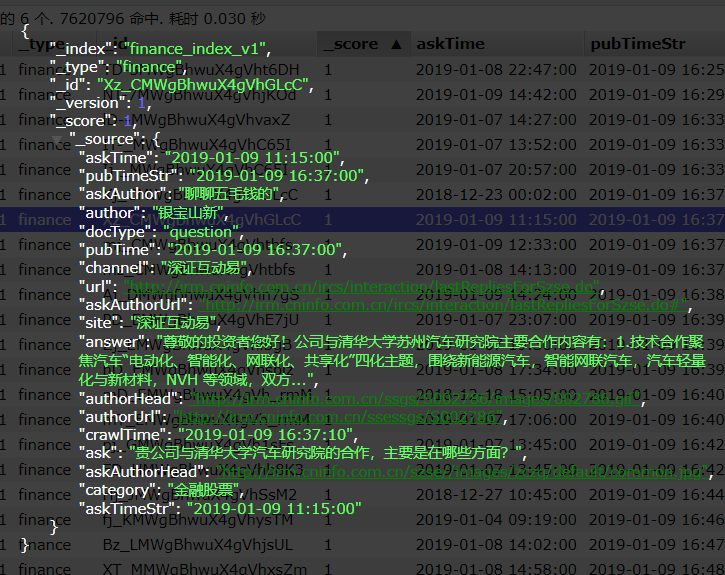


图4.5 ElasticSearch数据库保存数据图

## 4.6 Elasticsearch成果及总结

实现了elasticsearch在分布式es数据库上的条件查询查询

通过学习Elasticsearch，我们了解了倒排索引的搜索方法对于搜索速度和准确性的提高，了解了搜索的最新方法，从一开始学习Elasticsearch下载安装到学习其基本语言，到Elasticsearch成功配置引入springboot中，一路上感觉我们付出了辛苦同时收获了知识。

在项目中一开始时，我们使用的单个match查询或者term查询，发现只能做到简单查询，功能实现单一简单。后我们使用了multimatch和query实现了聚合查询和组合查询实现有多种功能的可选择性的查询。在这期间的不断改进和步步求精，让我们对Elasticsearch有了及其深刻的了解。

# 第5章 Lda在文本主题分类中的应用

## 5.1学习统计基本框架

学习lda框架大致分为三个基本知识：

1. 学习贝叶斯分布：lda实际为三层贝叶斯嵌套，其中边缘概率

p(wm→|α⃗ ,β⃗ )

=∫θm∫Φ∫zm→∏nNmp(wm,n|ϕzm,n→)p(zm,n|θm→)p(θm→|α⃗ )p(Φ|β⃗）

（2）多项式分布：主题和词汇的分布就是多项式分布！因为一个主题里面不同词汇出现的概率不同，如果只有1个词汇，那么它就是二项分布，但是词汇不可能只有1个，所以它理所当然就是符合多项式分布

（3）狄利克雷函数：

多项式分布和Dirichlet分布是一对共轭分布，共轭分布有什么好处呢？好处在于计算后验概率有极大的便利！说到底是LDA看中它计算方便。增加了先验概率分布，那么在确定文章与主题分布还有主题与词汇分布的时候，就由先验概率分布先随机生成确定多项式分布的参数。

## 5.2 LDA原理

### 5.2.1 LDA贝叶斯模型

　　LDA是基于贝叶斯模型的，涉及到贝叶斯模型离不开“先验分布”，“数据（似然）”和"后验分布"三块。在朴素贝叶斯算法原理小结中我们也已经讲到了这套贝叶斯理论。在贝叶斯学派这里：

先验分布 + 数据（似然）= 后验分布

　　这点其实很好理解，因为这符合我们人的思维方式，比如你对好人和坏人的认知，先验分布为：100个好人和100个的坏人，即你认为好人坏人各占一半，现在你被2个好人（数据）帮助了和1个坏人骗了，于是你得到了新的后验分布为：102个好人和101个的坏人。现在你的后验分布里面认为好人比坏人多了。这个后验分布接着又变成你的新的先验分布，当你被1个好人（数据）帮助了和3个坏人（数据）骗了后，你又更新了你的后验分布为：103个好人和104个的坏人。依次继续更新下去。

### 5.2.2 二项分布与Beta分布

　　对于上一节的贝叶斯模型和认知过程，假如用数学和概率的方式该如何表达呢？

　　对于我们的数据（似然），这个好办，用一个二项分布就可以搞定，即对于二项分布：

Binom(k|n,p)=(nk)pk(1−p)n−k

Binom(k|n,p)=(nk)pk(1−p)n−k

　　其中p我们可以理解为好人的概率，k为好人的个数，n为好人坏人的总数。

　　虽然数据(似然)很好理解，但是对于先验分布，我们就要费一番脑筋了，为什么呢？因为我们希望这个先验分布和数据（似然）对应的二项分布集合后，得到的后验分布在后面还可以作为先验分布！就像上面例子里的“102个好人和101个的坏人”，它是前面一次贝叶斯推荐的后验分布，又是后一次贝叶斯推荐的先验分布。也即是说，我们希望先验分布和后验分布的形式应该是一样的，这样的分布我们一般叫共轭分布。在我们的例子里，我们希望找到和二项分布共轭的分布。

　　　　和二项分布共轭的分布其实就是Beta分布。Beta分布的表达式为：

Beta(p|α,β)=Γ(α+β)Γ(α)Γ(β)pα−1(1−p)β−1

Beta(p|α,β)=Γ(α+β)Γ(α)Γ(β)pα−1(1−p)β−1

　　　　其中ΓΓ是Gamma函数，满足Γ(x)=(x−1)!Γ(x)=(x−1)!

仔细观察Beta分布和二项分布，可以发现两者的密度函数很相似，区别仅仅在前面的归一化的阶乘项。那么它如何做到先验分布和后验分布的形式一样呢？后验分布P(p|n,k,α,β)P(p|n,k,α,β)推导如下：

P(p|n,k,α,β)∝P(k|n,p)P(p|α,β)

=P(k|n,p)P(p|α,β)

=Binom(k|n,p)Beta(p|α,β)

=(nk)pk(1−p)n−k×Γ(α+β)Γ(α)Γ(β)pα−1(1−p)β−1

∝pk+α−1(1−p)n−k+β−1

将上面最后的式子归一化以后，得到我们的后验概率为：

P(p|n,k,α,β)=Γ(α+β+n)Γ(α+k)Γ(β+n−k)pk+α−1(1−p)n−k+β−1

P(p|n,k,α,β)=Γ(α+β+n)Γ(α+k)Γ(β+n−k)pk+α−1(1−p)n−k+β−1

可见我们的后验分布的确是Beta分布，而且我们发现：

Beta(p|α,β)+BinomCount(k,n−k)=Beta(p|α+k,β+n−k)

Beta(p|α,β)+BinomCount(k,n−k)=Beta(p|α+k,β+n−k)

　 这个式子完全符合我们在上一节好人坏人例子里的情况，我们的认知会把数据里的好人坏人数分别加到我们的先验分布上，得到后验分布。

　　我们在来看看Beta分布Beta(p|α,β)Beta(p|α,β)的期望:

E(Beta(p|α,β))

=∫10tBeta(p|α,β)dt

=∫10tΓ(α+β)Γ(α)Γ(β)tα−1(1−t)β−1dt

=∫10Γ(α+β)Γ(α)Γ(β)tα(1−t)β−1dt

E(Beta(p|α,β))

=∫01tBeta(p|α,β)dt(7)

=∫01tΓ(α+β)Γ(α)Γ(β)tα−1(1−t)β−1dt(8)

=∫01Γ(α+β)Γ(α)Γ(β)tα(1−t)β−1dt

　　由于上式最右边的乘积对应Beta分布Beta(p|α+1,β)Beta(p|α+1,β),因此有：

∫10Γ(α+β+1)Γ(α+1)Γ(β)pα(1−p)β−1=1

∫01Γ(α+β+1)Γ(α+1)Γ(β)pα(1−p)β−1=1

　　这样我们的期望可以表达为：

E(Beta(p|α,β))=Γ(α+β)Γ(α)Γ(β)Γ(α+1)Γ(β)Γ(α+β+1)=αα+β

E(Beta(p|α,β))=Γ(α+β)Γ(α)Γ(β)Γ(α+1)Γ(β)Γ(α+β+1)=αα+β

　　　　这个结果也很符合我们的思维方式。

### 5.2.3 多项分布与Dirichlet 分布

　　现在我们回到上面好人坏人的问题，假如我们发现有第三类人，不好不坏的人，这时候我们如何用贝叶斯来表达这个模型分布呢？之前我们是二维分布，现在是三维分布。由于二维我们使用了Beta分布和二项分布来表达这个模型，则在三维时，以此类推，我们可以用三维的Beta分布来表达先验后验分布，三项的多项分布来表达数据（似然）。

　　三项的多项分布好表达，我们假设数据中的第一类有m1m1个好人，第二类有m2m2个坏人，第三类为m3=n−m1−m2m3=n−m1−m2个不好不坏的人,对应的概率分别为p1,p2,p3=1−p1−p2p1,p2,p3=1−p1−p2，则对应的多项分布为：

multi(m1,m2,m3|n,p1,p2,p3)=n!m1!m2!m3!pm11pm22pm33

multi(m1,m2,m3|n,p1,p2,p3)=n!m1!m2!m3!p1m1p2m2p3m3

　　那三维的Beta分布呢？超过二维的Beta分布我们一般称之为狄利克雷(以下称为Dirichlet )分布。也可以说Beta分布是Dirichlet 分布在二维时的特殊形式。从二维的Beta分布表达式，我们很容易写出三维的Dirichlet分布如下：

Dirichlet(p1,p2,p3|α1,α2,α3)

=Γ(α1+α2+α3)Γ(α1)Γ(α2)Γ(α3)pα1−11(p2)α2−1(p3)α3−1

Dirichlet(p1,p2,p3|α1,α2,α3)

=Γ(α1+α2+α3)Γ(α1)Γ(α2)Γ(α3)p1α1−1(p2)α2−1(p3)α3−1

同样的方法，我们可以写出4维，5维，。。。以及更高维的Dirichlet 分布的概率密度函数。为了简化表达式，我们用向量来表示概率和计数

这样多项分布可以表示为：

Dirichlet(p⃗ |α⃗ )Dirichlet(p→|α→)

而多项分布可以表示为：

multi(m⃗ |n,p⃗ )multi(m→|n,p→)。

　　一般意义上的K维Dirichlet 分布表达式为：

Dirichlet(p⃗ |α⃗ )=Γ(∑k=1Kαk)∏Kk=1Γ(αk)∏k=1Kpαk−1k

Dirichlet(p→|α→)=Γ(∑k=1Kαk)∏k=1KΓ(αk)∏k=1Kpkαk−1

　　而多项分布和Dirichlet 分布也满足共轭关系，这样我们可以得到和上一节类似的结论：

Dirichlet(p⃗ |α⃗ )+MultiCount(m⃗ )=Dirichlet(p⃗ |α⃗ +m⃗ )

Dirichlet(p→|α→)+MultiCount(m→)=Dirichlet(p→|α→+m→)

　　对于Dirichlet 分布的期望，也有和Beta分布类似的性质：

E(Dirichlet(p⃗ |α⃗ ))=(α1∑k=1Kαk,α2∑k=1Kαk,...,αK∑k=1Kαk)

## **5.3．LDA实验**

### 5.3.1实验目的

在LDA中，所有的文档共有同样的话题集，但是每个文档以不同的比例展示对应的话题。LDA的主要目标是自动发现一个文档集合中的话题。这些文档本身是可以观测到的，而话题的结构——话题、每个文档的话题分布和每个文档的每个词的话题赋值——是隐藏的（可称为hidden structure）。话题建模的核心计算问题就是使用观测到的文档来推断隐藏话题结构。这也可以看作是生成（generative）过程的逆过程——借助LDA算法可以得到话题的结构，需要指出的是，算法本身并不需要用到这些话题的信息，文档本身也没有使用话题或者关键字进行标注。这个隐藏结构最有可能产生现在可以观测到的文档集合。

话题模型方便的地方就是可以通过推断的隐藏结构来组成文档的主题结构。这样的信息对于信息检索，分类和语料研究提供了有力的支撑。所以可以这样说，话题模型给出了一种管理、组织和标注文本的大量集合的算法解答。

### 5.3.2实验要求

通过对lda的研究实现对文章句子的拆分，并结合字典来从大到小提取关键词及其概率，通过概率建立主题多项式分布和文章多项式分布，最终集成textrank方法实现主题句的抽取，摘要，并计算概率，通过人工测试，完成模型研究。

**5.3.3运行环境**

1. 服务部署Centos7
2. Python3.6.7

### 5.3.4任务分配

|  |  |
| --- | --- |
| 小组成员 | LDA任务分配 |
| 王海 | Lda算法模块搭建，实现完成词典主题分布，并实现分句功能 |
| 吕鑫 | Lda算法修改，完成分词计算概率，提取主题词汇分布概率矩阵 |
| 胡曦伟 | 集成textrank算法，实现从概率矩阵计算完成句子提取 |
| 马茼茼 | 实现从概率矩阵计算完成句子提取，并编写flask接口 |
| 谢欣旖 | 编写对应页面，实现通过接口与后端建立数据交互 |

### 5.3.5实验框架

### （LDA）主题模型是一个采用python手工编写，文本预处理一体化的可重用模型，目前包括主题词生成和摘要提取两个功能。系统后台通过对文档集的训练从而得到用户更易理解的文档“话题”。

Ngnix服务器

Pycharm

(python编译器)

WebStorm(界面设计)

主题词

摘要提取

操作系统

(windows,Linux）

图3.4-1 实验框架

### 5.3.6模块运行截图

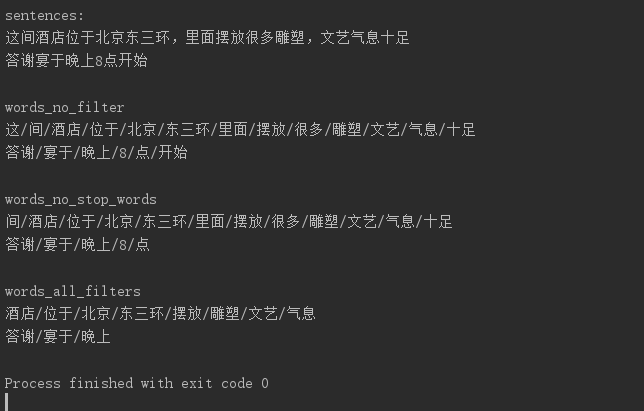


图5.3.6-1（文章分词）

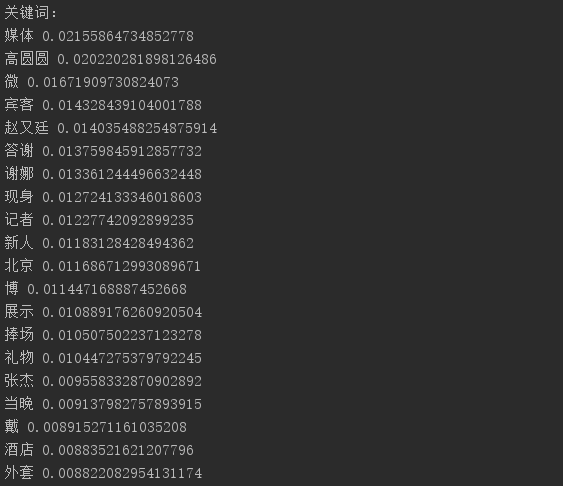


图5.3.6-2（主题词及分布概率）

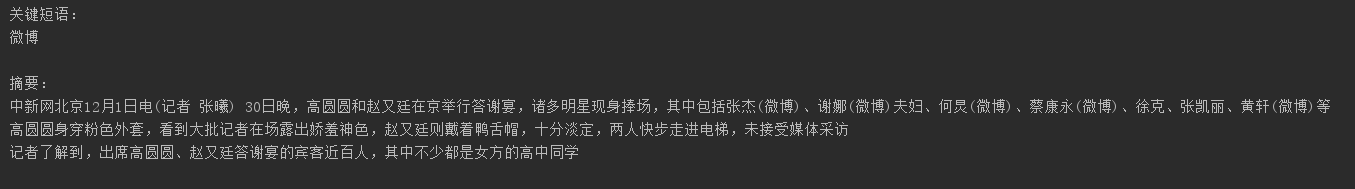


图5.3.6-3（关键句）

#### 5.3.7人工计算结果（模糊）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 文章字数 | 200 | 800 | 1500 | 2000 |
| 平均准确率 | 80.347 | 91.725 | 94.0625 | 94.5233 |
| 文章数目 | 5 | 5 | 5 | 5 |

## **5.4最终实验结果**

### 5.4.1主题分布



图5.4.1（主题词）

### 5.4.2提取摘要

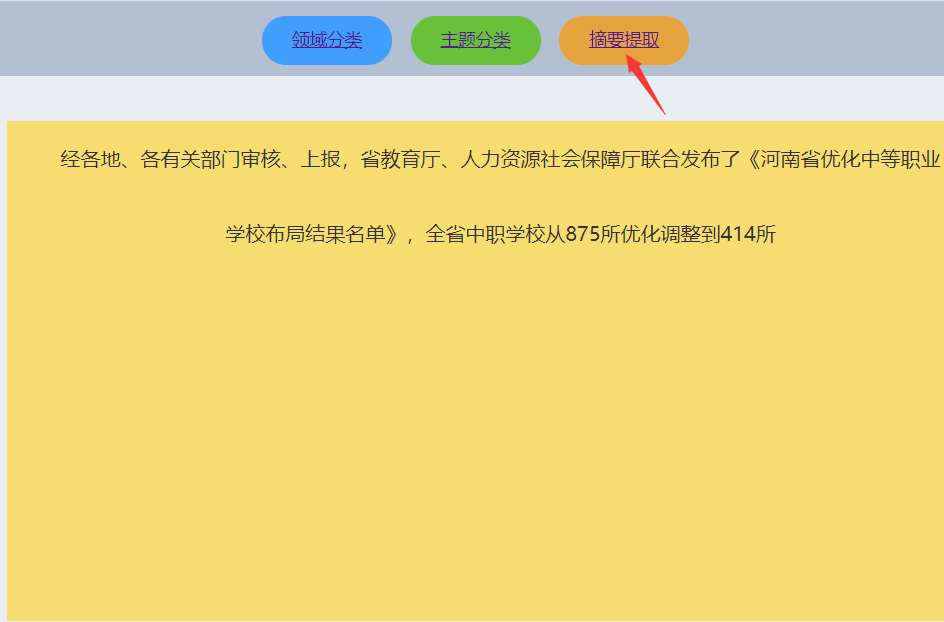


图5.4.2（关键句）

## **5.5 LDA算法成果及总结**

对于lda算法的学习研究，在一开始对于大量没有接触过的数学理论和算法，显得有些吃力，通过深入学习不仅可以了解算法还可以提高数学逻辑和代码分析能力，对我们有很大的提高，对于lda还是存在一些问题和改进情况。  
（1）改进采样方法提高速度，所以文章如果短小择准确率不高，对于大数据进行计算择准确率提高，想要即在保持速度的同时提高准确率。  
（2）只能做到摘要是提取，对于主题的把控只能对于权重最高的句子，主题无法做到生成式摘要。

# **第6章 svm在文本主题分类中的应用**

## 6.1 学习统计基本框架

### 6.1.1核心思想

支持向量机的主要思想是利用核函数把低维非线性空间可分向量集映射到高维特征空间，然后在高维特征空间中构造最优的分类超平面，使此最优超平面逼近分类函数，从而使其具有较好的推广能力。

### 6.1.2核函数分析

SVM通过核函数将低维不可分的空间特征映射到高维线性可分的特征空间中，然后再进行线性分类。因此，核函数的选择往往是决定SVM非线性分类性能的关键。SVM是由训练样本集和核函数完全描述的，采用不同的核函数k(xi,xj)可以构造输入空间中不同类型的非线性决策面，也可导致训练支持向量机的算法不同。在实际问题中，通常是直接给出核函数。目前研究最多、最常用的核函数有:

(1)线性核函数 k(xi,x)=(xi,x)

(2)多项式核函数 k(xi,x)=((xi,x)+1)q q是参数

(3)Sigmoid核函数 k(xi,x)=ranh(v(xi,x)+c)

(4)径向基核函数 K(x,xi)=exp{-(｜x-xi｜2)/ δ2}

二次规划求解中。由于核函数的重要性，如何去构造、选择核函数及参数成为SVM分类器训练的重中之重。核函数一旦被确定，那么对于SVM分类器而言就只有一个参数可以调整，那就是惩罚函数C。确定参数C一般有三种方法。(1)通过求解一个无约束问题来寻求其最优值;(2)采用类似于K折交叉确认的方式，适当选定所有参数各自的一个选值区间，分别在各自区间内离散地配对取值，然后在测试集上比较相应的K折交叉确认误差，达到最小误差时对应的参数值即为最优的参数选择，又称网格筛选法;(3)用遗传算法来求最优参数值。这些方法中第二种方法比较常用。

### 6.1.3分类器评价

分类器的评价是分类器设计的最后阶段，是数据挖掘中最重要的步骤之一。它能决定是否采用某种学习方法，还可以用于指导分类器的生成。 某一分类方法在训练集上表现出很好的分类能力，但这不代表这一方法在新的数据集合中任然有如此好的表现，所以在分类器评价时需要一种能够预测该分类方法在实践中的表现好坏非评价手段。 在做分类器评价时，我们需要一组没有参与分类器建立的数据集作为测试集合(test set)。如果训练数据很充足，可以取一个较大的样本集合来做训练，取另一个不同于训练集合的大样本数据作为测试集合;但当数据样本不够大的时候，如何利用有限的数据，这就比较的难了。

对于有限的样本集合构建训练集合测试集的主要方法有:交叉验证，受试者工作曲线。交叉验证是统计分析中评估分类模型应用水平的一个独立验证数据集预测效果的技术。应用交叉验证的方法可以估计分类器在实践中的预测分类的精确度。一般交叉验证是将原有的数据集合分成两部分数据子集来评价和比较不同分类方法的分类效能。这两部分数据一部分是用于训练器学习的训练集合，另一部分则作为验证的验证集合。常用的交叉验证方法有:保持验证、K折交叉验证、留一法交叉验证。受试者工作特征曲线(ROC )是显示分类模型中敏感性和特异性之间折中的一个可视化工具。通过对ROC曲线的分析可以得到最优分类器。

正确率=(TP+TN)/(TP+FN+FP+TN)x 100%

敏感度(TPR)=TP/(TP+FP)x 100%

特异度=TN/(FN+TN)x 100%=1-FPR

在绘制ROC曲线时，以FPR为横轴，以TPR为纵轴，以曲线下的面积AUC

来衡量分类效能的好坏，当AUC在0.5 ~ 0.7时有较低准确性，AUC在0.7~0.9

时有一定准确性，AUC在0.9以上时有较高准确性。AUC=0.5时，说明分类方法

完全不起作用，无使用价值。

## **6.2 SVM实验**

### 6.2.1 实验目的

1. 熟悉爬虫的使用，可以利用网络爬虫抓取所需的网络语料
2. 熟悉中文分词软件，可以熟练使用接口完成分词任务
3. 熟悉文本分类必要的预处理步骤，并运用到实验实践中
4. 熟悉特征提取方法了解SVM机器学习方法，可以运用开源工具完成文本分类过程

### 6.2.2 实验要求

通过对SVM算法的研究实现对文章的情感分析，首先对所有新闻文件使用jieba分词进行分词、去除停用词。把处理得到的数据生成数据字典，并转化成的numpy矩阵，根据数据字典，将每篇训练集新闻文档和测试集新闻文档转换成特征向量。

### 6.2.3 任务分配

|  |  |
| --- | --- |
| 小组成员 | SVM任务分配 |
| 王海 | 数据收集，文章分词，去除停用次 |
| 吕鑫 | 提取tf-idf矩阵，并选取一个好的算法对矩阵进行降维 |
| 胡曦伟 | 用sklearn进行对模型的训练，对测试集的测试 |
| 马茼茼 | 拿数据进行分类，并编写相应的flask接口 |
| 谢欣旖 | 编写对应页面，实现通过接口与后端建立数据交互，解决跨域请求问题 |

### 6.2.4实验框架

SVM是一个采用python手工编写，文本预处理一体化的可重用模型，目前包括情感分析。系统后台通过对文档集的训练从而得到用户更易理解的文档“话题”。

基于CHI特征提取的实验流程图：

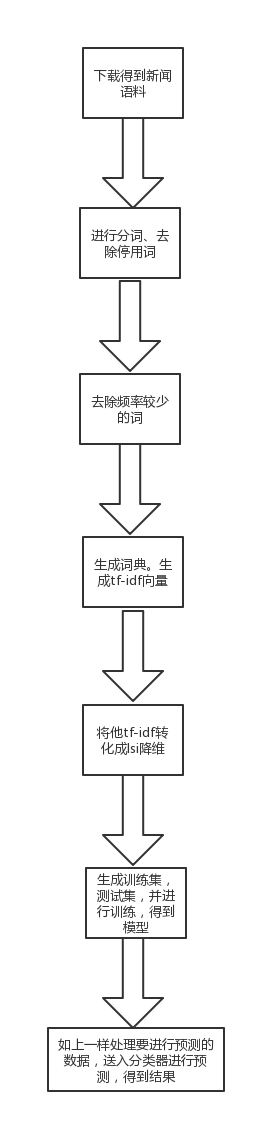


图6.2.4-1 提取特征实验流程图

### 6.2.5.模块运行截图

#### 6.2.5.1截图

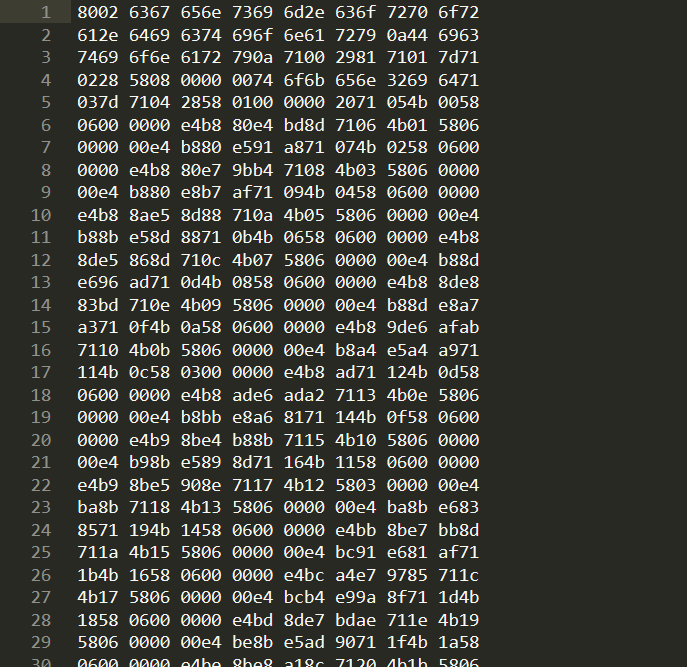
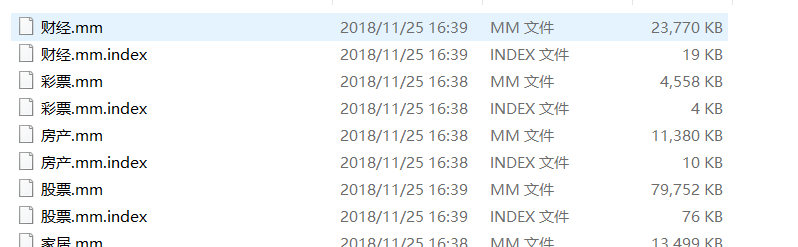


图5.1-1 tf-idf矩阵



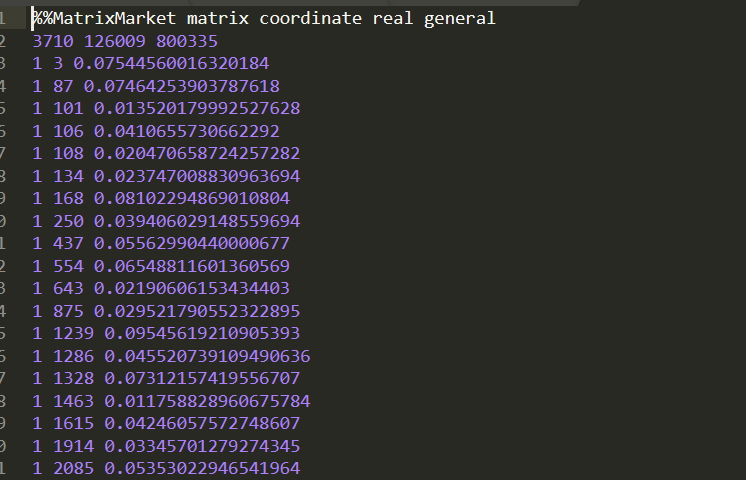


图5.1-2（mm文件）

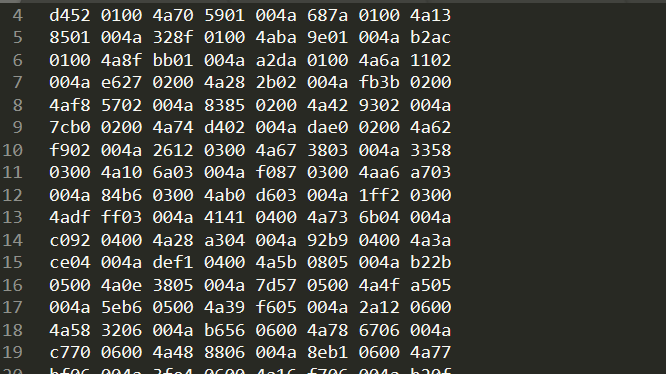


图5.1-3（mm.index文件）

#### 6.2.5.2人工计算结果（模糊）

（1）得到特征向量文件之后（未经过归一化处理），直接使用lib svm的默认参数，进行模型训练，得到的模型用于测试。

Precision为：57.1838%（79780/139515）

可以看出结果比较差。

（2）为了解决上述问题，使用svm的scale参数对训练集和测试集进行归一化处理把数字变为小数，下限为0，上限为1。再次使用linear svm默认参数训练模型，进行测试，可以得到：

Precision为：75.8535%（105827/139515）

由此可以推断出，进行归一化处理，是必要的一步。它可以大大的提高分类效果。

（3） 为了更好的提高分类效果，对模型进行调参，在调参的过程中，我们是对每个参数都进行了测试，最后得到一个相对较好的参数。

## **6.3 实验结果**

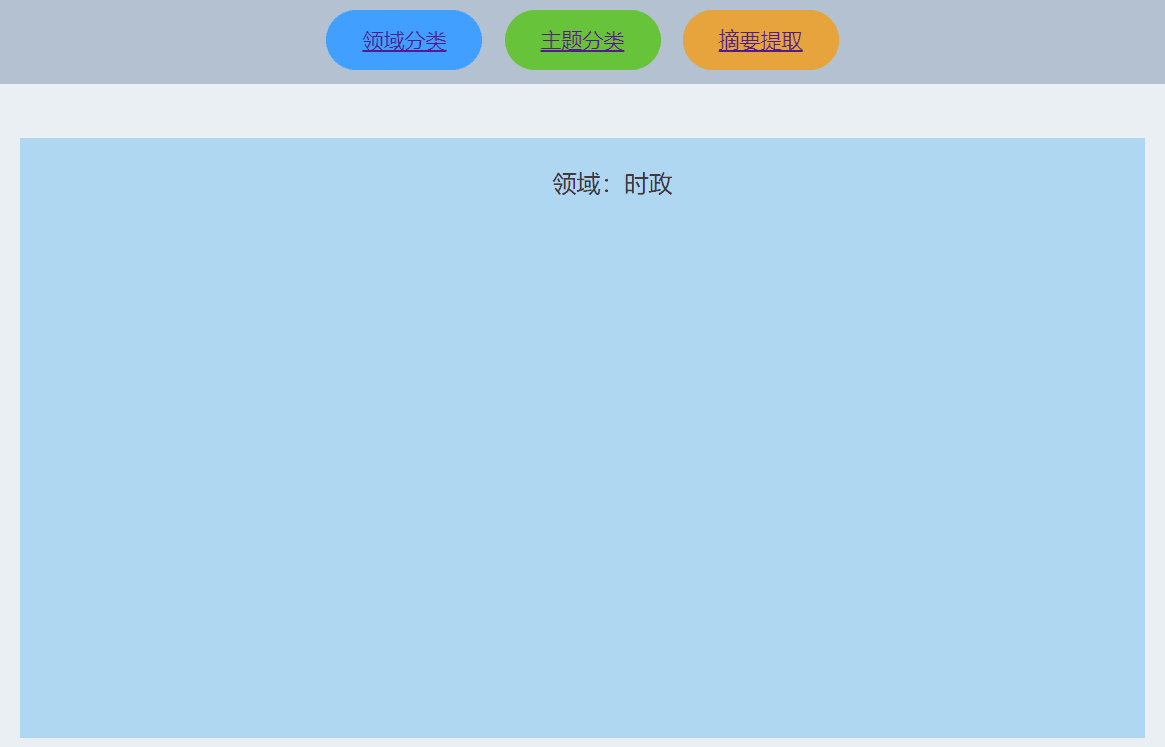


图6.3 领域分类实验结果图

## 6.4 SVM实验成果及总结

经过此次试验，学习了文本分类的整体流程，熟悉了机器学习中SVM的原理与使用方法，虽然已经取得了比较好的分类效果，但是还有以下需要改进的地方。

（1）从图表中可以看出，家居类新闻最影响分类效果，有可能与原始语料有关，可能存在噪音，可以从语料的选择和处理上改进。

（2）SVM有不同的核函数，不同的核函数会影响SVM的效率和最终的分类效果，这也是提升性能的一个需要研究的地方。

（3）在特征的提取中，本次实验只关注了对于字典中出现词对分类的影响，却没有关注没有出现的词的信息的利用。如果考虑这一方法，经济类新闻对整体分类效果的负面影响应该会大大减少，同时也能大大的增强其他新闻类别的分类性能。

（4）此次使用的训练文档数目上各不相同，可能会造成一定的影响，还有训练集的数目还是不够特别的多，会影响到精确度。

# **软件测试**

## 7.1测试人员任务分配

|  |  |
| --- | --- |
| 测试员 | 职责 |
| 王海 | 测试环境部署、制定测试要求与标准、  测试用例的执行、测试文档编写 |
| 吕鑫 | 错误及缺陷的记录跟踪和分析报告、  测试用例的编写和执行、测试文档编写， |
| 马茼茼 | 测试用例的编写和执行、测试文档编写 |
| 胡曦伟 | 测试用例的执行、测试文档编写 |
| 谢欣旖 | 测试用例的执行、测试文档编写 |

## 7.2测试

7.2.1测试步骤及方法

部署测试环境，设计和维护测试用例，通过测试用例对功能，性能如能否查找到对应目标、目标分类是否正确等方面进行测试。报告所发现的缺陷，完成软件测试质量分析报告。测试方法为黑盒测试。

7.2.2测试通过标准

按照需求分析文档进行操作流程，系统能正确响应，设计检索对应的test，分布测试；满足性能要求；产生了预期结果即测试通过。

7.2.3修正错误并重新测试的步骤与要求

填写错误报告表，对出现的错误进行记录；修正程序错误、修改设计说明，重新测试，若再次出现错误，则继续填写错误报告表，修改程序，继续测试。

7.2.4进度计划

根据设计的用例，先进行单元测试（功能测试和性能测试），在进行集成测试，系统测试。

## 7.3测试小结

7.3.1测试环境要求

1、软件

1.1操作系统：win7、win10，在这两种比较常用的操作系统上进行测试。服务器端操作系统为centos7。

1.2测试使用的浏览器：Google浏览器、FirFox浏览器、360浏览器、不同版本的IE浏览器。测试各浏览器是否兼容。

2、硬件：

2.1计算机：最低配置1G内存，单核，50G硬盘。

2.2 显卡：最低配置512M

2.3 显示器：台式机17英寸分辨率1024\*768、台式机19英寸分辨率1440\*900；笔记本14.1寸分辨率1366\*768，笔记本15.4寸分辨率1280\*800。

2.4服务器：服务器端最低配置2G内存，双核，50G硬盘。

3、网络：

3.1 网卡、网线，保证计算机能连接到互联网。

3.2 网速，1M带宽以上。

4、数据准备：

4.1 elasticsearch数据库中数据。

4.2 网点数据：每个运营中心下面添加5个网点。

4.3 信息数据添加，通过检索程序，检索后选取要分析的信息。

5、测试工具：

无，小组分工每个人测试几个功能（反复测试）。

7.3.2测试用例的简单描述

测试用例的命名规则为RC+模板编号+用例编号，如RC0101代表编为01模板下面的编号为01的用例。

# 第8章、总结

通过“实训一”这段时间的实训，完成了现阶段的课题研究。在这个过程中全体成员在项目开发的各个方面了解了很多，付出了很多，学到了很多，也领悟了很多，最终得到了预期的结果。我们在人际交流方面：学会了发现问题如何的与其他人沟通，能够加强交流其中学习到在课程设计的过程中，我们经历了感动，经历了一齐奋斗的酸甜苦辣。也一齐分享了成功的喜悦。这次的课程设计对我们每个人来说都是一个挑战。课程设计中文档的撰写我从来就没有担心过，就是实训类型我真的很担心，平时对这方面的知识接触的就不是很多，而且对于软件我就更抓狂了。这时候小组的力量就体现出来了，各司其职，各尽其能。发挥了群众的效用。和大家想的一样我们也会产生一些小矛盾，当然这是不可避免的。在产生小矛盾的时候，我们没有逃避。重要的是我们如何去解决它。为什么会产生这些矛盾，以怎样的方式去解决它，这是我们就应去思考的问题。我想经过这样的一个过程我们会学到很多，学会了怎样去和别人沟通，理解别人所做的事，别人也会宽容的对待我们，从而我们就在无形之中加强了我们的人际交往潜力。这个经验对我们以后的人生将会发挥很大的作用。毕竟我们是生活在人类这个群体之中的。假如世界上只剩下一个人，那么他不可能长久的生活下去的。

课程设计这样群众的任务光靠团队里的一个人或几个人是不可能完成好的，合作的原则就是要利益均沾，职责公担。如果让任务交给一个人，那样既增加了他的压力，也增大了完成任务的风险，降低了工作的效率。所以在群众工作中，团结是必备因素，要团结就是要让我们在合作的过程中：真诚，自然，微笑；说礼貌用语；不斤斤计较；多讨论，少争论，会谅解对方；对他人主动打招呼；会征求同学的意见，会关心同学，会主动认错，找出共同点；会理解帮忙，信守诺言，尊重别人，持续自己的特色。透过这次实训使我懂得了理论与实际相结合是很重要的，只有理论知识是远远不够的，只有把所学的理论知识与实践相结合起来，从理论中得出结论，才能真正学习到知识，从而提高自己的实际动手潜力和独立思考的潜力。在操作的过程中遇到问题，能够说得是困难重重，这毕竟第一次做的，难免会遇到过各种各样的问题，同时在设计的过程中发现了自己的不足之处，对以前所学过的知识理解得不够深刻，掌握得不够牢固。

再后期的工作中筛选，主题的分类，从单篇的分类到后期多篇或者数以千计文章的聚类，我们小组将持续更新以达到主题分析的最大优化和人力的最大简化

这次实训最后顺利完成了，在设计中遇到了很多专业知识问题，最后在刘小明老师的辛勤指导下，最后游逆而解。同时，在老师的身上我们学也到很多实用的知识，也体会到了科研得出结果的乐趣。再次我们表示感谢!同时，对给过我们帮忙的所有同学和各位指导老师再次表示忠心的感谢!