目录

新闻分类系统设计文档

摘 要：随着网络的普及使用，网络新闻的数量呈现爆炸性增长。本系统从新闻文本内容角度出发，将新闻进行多类分类，结合新闻短文本的特点基于 2012 年搜狗全网新闻语料以及爬取的 2016 年搜狗新闻语料，引入支持向量机、朴素贝叶斯等方法，应用中文文本处理方法处理新闻文本，并设计了应用 SVM、朴素贝叶斯新闻短文本分类系统。这些方法依据新闻内容作为出发点，利用以上方法对网络新闻进行分类识别。此外，报告最后将以上两种方法的结果进行比较分析。

关键词：文本分类；SVM；朴素贝叶斯；短文本；中文信息处理

一、 文本分类与短文本分类介绍

1.1 文本分类介绍

文本分类（Text Categorization）是指依据文本的内容，由计算机根据某种自动分类算法，把文本判分为预先定义好的类别。文本分类是信息存储和信息检索中的重要课题。互联网的飞速发展又给文本分类提供了新的应用平台。网页分类是文本分类在网页文本集合上的应用，它在信息过滤，基于个性化的信息服务等方面有着重要用途。文本分类大致可分为三个步骤：文本的向量模型表示，文本特征选择和分类器训练。数量巨大的训练样本和过高的向量维数是文本分类的两大特点。文本自动分类问题的最大特点和困难之一是特征空间的高维性和文档表示向量的稀疏性。在中文文本分类中，通常采用词条作为最小的独立语义载体，原始的特征空间由可能出现在文章中的全部词条构成。而中文的词条总数有二十多万条，这样高维的特征空间对于几乎所有的分类算法来说都偏大。

1.2 短文本分类介绍

当前，很多突发事件信息率先在微博、BBS、短信、新闻评论中以文本形式传播，这些文本比较短，称为短文本。对互联网短文本的研究有助于监测突发事件的产生、发展、消退，减少突发事件的不良影响。

构建一个互联网短文本突发事件的监测系统，对于维护国家安全、净化网络空间、保障公众知情权等具有十分重要作用。第一：可以有效打击互联网上的违法犯罪行为，利用先进的互联网监督技术，追查犯罪分子，依法制裁，促进社会和谐，在一定程度上减少互联网上的违法信息。第二：增加公众的知情权，让公众在第一时间了解相关网络事件真相，有效减少不实信息在互联网上的传播，误

导公众。同时，也可以方便民众自身观点立场，吸引相关方面的注意力。第三：通过短文本的分类，判断信息的优劣，在不良信息没有产生以前屏蔽不良信息。净化网络环境。第四：系统在舆情预警、跟踪发现特定话题的同时，可以分析发现新词语、分析流行语，监测社会价值倾向。

但是短文本分类有其固有的难点：

(1) 短文本关键词特征稀疏，与一般成句子的长文本相比，短文本的关键词特征稀疏（每个短文本中一般只含有数十个甚至几个关键词），难以充分挖掘出特征之间的关联性。

(2) 样本高度不均衡，短文本应用背景（如网络内容安）需要处理海量的文本流，而其中真正关注的检测对象（如敏感话题、事件）在数量上只占很小的一部分。

所以，高效准确的短文本分类系统成为亟待解决的问题。本系统以新闻分类为基础进行了尝试。

二、 相关工作

文本分类的任务是：对未知类别的文档进行处理，判断它所属预定义类别集中一个或多个类别。随着各种电子形式的文本文档以指数级的速度增长，有效的信息检索、内容管理及信息过滤等应用变得越来越重要和困难。文本分类是一个有效的解决办法，已成为一项具有实用价值的关键技术。近年来，多种统计理论和机器学习方法被用来进行文本分类，掀起了文本分类研究和应用的热潮。

目前文本分类方法分为 2 类，即基于规则的方法和基于概率统计的方法。基于规则的方法归纳出训练样本中规律性的内容以形成规则，并根据此规则确定文本类别。此方法采用的主要算法有决策树、粗糙集、Ripper 方法、boosting 方法等。基于规则的分类方法在规律不明显的领域中应用效果较差。基于概率统计的方法统计出文档中用词等方面的概率分布规律，其本质也是获取一种分类规则，但这种规则不易被人理解。此方法采用的主要算法有 K 邻近方法、贝叶斯方法、Rocchio 方法、支持向量机等。

2.1 中文文本处理及特征选取

中文信息处理起步时间晚，加上汉语的固有难点，很大程度加大了处理难度，主要体现在：①不同于英语单词，中文字之间的组合，产生了很多的词语，达到百万级别，导致特征维度巨大，影响中文信息处理的用时和准确度。②缺乏公认

有效的中文训练语料库，很多学者各自收集并标注自己的文本，产生很多重复劳动，并且效果不佳，分本分类等问题很依赖于人工标注的类别信息，训练数据在很大程度上决定了分类效果，但是目前主流的分类语料来源于英文，权威的会议、期刊也被英文垄断，很难有中文评价标准。③中文的一词多义、一语双关现象，有时候人类都很难理解某些文本语义，中文的语法规则也很难理解，机器理解这些文本更难。

(1) 文本分词

语是语言具有独立意义的最小单元，本分词系统以词语为最小单元，短文本通常是连续的字符串形式，不同于英文，语法结构明显，词语之前以空格为分界线，中文词语间没有明确的分隔符，因此需要增加分词，如下图所示：

图 2-1 互联网短文分类系统框图

使用较多的分词算法有：基于词典匹配或基于统计的分词算法，好的分词算法会兼顾分词精度和运算时间，比较成熟的中文分词工具有：中科院计算所的ICTCLAS（现已经更名 NLPIR）、复旦大学的 FudanNLP、哈尔滨工业大学的 LTP、ANSJ_SEG，开源的 IKAnalyzer 也比较常用。本系统采用 ANSJ_SEG 分词。

(2) 特征选择

特征选择又称为特征提取，是从所有特征中提取对文本分类有贡献的特征，根据评估函数对特征项评估，最终保留的部分就是特征。特征提取可以提取短文本中的有效信息，降低特征复杂程度、简化计算、提升计算效率。目前常用的特征提取方法有：TF-IDF、信息增益(IG)、互信息、文档频率等。本系统采用 TF-IDF 与信息增益(IG)特征。

(3) 文本表示

在对文档进行分类之前，必须把文档表示成为计算机可以处理的形式。空间向量模型是常用的有效的方法之一。空间向量模型的主要思想是：把文本看作一个多维向量，把从文本选出来的一个特征词条当作向量的一维。

本系统朴素贝叶斯部分采用分词词表表示文本，SVM 部分采用处理过的词表对应的 ONE-HOT 向量表示文本

2.2 贝叶斯（Bayes）文本分类

利用 Bayes 方法进行文本分类是目前比较常用的一种手段。Bayes 分类器是一种参数化的分类器，它的分类算法是基于 Bayes 学习模型的一种分类方法，在

机器学习领域中被广泛的研究。它的本质思想是利用词和类别的联合概率 给定文档属于各个类别的概率。Bayes 分类器将短信看作独立的词语集合，通过训练集，由 Bayes 理论得到每个词语在不同类别中的概率大小。

其中，朴素贝叶斯分类（Naïve Bayes）是建立在“Bayes 假设”的基础之上：给定一个实例的类标签，实例中每个属性的出现独立于实例中其他属性的出现，即：所有的特征之间是相互独立的。

优点：贝叶斯分类比较容易实现，具有较好的健壮性，大多数情况下具有较高的分类效率。

缺点：然而贝叶斯分类假定样本属性间没有相互关系，而在实际情况下这种独立性假设并不能完全满足，因此在样本属性关系紧密的情况下，算法的性能就会大打折扣。贝叶斯方法处理时间更快，需要空间更小。

2.3 SVM 文本分类

SVM 方法是另一种常用的垃圾短信分类算法。SVM 理论是一种寻求结构风险最小化的统计分类理论。SVM 方法是从线性可分情况下的最优分类面（Optimal Hyperplane）提出的，所谓最优分类面就是要求分类线不但能将两类样本无错误的分开，而且要使两类之间距离最大。本系统所使用的 SVM 分类器将短信词语作为特征，是否为垃圾短信作为类别（分别对应+1，-1）进行分类。

优点：SVM 擅长解决小样本、非线性和高纬模式识别中的分类问题，可以将低纬线性不可分问题映射为高维线性可分问题，并且存在全局最优解。

缺点：然而如果训练样本偏大，SVM 就比较难以实施，SVM 需要占用大量的内存空间和消耗比较多的运算时间。SVM 主要是解决二分类问题的，对于多分类问题则需要通过多个二分类 SVM 的组合来解决。

2.4 网页爬虫及交互界面

(1) 网页爬虫

网页爬虫又称为网页蜘蛛、网络机器人，是一种按照一定的规则，自动地抓取万维网信息的程序或者脚本，是网络数据采集的一种。相比于浏览器，网页爬虫收集和处理大量数据的能力更为卓越。

传统爬虫的工作流程一般是从一个或若干初始网页的 URL 开始，获得初始网页上的 URL，在抓取网页的过程中，不断从当前页面上抽取新的 URL 放入队列，直到满足系统的一定停止条件。

在抓取网页的过程中，还需要对抓取的网页进行解析以获得想要的内容。传

统爬虫中最常见的基本上是 HTML 文本，也是大多数时候会遇到的情况，例如抓取一个网页，得到的是 HTML，然后需要解析一些常见的元素，提取一些关键的信息。虽然 HTML 属于结构化的文本组织，但是一般所需关键信息并非直接可以得到，需要进行对 HTML 的解析查找，甚至一些字符串操作才能得到。常见的解析方式有：CSS 选择器、XPATH 和正则表达式。

(2) 交互界面

交互界面是人和计算机进行信息交换的通道，用户通过交互界面向计算机输入信息、进行操作，计算机则通过交互界面向用户提供信息，以供阅读、分析和判断。交互不一定需要很华丽的界面，但是使用过程肯定是很人性化，减少用户思考返回的次数。

可用性是交互界面的基本而且重要的指标，它是对可用程度的总体评价。也是从用户角度衡量产品是否有效、易学、安全、高效、好记、少错的质量指标。可用性是一个多因素概念，涉及到容易学习、容易使用、系统的有效性、用户满意，以及把这些因素与实际使用环境联系在一起针对特定目标的评价。

同时，交互界面的目标不止于此，它还包括要考虑用户的期望和体验，可用性保证产品可用，基本功能完备且方便；而体验在于给用户一些与众不同的或者意想之外的感觉。也就是说，可用，是产品应该做到的，理所应当的，体验则是额外的惊喜和收获。

三、 新闻分类系统算法设计及实现

新闻分类系统整体流程图如下：

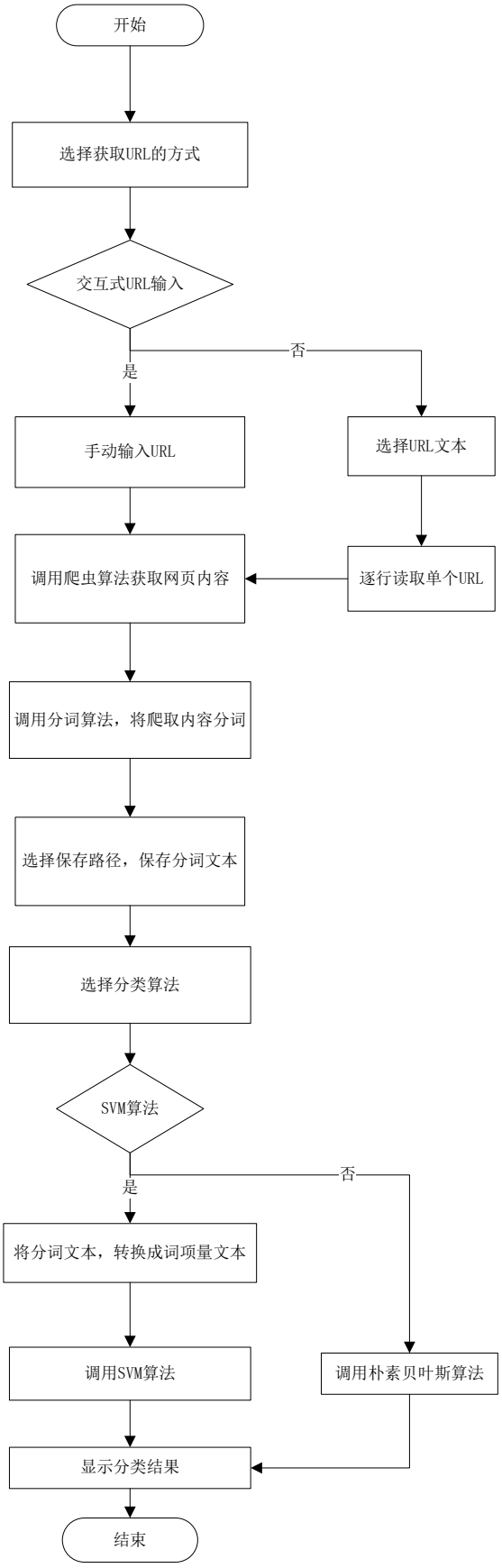

图 3-1 新闻分类系统整体流程图

3.1 中文文本处理及特征选取系统

3.1.1. 原始新闻文本抽取

(1) 伪 XML 格式转换

原始搜狗新闻语料库中新闻存储格式是缺少“根标签”的伪 XML 格式，并且含有大量“&”字符，要转换成“&”无法进行批量 XML 格式文本抽取，如下图所示：

```
<doc>
<url>http://lady.163.com/special/00261PJK/zresult.html?wt=%CE%D2%BB%E1%B7%A2%B2%C6%C2%F0&amp;
year=2011&month=6&day=17&hour=12&minute=00&prov=%C9%CF%BA%A3&city=%C6%D6%B6%AB%D0%C2%C7%F8&type=2</url>
<docno>01346ee5d5932277-b2d5d9a362314a50</docno>
<contenttitle></contenttitle>
<content></content>
</doc>
```

图 3-2 处理前的搜狗新闻语料

批量处理后的搜狗新闻语料示例如下：

```
<doc>
<url>http://lady.163.com/special/00261PJK/zresult.html?wt=%CE%D2%BB%E1%B7%A2%B2%C6%C2%F0&amp;amp;year=2011&amp;month=6&amp;day=17&amp;hour=12&amp;
minute=00&amp;prov=%C9%CF%BA%A3&amp;city=%C6%D6%B6%AB%D0%C2%C7%F8&amp;type=2</url>
<docno>01346ee5d5932277-b2d5d9a362314a50</docno>
<contenttitle></contenttitle>
<content></content>
</doc>
```

图 3-3 处理后的搜狗新闻语料

(2) 新闻 URL、内容及类别信息提取

将搜狗语料库语料进行格式转换之后，应用 dom4j 开源 JAR 包处理新闻预料，使用字符串匹配不同网站 URL 中类别关键词，比如“sports”、“finance”等，并且按照财经(类别号：1)、科技(类别号：2)、汽车(类别号：3)、房产(类别号：4)、体育(类别号：5)、娱乐(类别号：6)、其它类(类别号：7)的分类标号分别提取，将相应的新闻内容、URL 提取出，并分类存储。提取结示例如下：

```
http://finance.sina.com.cn/stock/hkstock/quote.html?code=692
http://finance.sina.com.cn/money/forex/20080613/20302275105.shtml
http://finance.sina.com.cn/stock/t/20080616/20332278186.shtml
http://finance.sina.com.cn/money/fund/20080617/06074989037.shtml
http://finance.sina.com.cn/money/future/fmnews/20080617/09154990091.shtml
http://finance.sina.com.cn/money/future/20080606/14154957917.shtml
http://finance.sina.com.cn/stock/t/20080627/02202299846.shtml
http://finance.qq.com/a/20080628/001580.htm
```

图 3-4 提取的财经类新闻 URL

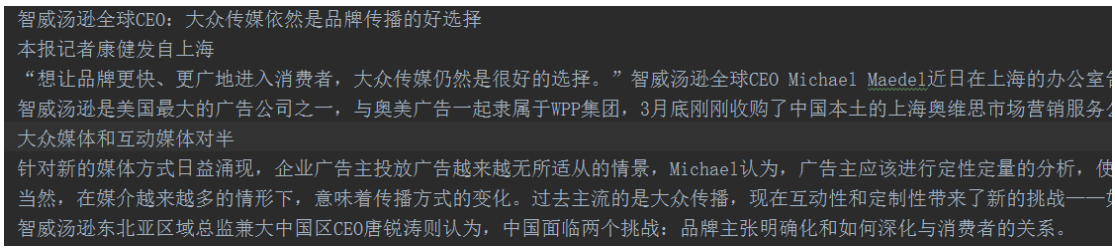

图 3-5 提取的某一新闻内容

3.1.2. 新闻文本分词去重

将新闻内容文本提取之后，应用 ANSJ_SEG 开源 JAR 包，对新闻内容文本进行分词、去停用词等操作，每篇新闻生成一个分词词项文件。例如下图：

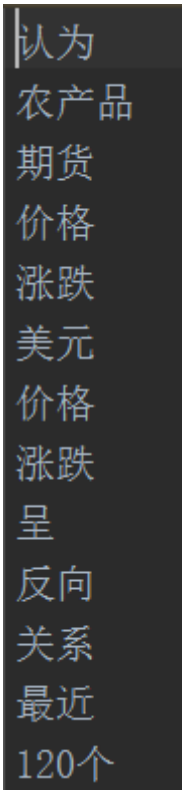

图 3-6 新闻分词词项文件

3.1.3. 新闻训练数据筛选及文本特征选取

(1) 新闻筛选

每一篇新闻对应一个分词词项文件，但有很多新闻分词结果中只包含数字或图片的超链接，缺少有用信息，不利于分类器训练。所以人工选取大于 2KB 的分词词项文件文件，每一类新闻选取 2000 篇符合条件的新闻，一共 14000 篇新闻作为训练数据。

(2) TF-IDF 特征筛选

TF-IDF 的主要思想是：如果某个词或短语在一篇文章中出现的频率 TF 高，并且在其他文章中很少出现，则认为此词或者短语具有很好的类别区分能力，适

合用来分类。TF-IDF 实际上是：TF× IDF，TF 词频(Term Frequency)，IDF 反文档频率(Inverse Document Frequency)。TF 表示词条 t 在文档 d 中出现的频率，IDF 定义如下：

$$\mathrm{IDF} = \log\frac{N}{N_t}$$

N 为总文档数目，N_t为含有词项 t 的文档数目。IDF 的主要思想是：如果包含词条 t 的文档越少，也就是N_t越小，IDF 越大，则说明词条 t 具有很好的类别区分能力。

将 14000 篇新闻分词词项进行去重整合，整合成一个包含 140000 词项的总词表，对词表中每个词项计算 TF-IDF 值，并由高到低对词项重排，生成按照 TF-IDF 值重排的词典，排位越高，表明此词项包含类别信息越多。

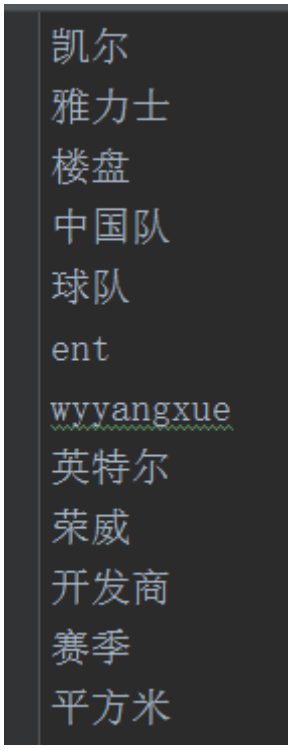

图 3-7 按照 TF-IDF 值重排的词典

(3) 信息增益（IG）特征筛选

信息增益是指期望信息或者信息熵的有效减少量，是信息论中比较重要的一个计算方法，该方法能够估算系统中新引入的特征所带来的信息量，即信息的增加量。

熵表示任何一种能量在空间中分布的均匀程。能量分布得越均匀，熵就越大。后来香农将其引入到信息论中称为信息熵。信息熵在随机事件发生之前，它是结果不确定性的量度；在随机事件发生之后，它是人们从该事件中所得到信息量。

信息增益是一种基于熵的评估方法，其用于特征选择时，衡量的是某个词的

出现与否对判断一个文本是否属于某个类所提供的信息量，其一般定义为某一特征值在文档中出现前后的信息量之差，计算公式如下公式所示。

$$IG(w) = P(w)\sum_{i=1}^{|c|} P(C_i \mid w)\log\frac{P(C_i \mid w)}{P(C_i)} + P(\overline{w})\sum_{i=1}^{|c|} P(C_i \mid \overline{w})\log\frac{P(C_i \mid \overline{w})}{P(C_i)}$$

其中，P(w)表示特征词 w 在文本中出现的概率，$P(C_i \mid w)$表示文本包含 w 时属于C_i类的条件概率，$P(C_i)$表示C_i类文本在文本集中出现的概率，$P(\overline{w})$表示文本中不包含特征词 w 的概率，$P(C_i \mid \overline{w})$表示文本不包含词条 w 时属于C_i类的条件概率，$|c|$表示类别总数。

据公式，对分词统计后的共 75976 个词项计算其信息增益值，排序后取前 5000 作为特征组，排序部分截图如图所示：

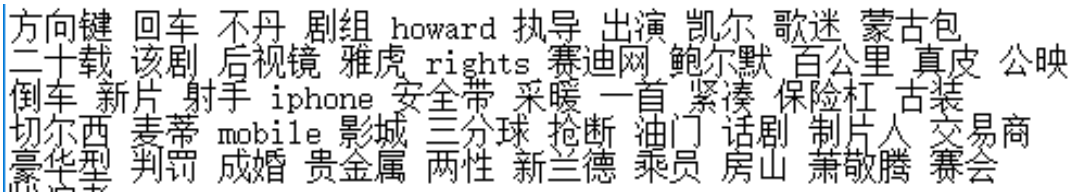

图 3-8 按照信息增益重排的词典

(4) ONE-HOT 特征向量

选取排序后词典的前 10000 个词项作为主特征维数，将每一篇新闻的分词词项与该 10000 个词项的词表进行比对，如果含有 10000 个词项的某一个词项，则对应位置输出 1，若没有，则输出 0，即每一篇新闻生成一个 10000 维的 ONT-HOT 特征向量来表示该新闻文本内容。

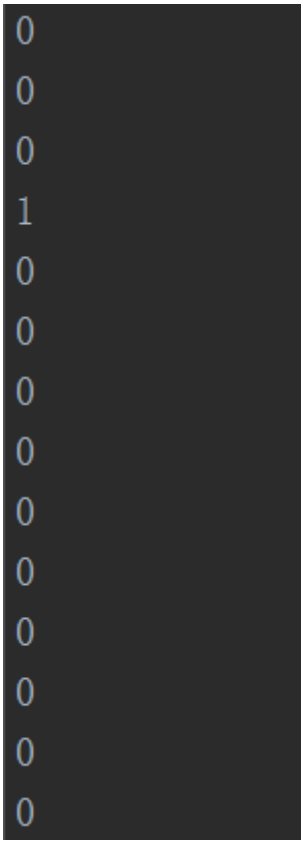

图 3-9 ONT-HOT 特征向量

3.2 基于贝叶斯（Bayes）算法的新闻分类系统

3.2.1. 贝叶斯分类模型

贝叶斯利用统计学的贝叶斯定理，通过某个对象的先验概率，计算出其后验概率，然后选择最大后验概率的类作为该对象所属的类。贝叶斯分类是假定各个属性之间是相互独立的。

贝叶斯定理公式如下：

$$\mathrm{P(A|B)} = \frac{P(B|A)P(A)}{P(B)}$$

在这个公式中，A，B 表示随机事件，P(A)、P(B)分别表示 A、B 发生的概率，P(A|B)表示在条件 B 下 A 发生的概率，P(B|A)表示在条件 A 下 B 发生的概率。

贝叶斯分类器的工作原理：

(1) 设训练数据集 D 对应的属性集为 n 个样本属性变量, C 是具有 m 个类 C 的类标号属性变量。训练数据集 D 中的每个样本都可以用一个 n 维的向量 T 表示。

(2) 假定存在 m 个类别$C_1, C_2, \ldots, C_j$，分类器将样本T分给类别C_j，当：

$$\mathrm{P}(C_j|\mathrm{T}) > \mathrm{P}(C_i|\mathrm{T}) \quad 1 \le \mathrm{i} \le \mathrm{m}, \mathrm{i}! = \mathrm{j}$$

$$\mathrm{P}(C_j|\mathrm{T}) = \frac{P(T|C_j)P(C_j)}{P(T)}$$

(3) 由于$P(T)$对所有类为常数，最大化$P(C_j|T)$可转化为最大化$P(T|C_j)P(C_j)$。类的先验概率$P(i)$可以用N_i/N来估计，其中N_i是数据集 D 中属于类 C 的样本个数，N 是数据集 D 的样本总数。

(4) 朴实贝叶斯定理是假定各个属性之间是相互独立的，没有依赖关系, 则：

$$P(T|C_j) = P(t_1, t_2, t_3, \cdots t_n|C_j)$$

对样本 T 进行分类，先要计算在每个类C_j条件下 T 的概率$P(T|C_j)$, 这样样本 T 就被分配到类C_j中，当且仅当$\mathrm{P}(\mathrm{T}|\mathrm{C_j})\mathrm{P}(C_j) > \mathrm{P}(\mathrm{T}|\mathrm{C_i})\mathrm{P}(C_i)$, 其中$1 \le \mathrm{i} \le \mathrm{m}$ 且 $\mathrm{i} \neq \mathrm{j}$, 即 T 被分配到最大的类别中。

3.2.2. 算法实现及改进

本分类器选用多元分布模型计算，根据《Introduction to Information Retrieval》，多元分布模型计算准确率更高。

(1) 算法实现：

1) 计算概率用到了 BigDecimal 类实现任意精度计算
2) 用交叉验证法做十次实验，对准确率、召回率取平均值，并求出 F1 值。
3) 根据正确类目文件和分类结果文计算混淆矩阵并且输出

(2) 算法改进：

为了进一步提高贝叶斯算法的分类准确率，对以下两个方面进行改进：

1) 优化特征词的选取策略（剔除 IDF 值偏大明显的人名、地名等词）。
2) 改进多项式模型的类条件概率的计算公式，改进为类条件概率：

$\mathrm{P}(\mathrm{T_k}|\mathrm{C})$=（类 C 下单词$\mathrm{T_k}$在各个文档中出现过的次数之和+0.001）/（类 C 下单词总数+训练样本中不重复特征词总数

其中，当分子当$\mathrm{T_k}$没有出现时，只加 0.001。

分类系统使用 JAVA 语言实现，算法流程图如下图所示：

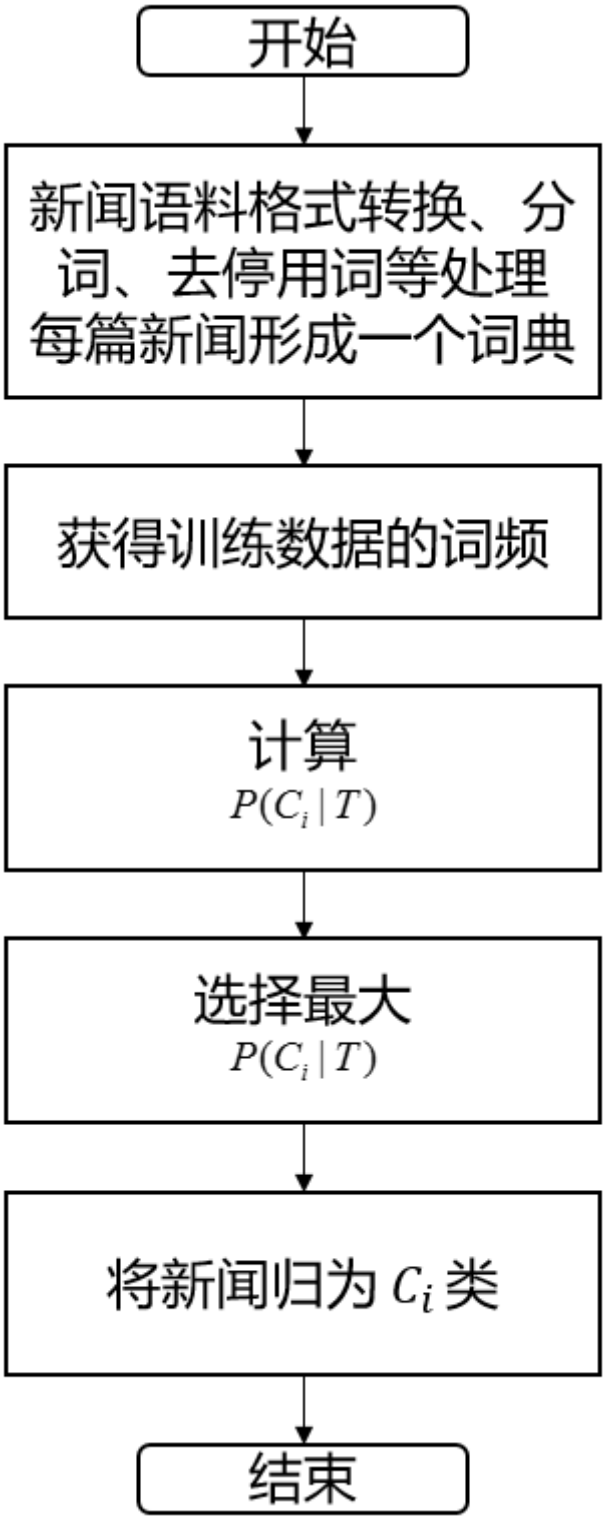

图 3-10 贝叶斯垃圾短信分类系统流程图

3.3 基于支持向量机(SVM)算法的新闻分类系统

3.3.1. 支持向量机(SVM)分类模型

由于考虑到新闻文本分类时的高维度，非线性特征，所以选择了能够很大程度上克服“维数灾难”和“过学习”等问题的支持向量机（SVM）算法。支持向

量机的原理可以描述为寻找一个满足分类要求的最优分类超平面，使得该超平面在保证分类精度的同时，能够使超平面两侧的空白区域大化，如图 2 所示。且在高维划分时巧妙运用了核函数，核函数虽然也是讲特征进行从低维到高维的转换，但它事先在低维上进行计算，而将实质上的分类效果表现在了高维上，也就避免了直接在高维空间中的复杂计算，极大地减轻了计算负担。理论上，支持向量机能够实现对线性可分数据的最优分类。

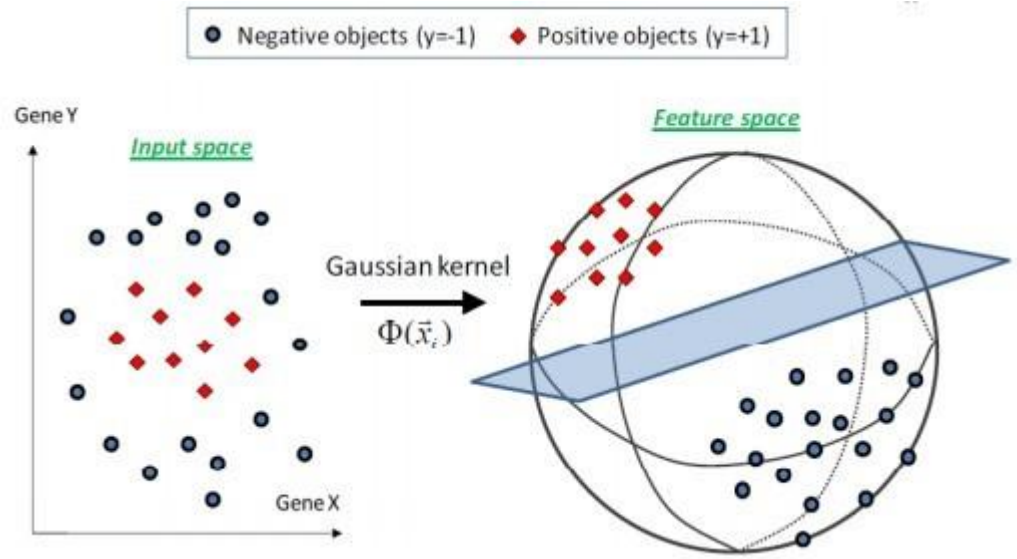

图 3-11 支持向量机原理示意

考虑一个用某特征空间的超平面对给定训练数据集作二值分类的问题. 对于给定样本点:

$$(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l), x_i \in R_n, y_i \in \{-1, +1\}$$

其中向量 x_i 可能是从对象样本集中抽取某些特征直接构造的向量，也可能是原始向量通过某个核函数映射到核空间中的映射向量。

在特征空间中构造分割平面：

$$(w \cdot x) + b = 0$$

使得

$$\begin{cases} (w \cdot x_i) + b \geq 1, y_i = 1 \\ (w \cdot x_i) + b \geq 1, y_i = -1 \end{cases}$$

可以计算出，训练数据集到某一给定的分割平面的最小距离为：

$$p(w,b) = \min \frac{w \cdot x_i}{|w|} - \max \frac{w \cdot x_i + b}{|w|} = \frac{2}{|w|}$$

从 SVM 对优化分割平面的定义可以看出，对该平面的求解问题可以简化为：在满足条件上式的情况下，计算能最大化 P(w,b)的分割平面的法向量 w 和偏移量 b.

由上式可见，最优分割平面的求解等价于最大化下面的式：

$$\Phi(w) = \frac{1}{2}\|w\|^2$$

3.3.2. 算法实现

本系统采用 JAVA 编程实现，使用 LIBSVM 开源 JAR 包实现。

LIBSVM 是台湾林智仁教授 2001 年开发的一套支持向量机的库，可以很方便的对数据做分类或回归。由于 LIBSVM 程序小，运用灵活，输入参数少，开源，易于扩展等优点，已经成为目前国内应用最多的 SVM 的库。该项目中也是主要引用了 LIBSVM 包来进行支持向量机的核心算法，使用的是最基本的 C-SVC 型 SVM，核函数采用常用的 RBF 函数。

其中核心的三个部分为数据格式化，训练模型和利用模型测试。数据格式化即将预处理后的分词文本写成 svmtrain 识别格式，即：

<label> <index1>:<value1> <index2>:<value2> ...

其中<label>是训练数据集的目标值，它是分类中标识某类的整数(支持多个类)，我们的分类目标为体育，科技，财经，娱乐，汽车，教育，其他七类，因此这部分分别设置为 1,2,3,4,5,6,7 七个连续整数；<value>为实数，也就是我们常说的自变量，即词项的是否存在。并通过 TF-IDF 排序来对各词项赋予不同权重，优化实验结果，最终采取以下权重取值：

特征量=N, 特征在 TF-IDF 排序中位置=X，weight =(N/2+1-X)*0.01

下图为 svmtrain 函数的调用流程图，描述了 LIBSVM 中分类模型训练函数的主要思路：

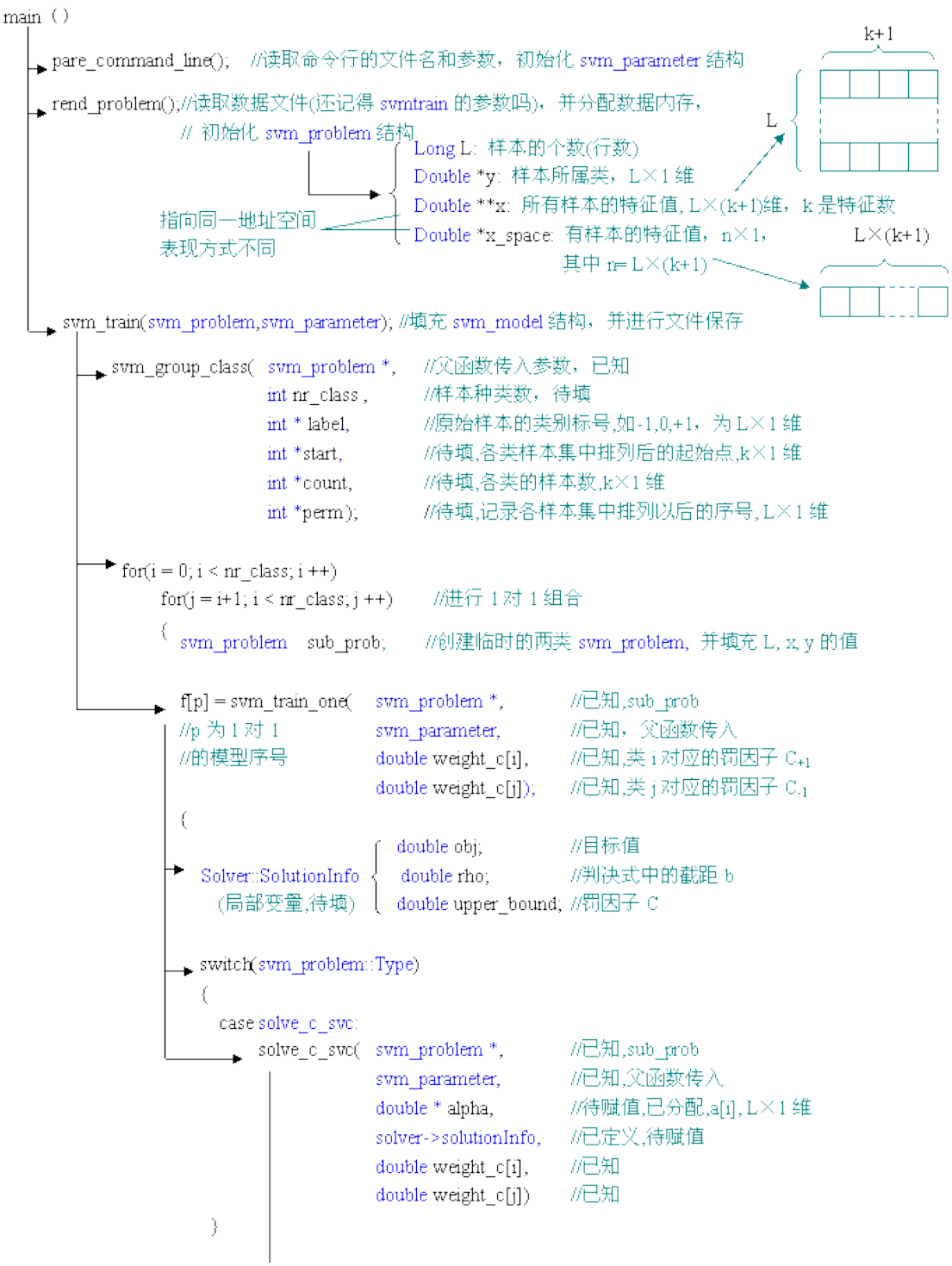

图 3-12 svmtrain 调用流程

3.4 网页爬虫及交互界面模块

3.4.1. 网页爬虫模块

(1) 模块主要内容：

对于一个给定的搜狐新闻 URL，使用 jsoup 爬取搜狐新闻正文。

jsoup 简介：jsoup 是一款 JAVA 的 HTML 解析器，可直接解析某个 URL 地

址、HTML 文本内容。它提供了一套非常省力的 API，可通过 DOM，CSS 以及类似于 jQuery 的操作方法来取出和操作数据。

(2) 模块流程图：

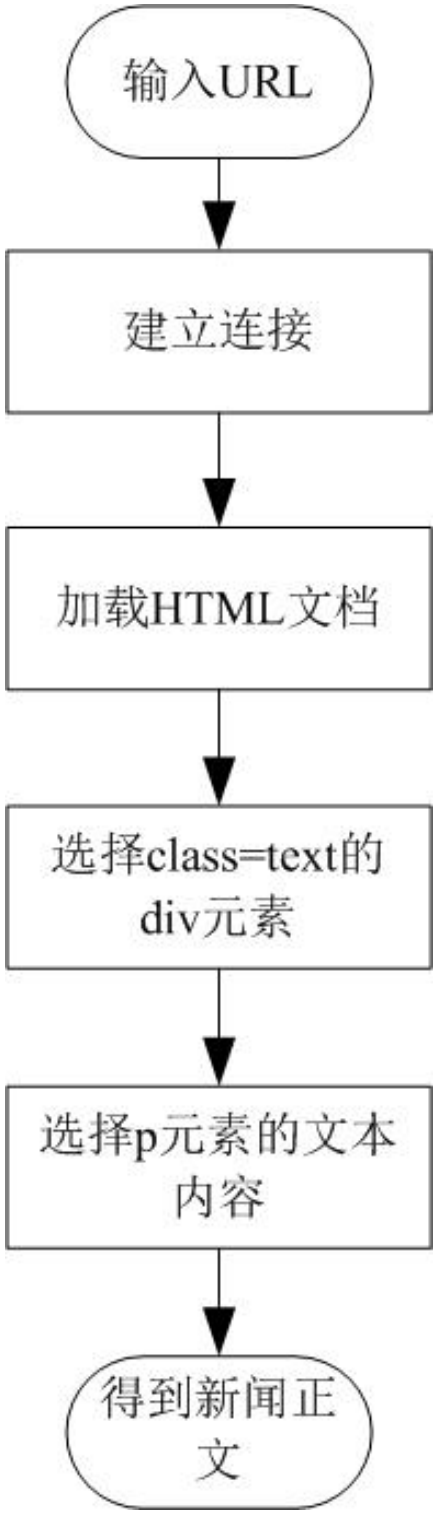

图 3-13 网页爬虫模块流程图

具体实现：

1） 输入目标 URL：

2） 建立连接 coon：

Connection conn = Jsoup.connect(url)；

3） 加载 HTML 文档 doc：

Document doc = conn.timeout(100000).get()；

4） 选择定义了“class=text”的 div 元素：

Elements results = doc.select(“div.text”)；

5） 选择 div 元素中的 p 元素：

Elements p_list = element.select("p")；

6） 选择 p 元素的文本内容，汇总成新闻正文 content：

Stringcontent;

content+=p.text() ;

3.4.2. 交互界面模块

根据任务要求，设计的新闻分类系统交互界面如下：

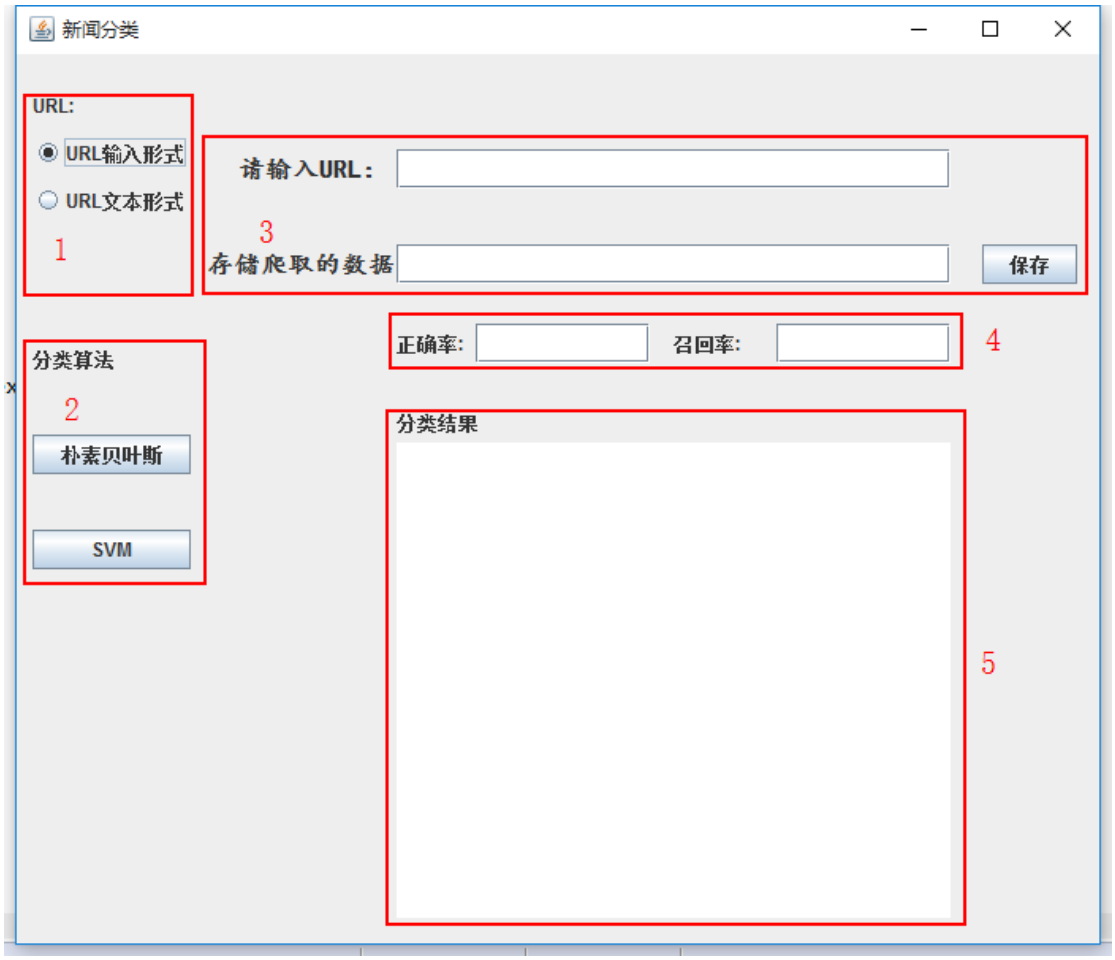

图 3-14 交互界面

主界面主要分为 5 个部分：

(1) 选择交互式 URL 输入或者文本输入。

(2) 选择使用的分类算法。

(3) 输入 URL 或者选择文本，以及结果的保存路径。

(4) 显示正确率与召回率。

(5) 显示分类结果。

该交互界面简洁明了，功能齐全，并且较为完整的展现了分类的结果。

四、 实验结果

4.1. 评价指标

算法实验采用 10 折交叉验证的方法，取 10%的训练数据做测试，其余 90%做训练，总共进行 10 次，取平均值作为实验结果。

实验结果的评价指标包括：

对于检测识别有如下四种情况：

真阳性（RR）：该类别被识别为该类别的样本数

假阳性（NR）：其他被识别为该类别的样本数

真阴性（NN）：其他类别识别为其他类别的样本数

假阴性（RN）：该被识别为其他类别的样本数

(1) 查准率（Precision, 简记为 P），查准率表示的意思是分类器分出结果的一个类，C 类中的文档在实际情况中确实属于 C 类的数目，这个数目在类 C 中所占的比例就是查准率。查准率越高，C 类中判断失误的文档数就越少，查准率的计算公式表示为：

$$Precision = \frac{RR}{RR + NR}$$

(2) 查全率（Recall, 简记为 R），查全率表示的意思是实际情况中所有属于 C 类的文档，经过分类器的判断，可能不是 100%的都分到了类 C 中，分到类 C 中的数目占实际所有属于 C 类的文档的比率。查全率越高，则表示将实际情况中属于类 C 的文档被分类器归为其它类的可能性越小。查全率的公式表示为：

$$Recall = \frac{RR}{RR + RN}$$

(3) F 值（F-measure）, F 值是一种整体性能评价方法，它综合和了上面两种评价指标。计算 F 值要先计算 P 和 R 的值，F 值得公式表示为：

$$F - Measure = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}$$

当λ=1时，查准率 P 和查全率 R 重要性相同：

$$F1-Measure=\frac{2*PR}{P+R}$$

当λ>1时，查全率 R 比查准率 P 更重要；当λ<1时，查准率比查全率更重要。

本系统采取七维（七类）混淆矩阵计算 P、R、F 值。

4.2. 基于贝叶斯（Bayes）算法的新闻分类系统实验结果

4.2.1. 基于 2012 年搜狗新闻语料库实验结果

(1) 十次交叉验证混淆矩阵结果如下：

	0	1	2	3	4	5	6	
0	174	8	8	12	0	1	2	0.848780487804878
1	31	144	7	1	1	2	14	0.72
2	11	3	191	0	1	1	2	0.9138755980861244
3	28	0	5	162	0	1	3	0.8140703517587939
4	4	1	11	1	169	3	11	0.845
5	9	3	6	0	2	161	20	0.8009950248756219
6	25	10	17	9	25	6	108	0.54

The accuracy for Naive Bayesian Classifier in 0th Exp is :0.7842998585572843

The recall for Naive Bayesian Classifier in 0th Exp is :0.9165289256198347

	0	1	2	3	4	5	6	
0	182	11	5	5	0	0	2	0.8878048780487805
1	56	124	6	1	0	1	13	0.6169154228855721
2	9	0	199	0	0	0	2	0.9476190476190476
3	24	0	2	171	0	2	1	0.855
4	3	0	7	2	171	1	17	0.8507462686567164
5	6	3	7	5	4	163	14	0.806930693069307
6	30	11	15	12	9	4	120	0.5970149253731343

The accuracy for Naive Bayesian Classifier in 1th Exp is :0.795774647887324

The recall for Naive Bayesian Classifier in 1th Exp is :0.9338842975206612

0	1	2	3	4	5	6

0	188	6	4	8	0	0	0	0.912621359223301
1	38	135	6	7	0	2	12	0.675
2	14	1	194	0	1	0	0	0.9238095238095239
3	17	1	2	174	0	2	4	0.87
4	7	1	3	3	172	1	14	0.8557213930348259
5	4	3	6	1	2	171	15	0.8465346534653465
6	42	9	14	12	13	7	104	0.5174129353233831

The accuracy for Naive Bayesian Classifier in 2th Exp is :0.8014084507042254

The recall for Naive Bayesian Classifier in 2th Exp is :0.9404958677685951

	0	1	2	3	4	5	6	
0	181	6	3	9	0	4	2	0.8829268292682927
1	54	127	3	6	0	1	10	0.6318407960199005
2	11	0	198	0	0	0	1	0.9428571428571428
3	21	1	4	171	0	0	3	0.855
4	9	1	8	0	171	2	9	0.855
5	7	6	3	2	1	161	21	0.8009950248756219
6	17	4	18	8	26	5	123	0.6119402985074627

The accuracy for Naive Bayesian Classifier in 3th Exp is :0.7983074753173484

The recall for Naive Bayesian Classifier in 3th Exp is :0.9355371900826446

	0	1	2	3	4	5	6	
0	187	4	2	8	0	0	5	0.9077669902912622
1	25	148	9	4	0	0	14	0.74
2	9	1	197	0	0	0	2	0.9425837320574163
3	8	0	3	183	0	2	4	0.915
4	4	1	2	1	177	3	13	0.8805970149253731
5	2	4	9	1	1	165	20	0.8168316831683168
6	22	7	18	8	9	8	128	0.64

The accuracy for Naive Bayesian Classifier in 4th Exp is :0.8356840620592384

The recall for Naive Bayesian Classifier in 4th Exp is :0.9793388429752066

	0	1	2	3	4	5	6	
0	183	8	6	8	0	0	0	0.8926829268292683
1	40	142	9	4	0	1	5	0.7064676616915423
2	11	2	196	0	0	0	1	0.9333333333333333
3	22	1	3	166	0	4	4	0.83
4	3	2	11	0	172	0	13	0.8557213930348259
5	5	3	17	7	3	158	9	0.7821782178217822
6	43	10	19	16	7	7	99	0.4925373134328358

The accuracy for Naive Bayesian Classifier in 5th Exp is :0.7859154929577464

The recall for Naive Bayesian Classifier in 5th Exp is :0.9223140495867769

	0	1	2	3	4	5	6	
0	185	8	4	7	0	0	2	0.8980582524271845
1	34	144	6	4	0	2	11	0.7164179104477612
2	10	2	197	0	0	0	1	0.9380952380952381
3	18	2	4	173	0	1	2	0.865
4	4	1	12	2	167	0	14	0.835
5	6	2	9	3	3	163	15	0.8109452736318408
6	40	5	16	12	11	2	115	0.572139303482587

The accuracy for Naive Bayesian Classifier in 6th Exp is :0.8062015503875969

The recall for Naive Bayesian Classifier in 6th Exp is :0.9454545454545454

	0	1	2	3	4	5	6	
0	183	4	4	14	0	0	0	0.8926829268292683
1	36	133	12	5	0	1	13	0.665
2	11	1	198	0	0	0	0	0.9428571428571428
3	29	1	2	165	0	0	3	0.825
4	6	0	16	0	161	1	17	0.8009950248756219
5	4	2	5	2	1	177	11	0.8762376237623762
6	37	6	26	7	22	5	98	0.48756218905472637

The accuracy for Naive Bayesian Classifier in 7th Exp is :0.7857646229739254

The recall for Naive Bayesian Classifier in 7th Exp is :0.9214876033057852

	0	1	2	3	4	5	6	
0	181	3	5	11	1	0	5	0.8786407766990292
1	31	140	10	5	1	3	11	0.6965174129353234
2	12	0	196	1	0	0	1	0.9333333333333333
3	18	5	0	176	0	0	1	0.88
4	4	1	8	0	175	1	12	0.8706467661691543
5	0	2	14	1	1	168	16	0.8316831683168316
6	18	12	18	5	14	7	127	0.6318407960199005

The accuracy for Naive Bayesian Classifier in 8th Exp is :0.8184377199155525

The recall for Naive Bayesian Classifier in 8th Exp is :0.9611570247933884

	0	1	2	3	4	5	6	
0	183	5	6	5	0	2	4	0.8926829268292683
1	47	128	9	2	0	0	14	0.64
2	7	0	201	1	0	0	0	0.9617224880382775
3	19	0	3	175	0	1	2	0.875
4	2	0	5	1	180	3	9	0.9
5	5	1	15	3	1	162	14	0.8059701492537313
6	38	7	23	10	9	7	106	0.53

The accuracy for Naive Bayesian Classifier in 9th Exp is :0.8021201413427562

The recall for Naive Bayesian Classifier in 9th Exp is :0.9380165289256198

The average accuracy for Naive Bayesian Classifier in all Exps is :0.80139140221

The average recall for Naive Bayesian Classifier in all Exps is :0.9394214876033056

The average F1 for Naive Bayesian Classifier in all Exps is :0.8704064449068027

(2) 十倍交叉验证评价结果表

表 4-1 基于 2012 年搜狗新闻语料库 Bayes 十倍交叉验证评价结果表

次数	P	R	F
1	0.7843	0.9165	0.8324
2	0.7958	0.9339	0.86485
3	0.8014	0.9405	0.87095
4	0.7983	0.9355	0.8669
5	0.8357	0.9793	0.9075
6	0.7859	0.9223	0.8541
7	0.8062	0.9454	0.8758
8	0.7858	0.9215	0.8536
9	0.8184	0.9612	0.8898
10	0.8021	0.9380	0.87005
平均值	0.80139	0.9394	0.8704

(3) 十倍交叉验证结果图

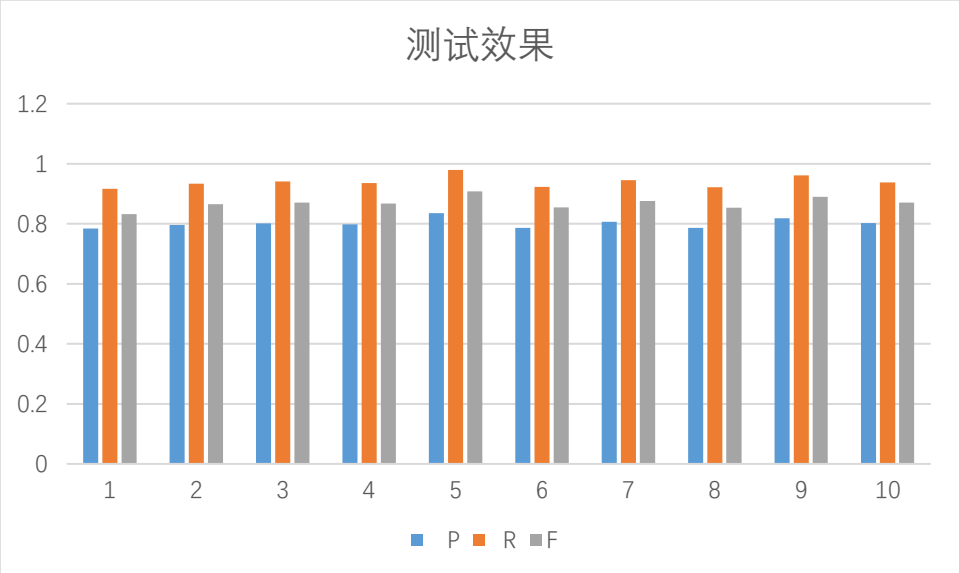

图 4-1 基于 2012 年搜狗新闻语料 Bayes 十倍交叉验证评价结果图

由此可见，贝叶斯分类器在 2012 年搜狗新闻语料库新闻分类上的效果表现良好。

4.2.2. 基于 2012 年与 2016 年混合搜狗新闻语料库实验结果

(1) 十次交叉验证混淆矩阵结果：

	0	1	2	3	4	5	6	
0	196	8	4	8	0	0	8	0.875
1	50	156	2	0	0	0	18	0.6902654867256637
2	24	2	182	0	2	0	12	0.8198198198198198
3	44	0	4	156	0	0	10	0.7289719626168224
4	2	0	6	0	192	0	36	0.8135593220338984
5	8	0	8	0	12	66	88	0.3626373626373626
6	24	6	8	2	36	2	166	0.680327868852459

The accuracy for Naive Bayesian Classifier in 0th Exp is :0.7196382428940569
The recall for Naive Bayesian Classifier in 0th Exp is :0.7527027027027027

	0	1	2	3	4	5	6	
0	180	16	6	6	0	0	18	0.7964601769911505
1	28	166	10	2	0	2	18	0.7345132743362832
2	14	0	202	0	0	0	8	0.9017857142857143
3	26	2	6	170	0	0	12	0.7870370370370371
4	4	0	16	0	198	2	16	0.8389830508474576
5	6	10	4	0	12	68	82	0.37362637362637363
6	26	18	12	6	16	0	166	0.680327868852459

The accuracy for Naive Bayesian Classifier in 1th Exp is :0.7400257400257401
The recall for Naive Bayesian Classifier in 1th Exp is :0.777027027027027

	0	1	2	3	4	5	6	
0	190	16	4	6	0	0	10	0.8407079646017699
1	58	134	2	2	0	0	32	0.5877192982456141
2	8	2	208	0	0	0	6	0.9285714285714286
3	36	6	0	170	0	0	2	0.794392523364486
4	0	0	0	2	192	0	42	0.8135593220338984
5	8	0	4	2	8	54	106	0.2967032967032967
6	32	6	2	8	16	0	180	0.7377049180327869

The accuracy for Naive Bayesian Classifier in 2th Exp is :0.7258687258687259
The recall for Naive Bayesian Classifier in 2th Exp is :0.7621621621621621

	0	1	2	3	4	5	6	
0	206	10	4	2	0	0	4	0.911504424778761
1	50	156	4	0	2	0	14	0.6902654867256637
2	10	0	204	0	0	0	8	0.918918918918919
3	30	2	2	178	0	0	4	0.8240740740740741
4	6	0	2	2	210	0	16	0.8898305084745762
5	6	6	16	4	2	76	72	0.4175824175824176

6	42	14	10	6	14	0	158	0.6475409836065574

The accuracy for Naive Bayesian Classifier in 3th Exp is :0.7654639175257731
The recall for Naive Bayesian Classifier in 3th Exp is :0.8027027027027027

	0	1	2	3	4	5	6	
0	204	8	2	6	2	0	4	0.9026548672566371
1	38	168	2	4	0	0	14	0.7433628318584071
2	18	2	198	0	2	0	4	0.8839285714285714
3	22	2	6	168	0	0	16	0.7850467289719626
4	4	0	4	2	196	0	30	0.8305084745762712
5	10	6	6	0	14	66	80	0.3626373626373626
6	32	4	8	4	18	6	172	0.7049180327868853

The accuracy for Naive Bayesian Classifier in 4th Exp is :0.7551546391752577
The recall for Naive Bayesian Classifier in 4th Exp is :0.7918918918918919

	0	1	2	3	4	5	6	
0	194	6	4	14	0	0	8	0.8584070796460177
1	58	140	4	2	2	0	22	0.6140350877192983
2	20	4	192	0	0	0	8	0.8571428571428571
3	36	0	0	166	0	0	14	0.7685185185185185
4	4	2	0	0	206	0	24	0.8728813559322034
5	8	2	6	2	6	74	84	0.4065934065934066
6	24	4	18	6	24	0	170	0.6910569105691057

The accuracy for Naive Bayesian Classifier in 5th Exp is :0.7329910141206675
The recall for Naive Bayesian Classifier in 5th Exp is :0.7716216216216216

	0	1	2	3	4	5	6	
0	202	8	2	8	0	0	6	0.8938053097345132
1	34	160	10	0	0	0	22	0.7079646017699115
2	10	2	200	0	0	0	10	0.9009009009009009
3	20	2	2	188	0	0	4	0.8703703703703703
4	2	2	8	0	196	0	28	0.8305084745762712
5	2	8	2	4	0	62	104	0.34065934065934067
6	6	10	2	0	38	2	186	0.7622950819672131

The accuracy for Naive Bayesian Classifier in 6th Exp is :0.7693298969072165
The recall for Naive Bayesian Classifier in 6th Exp is :0.8067567567567567

	0	1	2	3	4	5	6	
0	186	6	10	12	0	0	12	0.8230088495575221
1	38	142	10	4	0	2	32	0.6228070175438597

2	4	4	208	0	0	0	8	0.9285714285714286
3	14	4	2	184	0	0	10	0.8598130841121495
4	2	0	0	2	208	0	24	0.8813559322033898
5	2	8	6	8	2	70	86	0.38461538461538464
6	14	14	8	2	20	0	186	0.7622950819672131

The accuracy for Naive Bayesian Classifier in 7th Exp is :0.7619047619047619
The recall for Naive Bayesian Classifier in 7th Exp is :0.8

	0	1	2	3	4	5	6	
0	206	2	2	4	0	4	8	0.911504424778761
1	42	158	4	0	0	0	22	0.6991150442477876
2	16	0	204	0	0	0	4	0.9107142857142857
3	28	4	0	176	0	0	8	0.8148148148148148
4	6	0	0	2	194	0	34	0.8220338983050848
5	6	6	12	0	2	68	88	0.37362637362637363
6	44	6	6	0	12	0	176	0.7213114754098361

The accuracy for Naive Bayesian Classifier in 8th Exp is :0.7606177606177607
The recall for Naive Bayesian Classifier in 8th Exp is :0.7986486486486486

	0	1	2	3	4	5	6	
0	202	6	6	4	0	0	6	0.9017857142857143
1	58	136	6	2	0	0	24	0.6017699115044248
2	8	0	210	4	0	0	0	0.9459459459459459
3	34	2	6	164	0	0	8	0.7663551401869159
4	2	0	2	0	224	0	8	0.9491525423728814
5	4	4	30	0	16	62	66	0.34065934065934067
6	32	4	12	14	12	0	170	0.6967213114754098

The accuracy for Naive Bayesian Classifier in 9th Exp is :0.7545219638242894
The recall for Naive Bayesian Classifier in 9th Exp is :0.7891891891891892

The average accuracy for Naive Bayesian Classifier in all Exps is :0.748551666286425
The average recall for Naive Bayesian Classifier in all Exps is :0.7852702702702702

(2) 十倍交叉验证评价结果表

表 4-2 基于 2012 年与 2016 年混合搜狗新闻语料库 Bayes 十倍交叉验证评价结果表

	P	R	F
1	0.7196	0.7527	0.735777926
2	0.74	0.777	0.758048780
3	0.725868	0.762162162	0.74357246
4	0.765463	0.8027	0.783639392
5	0.75515	0.79189	0.773083739
6	0.7329	0.77162	0.751761755
7	0.7693	0.806756	0.787582917
8	0.7619	0.8	0.780485306
9	0.76061	0.798648	0.779165033
10	0.7851	0.789189	0.787139190

(3) 十倍交叉验证结果图

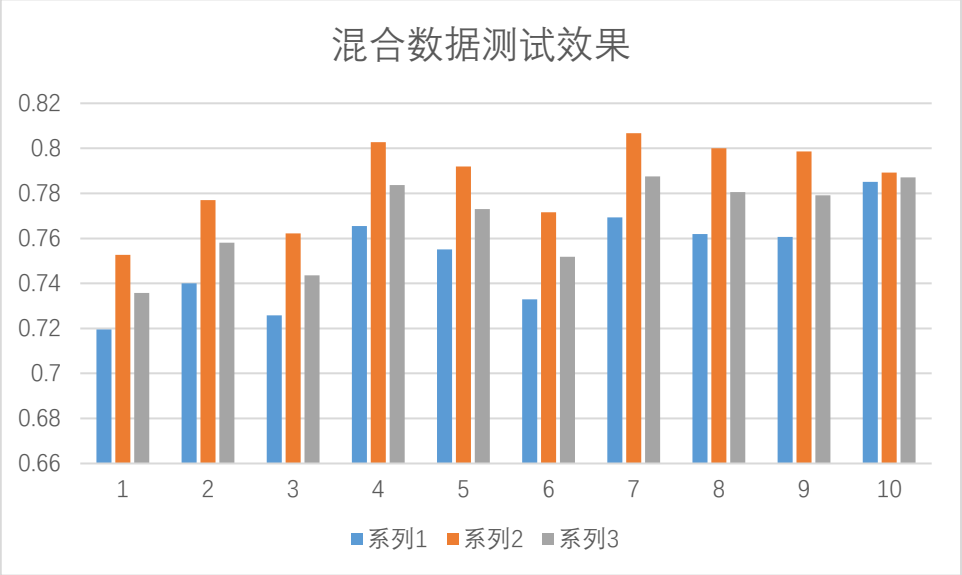

图 4-2 基于 2012 年与 2016 年混合搜狗新闻语料 Bayes 十倍交叉验证评价结果图

蓝色：P 橘黄：R 灰色：F

4.2.3. 两次结果对比总结

由以上十次交叉验证结果可得，Bayes 分类器在 2012 年新闻语料与 2012、2016 年混合新闻语料上分类表现都良好。但是在 2012 年新闻语料上表现优于在混合语料上表现，这说明短时间内的特征分布比长时间内的特征分布稳定。

4.3. 基于支持向量机(SVM)算法的新闻分类系统实验结果

4.3.1. 基于 2012 年搜狗新闻语料库实验结果

(1) 训练所得的模型文件的部分参数如下图：

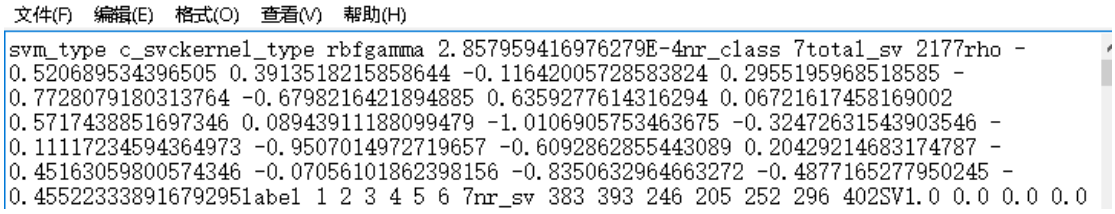

图 4-3 model 文件参数部分截图

(2) 十倍交叉验证评价结果表

表 4-3 基于 2012 年搜狗新闻语料库 SVM 十倍交叉验证评价结果表

	train_dataset	predict_dataset	precision	recall	F1
1	（0,,900）*7	（901,1000）*7	0.839719028	0.788571429	0.813341908
2	[(1,800)+(901,1000)]*7	（801,900）*7	0.884712096	0.851428571	0.867751295
3	[(1,700)+(801,1000)]*7	（701,800）*7	0.884158335	0.854285714	0.868965366
4	[(1,600)+(701,1000)]*7	（601,700）*7	0.878972158	0.831428571	0.854539586
5	[(1,500)+(601,1000)]*7	（501,600）*7	0.881828985	0.851428571	0.866362175
6	[(1,400)+(501,1000)]*7	（401,500）*7	0.902538441	0.865714286	0.883742929
7	[(1,300)+(401,1000)]*7	（301,400）*7	0.838324854	0.782857143	0.809642102
8	[(1,200)+(301,1000)]*7	（201,300）*7	0.838404636	0.771428571	0.803523356
9	[(1,100)+(201,1000)]*7	（101,200）*7	0.853607512	0.791428571	0.821342924
10	(101,1000)*7	（1,100）*7	0.87057102	0.814285714	0.841488217
A	——	——	0.858281536	0.795809524	0.825865803

(3) 十倍交叉验证结果图

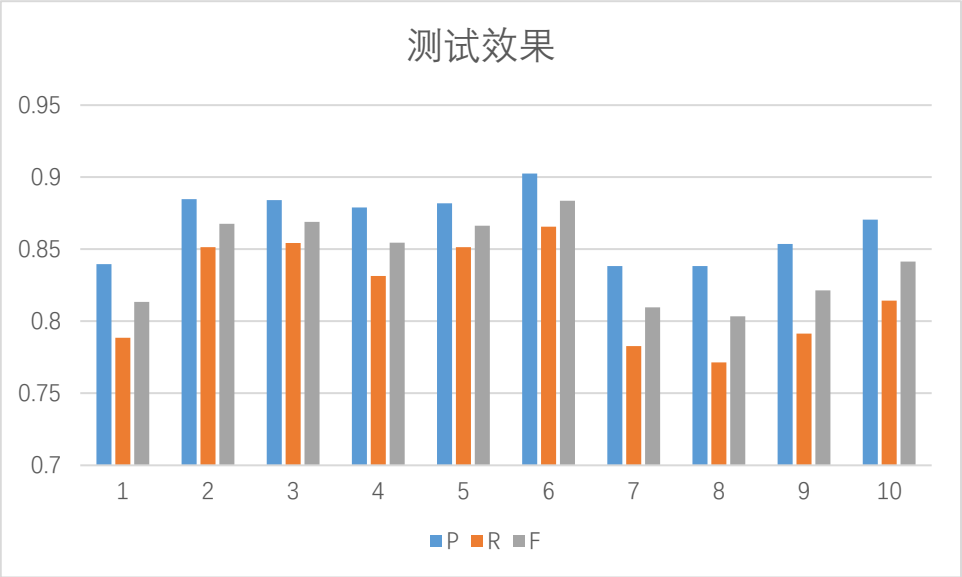

图 4-4 基于 2012 年与 2016 年混合搜狗新闻语料 SVM 十倍交叉验证评价结果图

由此可见，SVM 分类器在 2012 年语料库七类新闻分类上的效果表现良好，但是每次训练效果波动较大。

4.3.2. 基于 2012 年与 2016 年混合搜狗新闻语料库实验结果

(1) 十次交叉验证评价表

表 4-4 基于 2012 年与 2016 年混合搜狗新闻语料库 SVM 十倍交叉验证评价结果表

	train_dataset	predict_dataset	precision	recall	F1
1	（0,,900）*7	（901,1000）*7	0.817172606	0.7875	0.802061959
2	[(1,800)+(901,1000)]*7	（801,900）*7	0.82206584	0.764285714	0.792123506
3	[(1,700)+(801,1000)]*7	（701,800）*7	0.8311856	0.792857143	0.811569083
4	[(1,600)+(701,1000)]*7	（601,700）*7	0.82728169	0.7875	0.806900815
5	[(1,500)+(601,1000)]*7	（501,600）*7	0.82898095	0.792857143	0.810516747
6	[(1,400)+(501,1000)]*7	（401,500）*7	0.82741123	0.775	0.800348489
7	[(1,300)+(401,1000)]*7	（301,400）*7	0.83637285	0.796428571	0.815912119
8	[(1,200)+(301,1000)]*7	（201,300）*7	0.8351811	0.798214286	0.816279379
9	[(1,100)+(201,1000)]*7	（101,200）*7	0.84425438	0.780357143	0.811049197
10	(101,1000)*7	（1,100）*7	0.87057102	0.814285714	0.841488217
	——	——	0.83404773	0.78892857	0.810824951

(2) 十次交叉验证结果图

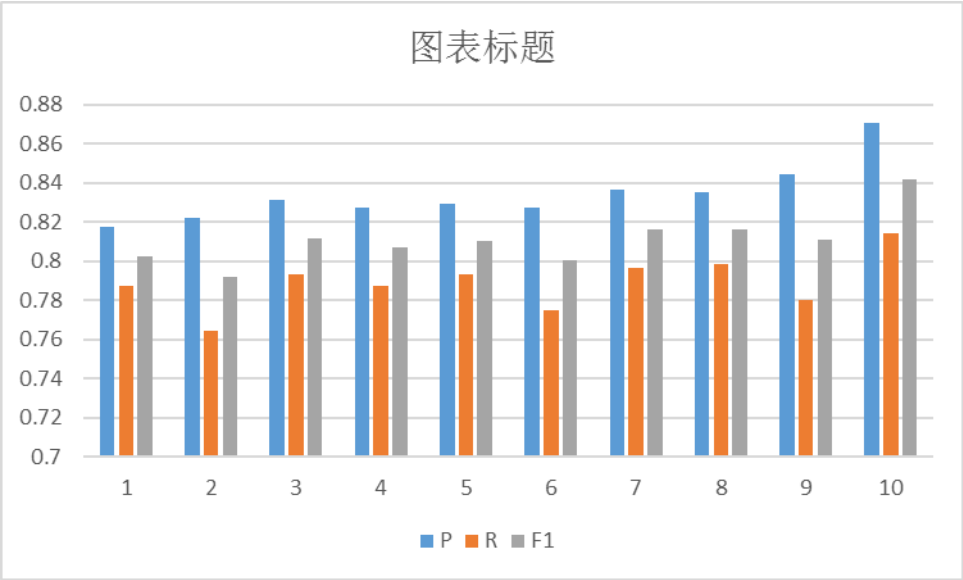

图 4-5 基于 2012 年与 2016 年混合搜狗新闻语料 SVM 十倍交叉验证评价结果图

4.3.3. 对比总结

由结果可得，SVM 新闻分类器在 2012 年语料和 2012、2016 年混合语料上分类结果都表现良好，但是在 2012 年语料上的结果略优于在混合语料上的结果。所以，得出与 Bayes 分类器测试一致的结论，说明短时间内的特征分布比长时间内的特征分布稳定。

4.4. 2012 年搜狗新闻语料训练结果与 2016 年新闻测试结果对比分析

在搜狗新闻网站中，按照类别分别爬取 100 个 URL 进行系统性能测试。

(1) Bayes 新闻分类器测试结果对比

表 4-5 Bayes 分类器测试结果表

NB	1	2	3	4	5	6	7
2012 语料库	0	0.06	0.69	0.42	0.64	0.19	0.02
混合语料库	0.69	0.86	0.74	0.34	0.83	0.34	0.2

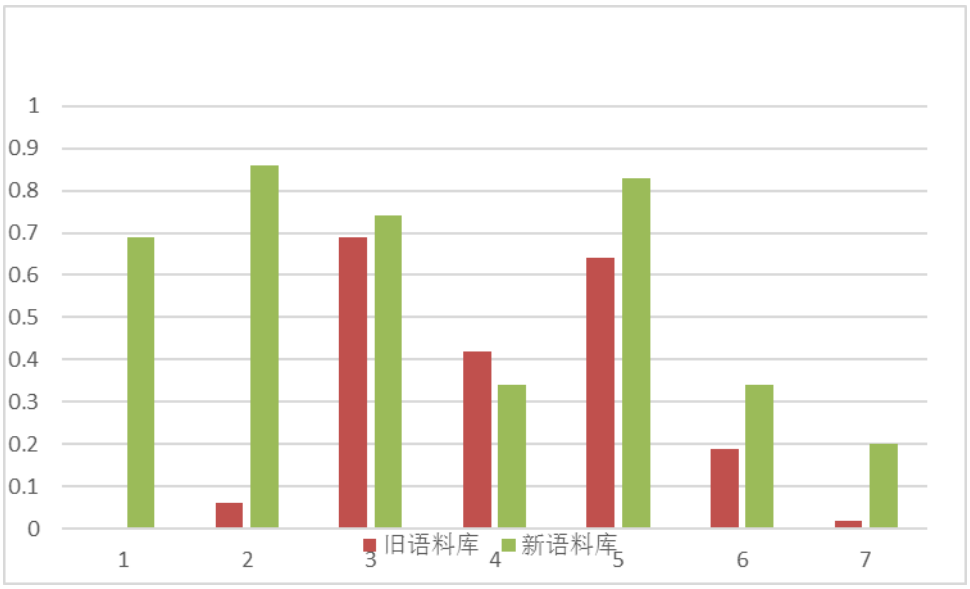

图 4-6 Bayes 分类器测试结果图

(2) SVM 新闻分类器测试结果对比

表 4-6 SVM 分类器测试结果表

SVM	1	2	3	4	5	6	7
2012 语料库	0.05	0.2	0.61	0	0.61	0.02	0.81
混合语料库	0.53	0.6	0.81	0.75	0.81	0.65	0.98

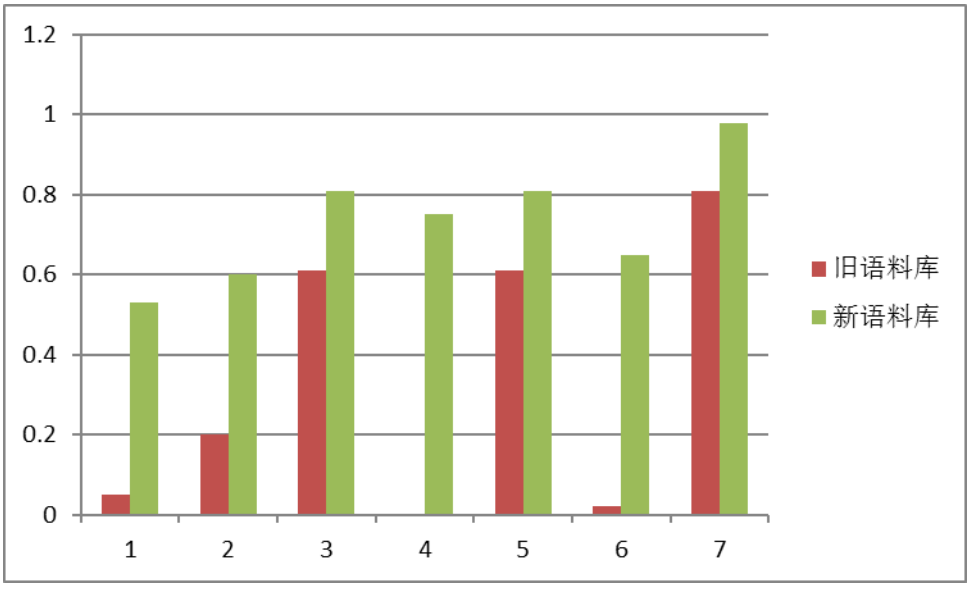

图 4-7 SVM 分类器测试结果图

(3) 结果分析

由以上测试结果看出，单独使用 2012 年新闻语料进行训练，在新新闻 URL

测试的结果是很差的，当混合不同年份新闻之后，SVM 与 Bayes 测试结果都有明显的提升。结合查阅资料分析，时序对于文本类别的特征分布影响很大，需要每过一个时间段对特征分布进行更新，这样才能保证系统性能。

对比 SVM 和 Bayes 分类器结果可以看出，SVM 分类性能整体优于 Bayes 分类性能。

此外，单独对比每一类别新闻的分类结果可以看出，在新闻种类比较混杂的“others”类上，SVM 的分类效果比 Bayes 分类效果好，这表明 SVM 分类方法对于特征的处理效果优于 Bayes，系统泛化性和稳定性好。

五、 创新点与问题总结

(1) 优化特征词的选取策略（剔除 IDF 值偏大明显的人名、地名等词）。

(2) 改进多项式模型的类条件概率的计算公式，改进为类条件概率：

$P(T_k|C)$=（类 C 下单词T_k在各个文档中出现过的次数之和+0.001）/（类 C 下单词总数+训练样本中不重复特征词总数

其中，当分子当T_k没有出现时，只加 0.001。

(3) 人工剔除分词结果小于 1KB 的新闻，来提高训练数据的泛化性和特征选取的准确度。

(4) 在文本分类系统中，通常都采用向量空间模型来描述文本信息。向量空间模型采用文本中的词条作为特征项，这就使其特征项可能涉及到整个文本集中的所有词条，导致了特征空间的维数非常之高,预处理后达到 7 万多维，且有许多特征对文本分类无关或者相关性不大。如果直接把这些特征作为分类器的输入进行训练，不仅使算法在计算上困难甚至不可行，也会影响到统计精度，从而降低分类器的推广能力和泛化能力，呈现“过学习”现象，所以特征降维是文本分类问题中的关键步骤之一。

(5) 在对特征向量加权处理中，TF-IDF 的值非常的小且集中，加权效果不佳，因此尝试使用了 TF-IDF 排序次序 N 与其次序中位数 X/2 的差的 10-2 倍作为每一特征词项维度的权重，得到了较好的实验效果。

(6) 本系统最大的问题是对于 2016 年新闻分类准确率低，原因是用于训练的搜狗新闻语料库是 2012 年以前的新闻，我们用 2016 年新闻进行测试，来对比时序对于新闻类别特征的影响。

六、 总结与展望

总结：

(1) 通过 10-fold cross 取平均得到 Bayes、SVM 新闻分类系统的分类结果，由实验结果可得，Bayes 与 SVM 分类效果良好，总体达到了新闻多类分类的目的。

(2) 将 2012 年以前的搜狗新闻语料库训练结果与 2016 年新闻语料测试结果进行对比，发现时序对于新闻类别特征的分布影响是比较大的。

(3) SVM 是基于二分类实现的分类器，在多分类问题上效果欠佳，可以考虑集成多种其他种类分类器处理多类别分类问题。

改进方案：

1) 未来通过改进的类条件概率估计值法和改进的判别函数来解决相互独立和查全率低的问题。

2) 需考虑数据集不平衡问题所导致的结果差异，并进行处理。同时需要考虑实验样本的大小对实验结果的影响。

3) 和别的数据挖掘方法相结合，如遗传算法等，进一步提高分类准确率和查全率。

4) 尝试贝叶斯的多项式模型等进行分类，用以比较并提高实验结果。

5) 尝试提取不同类别短信的不同文本特征，从而提高分类准确率。

6) 尝试一些非监督学习的分类模型，进行结果对比。

7) 对特征向量进行加权改进提高正确率。

8) 将卡方统计量的构造思想用于构造属性的相关性度量公式，以此改进贝叶斯算法。

9) 在后续系统改进中，增加训练集中新闻数量，增加训练集中近期新闻的数量，尽可能将训练数据的新闻的特征与测试数据新闻特征分布相同。

10) 贝叶斯与 SVM 有各自的优点和不足，在后续系统优化过程中可以考虑利用投票系统将两种分类器集成。

整个大作业过程不仅让我们小组成员学习到了推荐系统方面的相关知识，而且锻炼我们对问题的不断认识，同时加强了我们与其他人合作解决问题的能力。

七、 参考文献

[1] 代六玲， 黄河燕， 陈肇雄． 中文文本分类中特征抽取方法的比较研究[J].

中文信息学报，2004，18(1)：27-33.
[2] 周茜，赵明生，扈旻．中文文本分类中的特征选择研究[J]．中文信息学报，2004，18(3)：18-24.
[3] 孙晋文，肖建国．基于 SVM 文本分类中的关键词学习研究[J]．计算机科学，2006，33(11)：182-184.
[4] 余芳，姜云飞．一种基于朴素贝叶斯分类的特征选择方法[J]．中山大学学报：自然科学版，2004，43(5)：118-120.

小组成员及分工

左新宇　中科院自动化所 计算机与控制学院　201618014629039
新闻语料处理模块，新闻筛选，文本处理，特征选取，设计文档整合
于倩　中科院信工所　网络空间安全学院　2016E8018661108
SVM 新闻分类模块，SVM 部分设计文档书写
陈瑶　中科院信工所　网络空间安全学院　201628018627001
Bayes 新闻分类模块，Bayes 部分设计文档书写
梁小霞　中科院信工所　网络空间安全学院　2016E8018661092
网页界面模块及系统整合与测试，网页界面模块设计文档书写，汇报 ppt 制作
卓新旺　中科院信工所　网络空间安全学院　2016E8018661202
爬虫模块以及 2016 年新闻语料爬取，爬虫模块设计文档书写，汇报 ppt 制作

所有组员一起参与了新闻分类系统整体各个模块的确定、新闻语料搜集以及算法实过程中的交流等工作。