**基于树模型的网易云音乐情感分析**

小组成员： 田 也 3120181028

商瑞红 3220180732

王若琳 3120181040

王孟岚 3120181037

目 录

[1 概述 2](#_Toc8148711)

[2 相关工作 2](#_Toc8148712)

[2.1 情感分析的基本思路和技术概述 2](#_Toc8148713)

[2.2 使用情感词典及与其关联信息分析文本情感 3](#_Toc8148714)

[2.3 使用机器学习方法分析文本情感 4](#_Toc8148715)

[3 解决方法 5](#_Toc8148716)

[3.1 离线系统 6](#_Toc8148717)

[3.2 在线系统 7](#_Toc8148718)

[3.3 支持度和置信度 9](#_Toc8148719)

[3.4 FP-Tree算法 9](#_Toc8148720)

[4 系统架构与实现流程 14](#_Toc8148721)

[5 资源引用 16](#_Toc8148722)

[6 运行方法 17](#_Toc8148723)

[7 实验分析及结论 19](#_Toc8148724)

[7.1 实验原理分析 19](#_Toc8148725)

[7.2 实验意义分析 20](#_Toc8148726)

[7.3 实验结果分析 20](#_Toc8148727)

[8 系统实现难点及解决方案 22](#_Toc8148728)

[参考文献 24](#_Toc8148729)

# 1 概述

随着计算机和互联网技术的迅猛发展,网络己经成为人们获取信息的不可或缺的重要来源。自互联网进入时代以来,网民越来越习惯将网络作为自己表达观点、想法、态度的平台,而不只是被动的接受网站所发布的信息。由于大量的用户参与到信息的产生,网络信息的内容形式也变得越来越多样化,大量的具有个人观点性的内容充斥着网络。而这些观点对于电子商务、网络信息安全、网络舆情等方面具有非常重要的意义，因此情感分析（SA）的技术应运而生。

情感分析又称为倾向性分析和意见挖掘，它是对带有情感色彩的主观性文本进行分析、处理、归纳和推理的过程，其中情感分析还可以细分为情感极性（倾向）分析，情感程度分析，主客观分析等。

情感极性分析的目的是对文本进行褒义、贬义、中性的判断。在大多应用场景下，只分为两类。例如对于“喜爱”和“厌恶”这两个词，就属于不同的情感倾向。

情感程度分析主要是对同一情感极性中再进行划分或者细分，以描述该极性的强度。例如“喜爱”和“敬爱”都是褒义词，但是“敬爱”相对来说褒义的程度更加强烈一些。

# 2 相关工作

## 2.1 情感分析的基本思路和技术概述

**2.1.1 情感分析的对象和目标**

研究对象：网页中的文本数据，尤其是经过用户主动发布的，与发布对象直接相关联的文本。大致上可以分为针对词语、针对语句和针对篇章三个不同的粒度。

研究目标：识别出文本中所包含的主观性词语或句子，并对其情感趋势进行分析和判断。

**2.1.2 主流的研究思路和基本技术**

目前，在情感分析方面，主要使用的技术分为两大类：一类是采用情感词典的方法，助情感词典统计待分析文本中正向情感词和负向情感词的数目，根据他们的差值分析文本的情感极性；另一类是采用机器学习的方法，标注训练语料和测试语料，使用支持向量机、最大熵、KNN等分类器进行情感分类。文献[1]构建一个Twitter情感分析系统，能够实时地对有关总统选举的评论信息进行情感倾向性分析。文献[2]以词语的极性及其词性为特征，借助树内核模型对微博文本进行情感分类研究，并取得了一定的成果。文献[3]采用主题相关和无关的方式对微博文本进行情感极性分类，分为正向情感和负向情感。

机器学习的方法精确度更高，因为词典匹配会由于语义表达的丰富性而出现很大误差，而机器学习方法不会。而且它可使用的场景更多样。无论是主客观分类还是正负面情感分类，机器学习都可以完成任务。而无需像词典匹配那样要深入到词语、句子、语法这些层面。

而词典方法适用的语料范围更广，无论是手机、电脑这些商品，还是书评、影评这些语料，都可以适用。但机器学习则极度依赖语料，把手机语料训练出来的分类器拿去给书评分类，那是注定要失败的。

## 2.2 使用情感词典及与其关联信息分析文本情感

国外的文本情感分析研究开始于20世纪90年代末，早期的Riloff和Shepherd[4]对基于语料数据做了构建语义词典的相关研究。Hatzivassiloglou和McKeown[5]在大规模语料数据集上考虑了形容词的语义情感倾向性受连词的制约，尝试对英文的词语做情感倾向性判断。后来,越来越多的研究思路考虑了情感词或词组与特征词的依靠。Turney等[6]使用点互信息(Pointwise Mutual Information ,PMI)方法扩展基准的褒贬词汇，然后把极性语义(ISA)算法用于分析文本的情感，在处理通用语料数据时达到了74%。Tsou等[7]对词语的语义倾向性做计算,同时将极性元素分布、密度与语义强度对新闻语义倾向进行统计,以此衡量出大众对政治人物的评价。

中文方面，国内的徐琳宏、林鸿飞[8]从句子的词汇和结构作考虑，提取影响语句情感的9个语义特征，采用手工与自动获取相结合的方法，构建情感词汇本体库，对情感分析研究做了初步的尝试。李钝、曹付元等[9]从语言学的角度出发，采纳“情感倾向定义”权重优先的计算方式得到短语中词语语义倾向度，并分析词语的组合方式特点，提出中心词概念对词语的倾向性做计算，从而识别出短语的倾向性及其强度。该方法为更大粒度的文本情感分析打下了基础，具有一定的价值。

近年来，闻彬，何婷婷等[10]提出一种基于语义理解的文本情感分类方法，通过在情感词识别中引入情感义原，赋予概念情感语义，对概念的情感相似度重新定义，得到词语情感语义值。分析文本情感倾向性是否受语义层副词的出现规律的影响。该方法对有效地判定文本情感倾向性得到了一定的提高。赵妍妍等[11]提出了一种基于句法路径的情感评价单元自动识别方法,该方法自动获取句法路径来描述评价对象及其评价词语之间的修饰关系，并通过计算句法路径编辑距离来改进情感评价单元抽取的系统性能，在对电子产品领域的应用，取得了较好的实验效果。王素格等[12]提出了基于赋权粗糙隶属度的文本情感分类方法，该方法利用特征倾向强度，定义赋权粗糙隶属度，应用在真实汽车评论语料上，取得了不错的分类性能。

总体来看，使用情感词典及与其关联信息来分析文本情感，其优点是应用在词语特征级，句子级，粒度细，分析精准。但受到自然语言处理技术及相关抽取技术的限制，该方法容易丢失数据集中隐藏着的重要模式，使得未来研究工作中还有很大的提高空间。

## 2.3 使用机器学习方法分析文本情感

这类方法常用的机器学习模型有：中心向量分类法，朴素贝叶斯(Naive Bayes),最大熵(Maximum Entropy)，K最近邻分类和支持向量机(SVM)。在国外，Pang等[10,11]用机器学习的方法对电影评论进行情感极性分类，分为正向情感和负向情感，他分别采用了朴素贝叶斯、最大熵、支持向量机三种分类方法做实验，并将与他们手工分类结果做比较，发现支持向量机方法在这种机器学习方法中效果最好，分类精确度达到80%。Whitelaw等[12]通过提取电影评价信息中带形容词的词组，结合标准词袋特征表示，使用向量空间模型来表示文本，用支持向量机作分类，区分出带正面与负面评论信息，将准确率提高到了90.2%。Moens[14]用机器学习方法分别对法语、荷兰语、英语做情感分析实验， 结果显示三种语言的情感分类准确率分别达到68％、70％、83％。由此，可见机器学习方法在外文情感分析中展示出了一定的优势。

中文方面，文献[17-19]都是基于机器学习的方法分析文本情感，唐慧丰等[17]通过用名词、副词、形容词、动词做不同的文本表示特征，以信息增益、文档频率、CHI统计量和互信息做不同的特征选择方法，分别以中心向量法、贝叶斯分类、K最近邻和支持向量机做不同的文本分类方法做对比实验，其结果显示：在足够大的训练集与选择合适特征的情况下，采用n-Gram特征表示、信息增益特征选择和支持向量机分类方法，能取得较好的情感分类效果。夏火松等人[18]通过TF-IDF权重计算方法，使用基于RBF核函数的支持向量机方法的分类器，对携程网客户评论做分析，研究了停用词表在情感分类中的影响。乔向杰等[20]基于模糊推理方法，得到学生对学习事件的期望度推理，并运用贝叶斯网络对其构建的模型做实验，验证其模型的合理性。

纵观近期的研究发展，基于机器学习方法的情感分析关键在于特征信息的有效提取。优点是知识获取客观，准确性较高，缺点是对训练语料依赖性比较高，训练周期相对较长。总体来看，使用机器学习方法并不比运用情感词典及与其关联信息方法具有明显的优势。另外，针对中英文上的研究方法还有很多不同点，随着语义信息的加入以及训练语料集的发展，机器学习方法应该有较好的发展空间。

# 3 解决方法

现有方法大致可分为基于情感字典和基于机器学习的方法，这两种方法都或多或少有些不足。前者需要用到标注好的情感词典，英文的词典有很多，中文主要有知网整理的情感词典Hownet和台湾大学整理发布的NTUSD两个情感词典，还有哈工大信息检索研究室开源的《同义词词林》可以用于情感词典的扩充。词典质量的高低直接决定了情感分析效果的优劣，而且其中将带匹配字段与词典中感情词匹配时需要人为设置的权重也具备很强的经验性，因此整个系统需要具备很深语言信息处理素养的人才能完成。相比之下，基于机器学习的方法则更容易理解与实现。该方法需要大量的人工标注的语料作为训练集，通过提取文本特征，构建分类器来实现情感的分类。而如何定义与提取文本特征，则是决定方法效果的关键。目前已有相关的成熟方法，但相对体量比较大，而且基于机器学习的方法对不同的数据集需要训练不同的模型，同时数据量的多少与不同类别文本之间的平衡性对训练模型的影响很大。

考虑到基于词典方法中，词与词之间独立缺乏相关性以及权重难以设定的不足，以及基于机器学习方法会碰到数据需求量大和数据不平衡的问题，本文提出了基于树模型的无监督情感分类系统。



图1 系统结构图

本系统可以分为离线系统和在线系统两部分，如图1所示。

其中离线部分，利用收集的评论数据集，首先进行预处理，然后利用情感词典进行匹配，将数据集分为积极词汇集和消极词汇及，分别生成对应的频繁模式树（FP树），并且最终提取相应的积极频繁项集和消极频繁项集。

在线部分将评论单句作为输入，首先利用最小生成树将句子的分词合并成多个分组，并且分别与积极和消极频繁项集进行匹配，分别计算两种情感极性的分值与情感的偏向，最终实现情感分类。

## 3.1 离线系统

离线系统输入收集的评论数据集，对数据集进行分词，并标注词性，本系统仅使用动词，形容词和程度副词进行分析。根据词性剔除停用词和无关词，通过FP树构建算法对数据集进行处理，构建FP树并挖掘频繁项集，最终输出为频繁项集的词典。根据数据集构建积极与消极两个频繁项集。

|  |
| --- |
| def createTree(dataSet,minSup=1)  创建FP树 |
| 输入：数据集  输出：FP树根节点，头结点表 |

|  |
| --- |
| def mineTree(inTree,headerTable,minSup,preFix,freqItemList)  遍历FP树获得频繁项集 |
| 输入：FP树根节点，头结点表，最小项数，节点前缀，频繁项集 |

## 3.2 在线系统

在线系统输入待测试的单句，系统首先对输入的单句进行分词，应标注词性，剔除停用词与无关词项。根据离线系统构建的积极与消极的频繁项集对单句的分词序列进行分组，产生分组序列。系统再利用分组序列，根据设定的情感分值计算规则进行计算，分别获得积极与消极的分值。通过比较两个分值的大小与差，可表示单句的情感极性倾向于情感程度。最终输出情感的极性：积极或消极，以及情感的分值：0-10分。

|  |
| --- |
| def get\_sentence\_grade(sentence)  获取单句的情感分值 |
| 输入：string，测试用单句  输出：int，情感分值（0-10分） |

|  |
| --- |
| def get\_douban\_grade(sentence\_list)  获取多句情感分值 |
| 输入：list<string>，单句列表  输出：list<int>，每句分值  dict，多句中积极与消极比例  dict，各个分值的单句个数 |

双树模型分别指频繁模式树和最小生成树，分别用来解决词语相关性、数据需求与不平衡的问题和词语位置及相对重要性的问题。

频繁模式树（FP-Tree）是数据挖掘中用来挖掘频繁模式的一种算法模型。频繁模式（frequent pattern）是指频繁地出现在数据库中的模式，可以是项集，也可以是子序列或子结构。所谓项集，即项的集合。如果项集中包含K个项，则成为K项集。例如，集合{bicycle, lock}就是一个2项集。如果项集在数据库中出现的次数（称为支持度）超过某个设定的阈值（称为最小支持度），则称该项集为频繁项集。所谓子序列，序列具有了一定的先后顺序，比如相继购买相机、电脑、内存卡，如果这样的序列频繁地出现在购物历史数据库中，则称它为一个频繁的序列模式。一个子结构则具备诸如子图、子树或子格等更加丰富的结构形式。这样的子结构可能与项集，又或者子序列结合在一起。同样的，频繁的结构模式的定义即为一个频繁出现的子结构。对于挖掘数据之间的关联，发现这种频繁模式起着至关重要的作用。

关联规则（association rule）则是表示这些频繁模式的一种形式。例如，购买自行车也趋向于同时购买自行车锁，可以用以下关联规则表示：

Bicycle → lock [support=80%; confidence=70%]

规则兴趣度的度量分两种：支持度（support）和置信度（confidence）。上述规则的支持度为80%，意味着所分析的所有事物的80%显示自行车和车锁被同时购买。置信度70%意味着购买自行车的顾客70%也购买了锁。

一般而言，挖掘关联规则需要两个步骤：

(1) 找到所有的频繁项集：根据定义，这些项集中的每一个出现的次数至少与设定的最小支持计数一样。

(2) 由频繁项集产生强关联规则：根据定义，这些规则必须满足最小支持度和最小置信度。

挖掘关联规则的其中一种方法叫做频繁模式增长（Frequent-Pattern Growth, FP-growth），它可以挖掘全部频繁项集而无须多次扫描数据库，产生候选项集。

分治策略是FP-growth的核心：首先是构造，将数据库压缩成一棵频繁模式树（FP树）和保留项集关联信息的项头表，每个项通过一个结点链指向它在树中出现的位置。基本结构如图2所示。需要注意的是项头表需要按照支持度递减排序，在FP-Tree中高支持度的节点只能是低支持度节点的祖先节点。

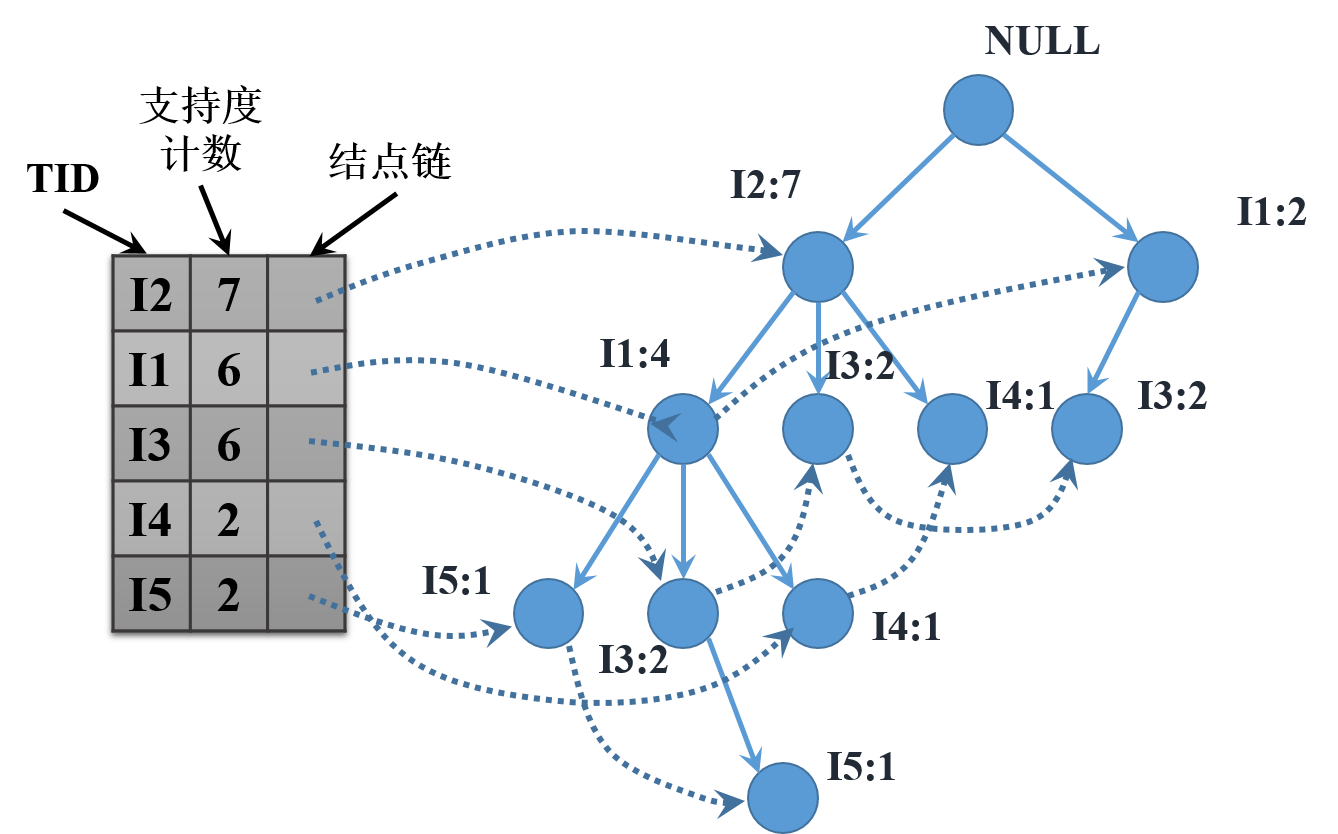


图2 FP-Tree的基本结构

构造之后是挖掘，把压缩后的数据库重新划分成一组条件数据库，在这里也可以叫做条件模式基。条件模式基即包含FP-Tree中与后缀模式一起出现的前缀路径的集合。每个数据库都会关联一个频繁项，又或者是一个模式段。对于每个模式片段，仅仅需要考察与它相关联的数据集即可完成挖掘过程。比如I1在该FP树中总共出现了2次，其祖先路径分别是{I2：7(频度为7)}，{NULL：2}。这2个祖先路径的集合就是频繁项I1的条件模式基。利用条件模式基来进行挖掘，可以明显地压缩被搜索的数据集的大小。

在关联规则挖掘领域最经典的算法是Apriori，其致命的缺点是需要多次扫描事务数据库。于是人们提出了各种裁剪（prune）数据集的方法以减少I/O开支，韩嘉炜老师的FP-Tree算法就是其中非常高效的一种。

## 3.3 支持度和置信度

严格地说Apriori和FP-Tree都是寻找频繁项集的算法，频繁项集就是所谓的“支持度”比较高的项集，下面解释一下支持度和置信度的概念。

设事务数据库为：*A　　E　　F　　G*

*A　　F　　G*

*A　　B　　E　　F　　G*

*E　　F　　G*

则{A,F,G}的支持度数为3，支持度为3/4。

{F,G}的支持度数为4，支持度为4/4。

{A}的支持度数为3，支持度为3/4。

{F,G}=>{A}的置信度为：{A,F,G}的支持度数 除以 {F,G}的支持度数，即3/4

{A}=>{F,G}的置信度为：{A,F,G}的支持度数 除以 {A}的支持度数，即3/3

强关联规则挖掘是在满足一定支持度的情况下寻找置信度达到阈值的所有模式。

## 3.4 FP-Tree算法

我们举个例子来详细讲解FP-Tree算法的完整实现。

事务数据库如下，一行表示一条购物记录：

*牛奶，鸡蛋，面包，薯片*

*鸡蛋，爆米花，薯片，啤酒*

*鸡蛋，面包，薯片*

*牛奶，鸡蛋，面包，爆米花，薯片，啤酒*

*牛奶，面包，啤酒*

*鸡蛋，面包，啤酒*

*牛奶，面包，薯片*

*牛奶，鸡蛋，面包，黄油，薯片*

*牛奶，鸡蛋，黄油，薯片*

我们的目的是要找出哪些商品总是相伴出现的，比如人们买薯片的时候通常也会买鸡蛋，则[薯片，鸡蛋]就是一条频繁模式（frequent pattern）。

FP-Tree算法第一步：扫描事务数据库，每项商品按频数递减排序，并删除频数小于最小支持度MinSup的商品。（第一次扫描数据库）

*薯片:7*

*鸡蛋:7*

*面包:7*

*牛奶:6*

*啤酒:4*

以上结果就是频繁1项集，记为F1。

第二步：对于每一条购买记录，按照F1中的顺序重新排序。（第二次也是最后一次扫描数据库）

*薯片,鸡蛋,面包,牛奶*

*薯片,鸡蛋,啤酒*

*薯片,鸡蛋,面包*

*薯片,鸡蛋,面包,牛奶,啤酒*

*面包,牛奶,啤酒*

*鸡蛋,面包,啤酒*

*薯片,面包,牛奶*

*薯片,鸡蛋,面包,牛奶*

*薯片,鸡蛋,牛奶*

第三步：把第二步得到的各条记录插入到FP-Tree中。刚开始时后缀模式为空。

插入第一条（薯片,鸡蛋,面包,牛奶）之后

插入第二条记录（薯片,鸡蛋,啤酒）

插入第三条记录（面包,牛奶,啤酒）

将第二步每行左边的称为表头项，树中相同名称的节点要链接起来，链表的第一个元素就是表头项里的元素。

如果FP-Tree为空（只含一个虚的root节点），则FP-Growth函数返回。

此时输出表头项的每一项+postModel，支持度为表头项中对应项的计数。

第四步：从FP-Tree中找出频繁项。

遍历表头项中的每一项（我们拿“牛奶：6”为例），对于各项都执行以下（1）到（5）的操作：

（1）从FP-Tree中找到所有的“牛奶”节点，向上遍历它的祖先节点，得到4条路径：

*薯片：7，鸡蛋：6，牛奶：1*

*薯片：7，鸡蛋：6，面包：4，牛奶：3*

*薯片：7，面包：1，牛奶：1*

*面包：1，牛奶：1*

对于每一条路径上的节点，其count都设置为牛奶的count

*薯片：1，鸡蛋：1，牛奶：1*

*薯片：3，鸡蛋：3，面包：3，牛奶：3*

*薯片：1，面包：1，牛奶：1*

*面包：1，牛奶：1*

因为每一项末尾都是牛奶，可以把牛奶去掉，得到条件模式基（ConditionalPattern Base,CPB），此时的后缀模式是：（牛奶）。

*薯片：1，鸡蛋：1*

*薯片：3，鸡蛋：3，面包：3*

*薯片：1，面包：1*

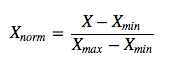
*面包：1*

（2）我们把上面的结果当作原始的事务数据库，返回到第3步，递归迭代运行。

下面是具体的算法描述。

|  |
| --- |
| 算法1 FP-growth |
| 输入：事务数据库D和最小支持度阈值minσ。  输出：D所对应的FP-tree。  方法：FP-tree构造  1. 扫描事务库D，获得D中所包含的全部频繁项集1F，及它们各自的支持度。对1F中的频繁项按其支持度降序排序得到L。  2. 创建FP-tree的根结点T，以“null”标记。再次扫描事务库。  3. 对于D中每个事务，将其中的频繁项选出并按L中的次序排序。设排序后的频繁项表为[p|P]，其中p是第一个频繁项，而P是剩余的频繁项。  4. 调用insert\_tree([p|P],T)。insert\_tree([p|P],T)过程执行情况如下：  5. 如果T有子女N使N .item\_name=p.item\_name，则N的计数增加1；  6. 否则创建一个新结点N，将其计数设置为1，链接到它的父结点T，并且通过node\_link将其链接到具有相同item\_name的结点。  7. 如果P非空，递归地调用insert\_tree(P，N)。  8. 结束 |

挖掘完频繁模式后，还需要进行数据标准化处理来消除数据不平衡的影响。数据标准化（归一化）处理是数据挖掘的一项基础工作，不同评价指标往往具有不同的量纲和量纲单位，这样的情况会影响到数据分析的结果，为了消除指标之间的量纲影响，需要进行数据标准化处理，以解决数据指标之间的可比性。原始数据经过数据标准化处理后，各指标处于同一数量级，适合进行综合对比评价。本文采用的是Min-Max Normalization，也称为离差标准化，是对原始数据的线性变换，使结果值映射到[0 - 1]之间。转换函数如下：



该方法实现对原始数据的等比例缩放，其中Xnorm为归一化后的数据，X为原始数据，Xmax、Xmin分别为原始数据集的最大值和最小值。

文本利用FP-Tree来挖掘本文中词语的相关性，找到其中的频繁项集并组合成关联规则。与传统方法相比，该方法既不像词典方法那样需要深厚的语言处理背景知识，也不像单词表示为特征向量那样难以解释，这是一个更为直观的模型。同时就实现而言，本文方法操作简便，实现过程中与基于机器学习的方法一样，一次调用模型即可，不必像词典那样，一个个单词匹配。

最小生成树模型，则是考虑到了词与词之间的距离信息。举个例子，在前半句话中出现的程度副词“非常”，就不该对后半句的形容词“好”有影响或者影响极小，所以我们需要对词与词之间的位置信息进行判断，判断的方法最常见的就是用滑动窗口，设置不同的大小分别去滑动，这是非常繁琐且浪费时间的。因此本文参考最小生成树模型，将运用场景从图变化到文本序列中，常用的最小生成树算法有Prim算法和Kruskal算法，本文采用的是Kruskal算法，也可认为是贪心的思想，且借鉴了并查集的数据结构，将分词合并成多个分组。对每个分组在FP树构成的频繁项集中进行查询。

将分词合并为分组的目的在于，从被分析单句中寻找出现的频繁项集，如果某个频繁项集出现在句中，可以认为单句更倾向于某种情感。在句中寻找频繁项集时，还应注意的问题是，需要考虑词之间距离的影响，相距较远的两个词即使出现在同一频繁项集中，也并没有情感上的联系。

在合并项集的过程中，我们首先观察当前词项在FP树频繁项集中出现的频度，选取当前频度最大的词项作为被选项开始合并。在FP树中，频度越大的词项有更大概率是某个频繁项集的子项。我们观察与被选项距离两个词以内的项，如果加入该词项后，新的词项也在FP树频繁项集中出现，则认为可合并，合并该词项，产生新的词项序列。

考虑如下情况：

*{很，棒，正品，保暖性，很，棒，必须，五星，好评}*

对应的频度如下：

*{0.58, 0.17, 0.05, 0, 0.58, 0.17, 0, 0.39, 0.41}*

首先观察频度最大的词项，为0.58。再观察该项距离为2以内的项，合并后是否存在于FP树的频繁项集内。如果存在，则作为新的词项更新入序列。查询“（很，棒）”存在于频繁项集内，更新序列：

*{（很，棒），正品，保暖性，很，棒，必须，五星，好评}*

新的词项频度更新为频繁项集中的频度：

*{0.26, 0.05, 0, 0.58, 0.17, 0, 0.39, 0.41}*

重复以上步骤，继续以频度最大项作为备选项，合并词项。当序列中所有频度均低于设定的阈值，或没有项集可合并时，停止操作，获得最终的词项分组与频度序列。例如：

*{（很，棒），正品，保暖性，（很，棒），必须，（五星，好评）}*

*{0.26, 0.05, 0, 0.26, 0, 0.2}*

使用贪心方法与并查集的数据结构，与滑动窗口相比，可以快速的合并序列中的频繁项，算法复杂度更低。

|  |
| --- |
| 算法2 合并词项 |
| 输入：分词序列L，频度阈值m。  输出：合并频繁项集后的分组序列LG。  方法：贪心法合并词项分组  1. 扫描分词序列L，确认频度是否大于阈值，若均小于阈值则结束。  2. 找到序列中频度最大项作为被选项。  3. 观察与被选项距离为2以内的词项，查询词项与被选项合并后是否存在于频繁项集中。  4. 若频繁项集中存在，则合并作为新的词项更新到序列中，频度更新为新词项在频繁项集中的频度。  5. 若频繁项集中不存在，不做任何操作。  6. 重复步骤1-5。  7. 结束 |

# 4 系统架构与实现流程

本系统使用python语言实现，以网站形式呈现，网站使用django架构搭建。系统的整体架构如图3所示，具体分为评论爬取模块、分词模块、分析模块、极性分值模块、前端展示模块及双树模块。其中，评论爬取模块为情感分析提供原始资源，每条评论以字符串形式存在，多条评论组合成一个列表；分词模块将每一条评论拆分为单个词，并标注词性；针对单条评论和多条评论，分别使用单项评论分析模块和整组评论分析模块；频繁模式树和最小生成树为评论分析提供支持；极性、分值模块计算、存储每一条评论所得到的分值，并判断整组评论的极性为积极或是消极；前端展示模块接受服务器传送的数据，同时将处理得到的用户情感分析结果以可视化的方式展示在界面上。



图3 系统架构

系统完整运行流程如图4所示，系统首先获取到前端传输的内容并进行解析，如果是网易云音乐条目页面URL，则解析出对应的评论存储页面(Ajax)，并获取到指定数量的评论条目。由于部分评论存在无用字符，因此需要对所有评论进行筛选，留下有价值的评论以便进行分析。因为分析是基于单句的，因此针对整组评论会进入循环流程，将每一条评论进行分词，并标注词性，同时获取到单句的频繁项集，根据频繁项集对评论进行打分。如果所有评论都已打分完成，则将评论、极性、分值组合成前端所需要的格式（json），并返回给前端。



图4 系统运行流程

其中的评论获取模块流程如图5所示，使用爬虫技术。在选择爬取内容时，考虑到携程、淘宝、天猫以及京东的商品评论无法直接获取到，都使用了ajax加密技术，需要从另外的页面中读取评论数据。虽然经过对网络流量进行监控，我们获取到评论的真实存储地址，并尝试进行爬取，但他们使用了第二重防爬虫策略：动态cookie，每次向评论存储地址发出请求时，浏览器都会传送一个cookie，服务器进行检测，通过检测后才会返回数据，否则只能访问到拒绝访问的json返回值。同时由于cookie是动态变化的，无法进行预测，爬虫程序的编写难度较大。因此通过小组讨论，我们决定改为对网易云音乐评论进行单句分析。具体的实现方式是对前端输入的网易云音乐URL进行解析，若解析不成功则进入错误提示页面，解析成功则生成对应的短评页面url。网易云音乐的短评数量多则上万条，为了减轻系统压力，网易云爬虫爬取最新的前200条短评，并生成短评列表返回。



图5 评论爬取模块运行流程

# 5 资源引用

(1) 客户端操作系统：Windows/Mac OS/Linux

(2) 浏览器：chrome v50.0以上/Firefox v42.0以上

(3) 系统服务器：python 2.7.10+

(4) 使用库资源： django 1.11

BeautifulSoup4 4.6.0

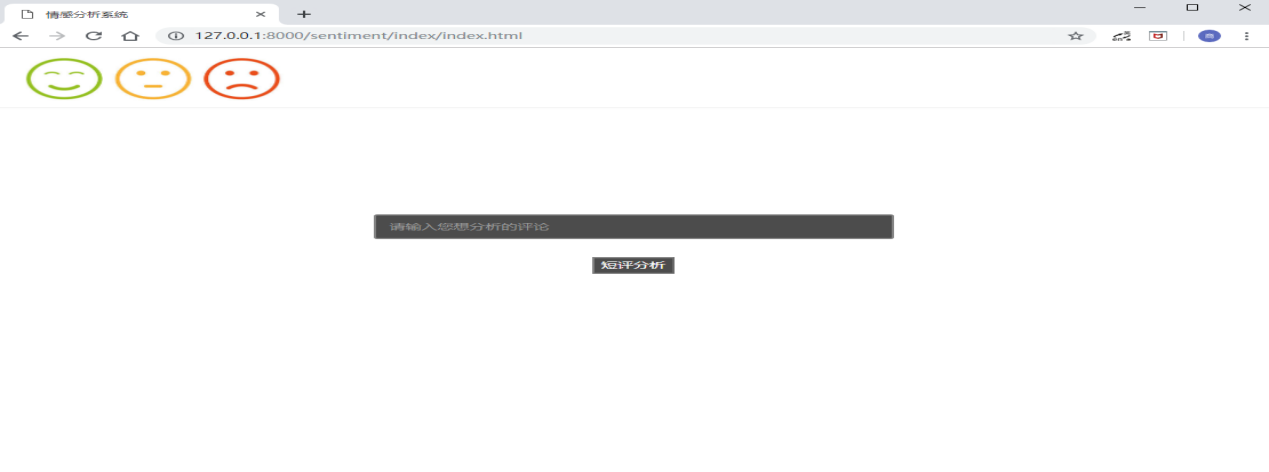
Jieba 0.39

Requests 2.18.4

(5) 程度副词词典，来自互联网

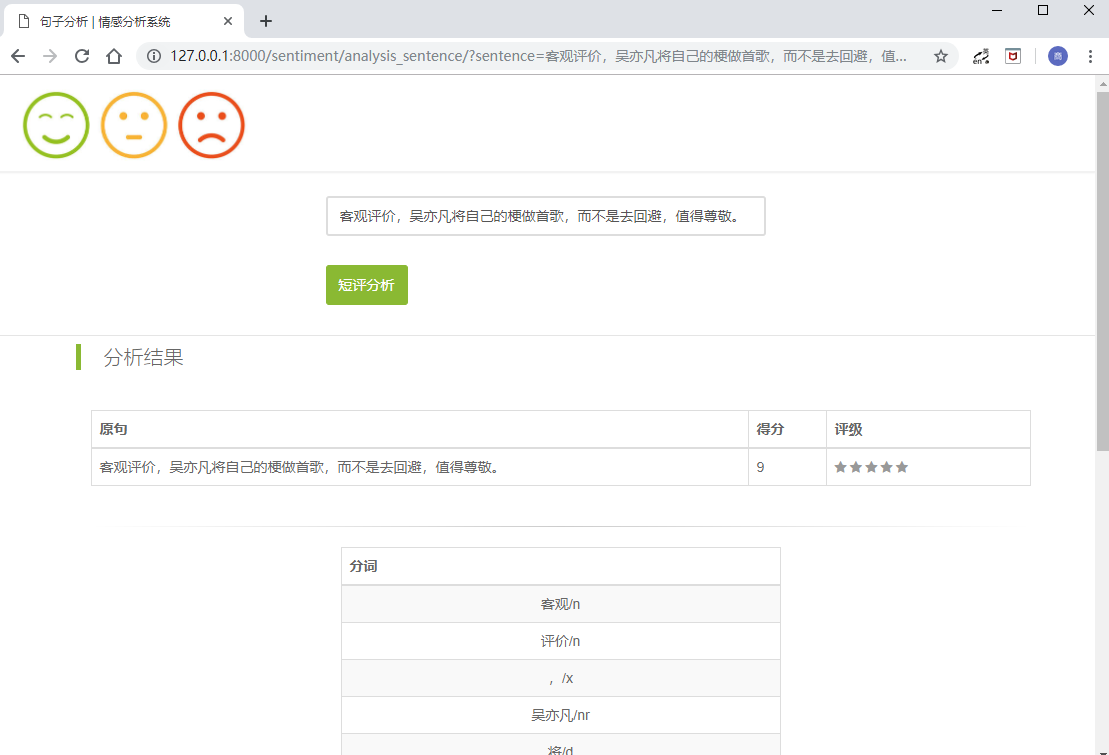
# 6 运行方法

运行PowerShell (Windows)或Terminal (macOS/Linux)，进入项目根文件夹，输入python manage.py runserver 8000 ，运行。在浏览器中输入：http://127.0.0.1:8000/sentiment/index/ ，即可看到系统主页面。

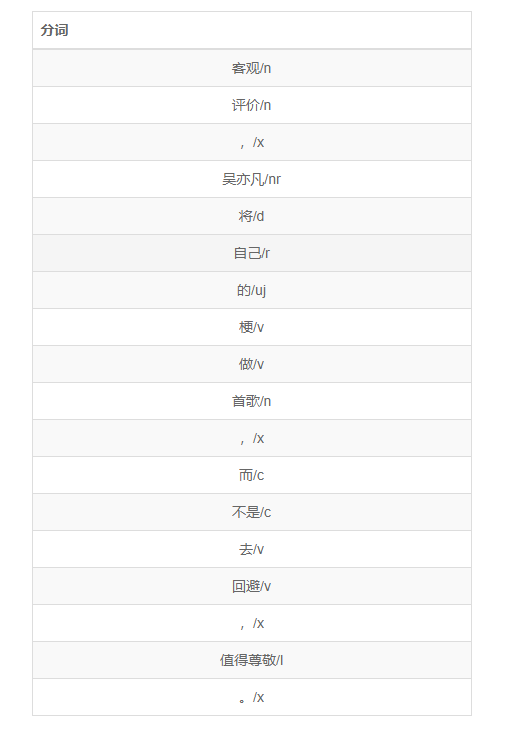


我们的系统Emotional Analysis实现了对网易云音乐短评的单句情感分析。在单句分析的输入框中输入要分析的句子，然后点击短评分析按钮，就会调转到单句分析结果的页面，如下图所示：

单句分析结果页面上面是单句分析的input输入框和button按钮，我们可以在这继续输入我们想要分析的句子，点击button继续分析。下面是句子的分析结果，展示出了原句和得分以及对句子进行分词的一个详细信息。



下图是音乐短评分词得到的分词结果的具体表示：



成功获得分词后，利用FP-Tree算法进一步生成单句的频繁项集，根据这些频繁项集对此评论进行打分，最后后端将打分结果以数值的形式传递给前端，上图分析结果表格中的得分一栏就是最终的情感分析结果可视化表示。

# 7 实验分析及结论

## 7.1 实验原理分析

本次实验将关联规则与情感分析进行了融合。关联规则最初提出的动机是针对购物篮分析问题提出的。假设分店经理想更多的了解顾客的购物习惯。特别是，想知道哪些商品顾客可能会在一次购物时同时购买？为回答该问题，可以对商店的顾客事物零售数量进行购物篮分析。该过程通过发现顾客放入“购物篮”中的不同商品之间的关联，分析顾客的购物习惯。这种关联的发现可以帮助零售商了解哪些商品频繁的被顾客同时购买，从而帮助他们开发更好的营销策略。关联规则挖掘过程主要包含两个阶段：第一阶段必须先从资料集合中找出所有的高频项目组，第二阶段再由这些高频项目组中产生关联规则。

关联规则挖掘的第一阶段必须从原始资料集合中，找出所有高频项目组。高频的意思是指某一项目组出现的频率相对于所有记录而言，必须达到某一水平。一项目组出现的频率称为支持度，以一个包含A与B两个项目的2-itemset为例，我们可以求得包含{A,B}项目组的支持度，若支持度大于等于所设定的最小支持度门槛值时，则{A,B}称为高频项目组。一个满足最小支持度的k-itemset，则称为高频k-项目组(Frequent k-itemset)，一般表示为Large k或Frequent k。算法并从Large k的项目组中再产生Large k+1，直到无法再找到更长的高频项目组为止。

关联规则挖掘的第二阶段是要产生关联规则(Association Rules)。从高频项目组产生关联规则，是利用前一步骤的高频k-项目组来产生规则，在最小信赖度(Minimum Confidence)的条件门槛下，若一规则所求得的信赖度满足最小信赖度，称此规则为关联规则。

而文本情感分析，是对带有情感色彩的主观性文本进行分析、处理、归纳和推理的过程。互联网(如博客和论坛以及社会服务网络)上产生了大量的用户参与的、对于诸如人物、事件、产品等有价值的评论信息。这些评论信息表达了人们的各种情感色彩和情感倾向性,如喜、怒、哀、乐和批评、赞扬等。基于此,潜在的用户就可以通过浏览这些主观色彩的评论来了解大众舆论对于某一事件或产品的看法。

我们可以直观的进行了解，情感分析是根据语句中的词的情感倾向进行分类及判断，而词语倾向的分类是基于词语的相关性的方法。关联分析是属于数据挖掘的核心技术，可以在统计上发现数据间的潜在联系。如果将两者进行融合，那我们既可以通过数据挖掘相关算法得到大量的评论中潜在的关联，又可以从这些关联中获得文本情感分析的基础。

## 7.2 实验意义分析

文本情感分析在实际中具有重要的实用价值。对用户舆情进行情感分析，主要是分析具有情感成分的词汇的情感极性（即情感的正性、中性、负性）和情感强烈程度，然后计算出每个语句的总值，判定其情感类别。还可以综合全文本中所有语句，判定总舆情数据样本的整体态度和情感倾向。

情感分析的优点如下所示：

（1）可扩展性：如果对成千上万的推文，用户对话或用户评论手动排序，数据量极大，既耗时又耗费人力。情感分析允许以有效且低成本的方式大规模处理数据。

（2）实时分析：我们可以使用情感分析来识别关键信息，以便在特定情景中实时提供态势感知。 情感分析系统可以帮助您立即识别社交媒体公关危机等情况并采取行动。

（3）一致的标准：人类没有观察到评估一段文字情感的清晰标准。据估计，在判断特定文本的情感时，不同的人仅仅在60-65％的情况下一致。这是一项主观任务，深受个人经历，思想和信仰的影响。 通过使用集中情感分析系统，公司可以对所有数据应用相同的标准。 这有助于减少错误并提高数据一致性。

通过本系统对网易云音乐评价进行的文本情感分析，我们能直观的获得对这个音乐作品综合性的评价，而不用人工一条条进行标记、分析。这种分析快速、高效。尤其适合大量、海量样本情况。

## 7.3 实验结果分析

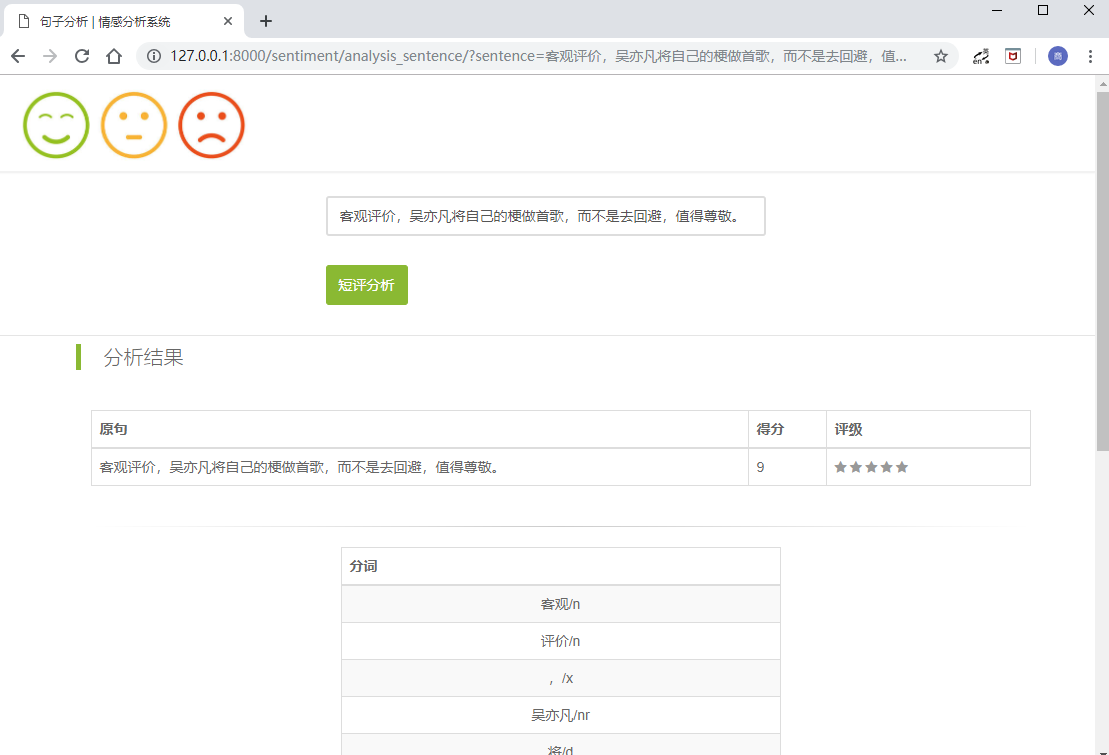
我们设计的系统在自行爬取的网易云音乐评论数据集和进行了实验，前人对该语料库使用了多种方法进行了测试，经过对各类文献的阅读，我们将其实验数据列表如下：

表1 语料库测试结果

|  |  |  |  |
| --- | --- | --- | --- |
| 语料库 | 分类方法 | 分类准确性 | 参考文献 |
| ChnSentiCorp-Htl-ba-2000 | SVM(BOW) | 0.827 | 1 |
| ChnSentiCorp-Htl-ba-2000 | SVM(word2vec) | 0.880 | 2 |
| ChnSentiCorp-Htl-ba-2000 | Naive bayes | 0.844 | 1 |
| ChnSentiCorp-Htl-ba-2000 | Logic regression | 0.856 | 1 |
| ChnSentiCorp-Htl-ba-2000 | 语义词典结合 | 0.776 | 3 |

可以看见，所有的在该语料库上的分类方法的准确性都在0.8左右，同时，机器学习的方法如支持向量机（svm）和朴素贝叶斯的方法准确率都比较高，而语义词典结合的方法在参考文献3中的方案准确率只有0.776，远低于机器学习的方案。我们自己设计的double tree分类方法，我们的准确率提高到了0.819。接近了机器学习的方法，如果结合情感词典后，相信准确率还会有提升。

同时，我们还根据double tree的方法，构建了网易云音乐评价的处理模型，通过对我们收集的语料的分析，构建double tree,然后对新的评价进行处理，我们发现，可以取得不错的效果。



上图显示了我们对评价的结果，针对每一条评价，我们可以从0到10对其进行评分，实现了对情绪程度的评价。

综上，该算法有以下优点：首先该方法比传统的基于语义词典的方法在准确率上有较大的提升；其次，double tree的方法考虑到了词语与词语之间的关联性，权重可靠，相比与机器学习的方法，我们的方法更为显性；同时考虑到机器学习的方法需要大量的平衡的标签数据，比较大的计算资源，速度较慢，我们的方法实现速度快，方法简单。最后，我们最后给出的程序不光可以给出正负情绪评价的分类，还可以对情绪的程度进行评分，同时我们的程序可以对单条评价进行处理，还可以稍加扩展对批量评价进行处理，并给出统计结果，具有较强的实用性和可扩展性。

# 8 系统实现难点及解决方案

1、 框架和开发语言的选择：

因为大部分同学都对python比较熟悉，所以我们选择python作为开发语言，框架选择django，django是一个Python定制框架，它鼓励快速开发,并遵循MVC设计，它是开源框架，有完美的文档支持和完整的路由系统，开发简便，易于理解。

2、 算法选择：

确定开发语言和框架后，就是情感分析算法的选取，开始想选择基于统计的机器学习方法，用有监督的（需要人工标注类别）机器学习方法来对文本进行分类，但这需要大量的人工标注，我们系统是基于网易云音乐进行情感分析，和单句进行情感分析，目前没有很好的现成的基于网易云的音乐标注数据集，如果不基于网易云音乐标注，而且采用其他的已标注数据集，用这样的数据集运行网易云短评情感分析效果不会很好。

而采用无监督的机器学习方法，这点与词典匹配有着本质的区别。词典匹配是直接计算文本中的情感词，得出它们的情感倾向分值。而机器学习方法的思路是先选出一部分表达积极情感的文本和一部分表达消极情感的文本，用机器学习方法进行训练，获得一个情感分类器。再通过这个情感分类器对所有文本进行积极和消极的二分分类。最终的分类可以为文本给出0或1这样的类别，也可以给出一个概率值，比如”这个文本的积极概率是90%，消极概率是10%“，而我们想给出更多的情感分析类别和不同的情感值。

调研了一些算法，例如LSTM+CRF、决策树、朴素贝叶斯分类等，最终我们决定选择基于规则的FP树算法来实现单句的情感分析功能，这个算法能保持数据完整性:它不会打破任何事物数据中的长模式，而且为频繁模式的挖掘保留了完整的信息。也有紧凑性的优点，减少了不相关的信息，非频繁的项会被删除，按频率递减排列，相似的更频繁的项更容易在树中被共享。因为开始对这个算法不是很熟悉，所以在集成算法的时候遇到了一些问题，最后在github上搜到了相关的算法，通过阅读论文、查阅资料了解算法，分析算法的实现和功能，最终调试通过。

3、 测试集获取：

无法直接获得公开的适合于本系统模型的测试集，为了保证系统训练集的完整性，同时将系统模型更好地与其他算法作比较，因此在本系统中我们并没有直接从训练集中抽取部分数据用于测试，而是编写爬虫程序，通过前端输入的网易云音乐URL实时爬取影评作为测试数据输入。

用户在前端输入某个音乐的url，调用评论爬取模块，首先通过requests库的requests.get()方法获得输入url的response，然后利用BeatifulSoup对response进行解析，包括解析到要访问资源的协议、主机IP地址、以及资源在主机中的具体地址，解析成功就能生成对应短评页面地址，否则进入错误提示页面，重新输入url。解析成功后根据得到的页面地址读取网页中的HTML文档，通过BeatifulSoup对象的find()和findAll()解析HTML文档中的标签，解析到短评标签就可爬取标签内的短评内容。由于网易云音乐的短评数量有些多达上万条，为了减轻系统压力，我们只爬取每首歌曲中的前200条短评作为测试数据，生成短评列表与系统模型分析得到的结果一起返回到用户页面。

除此之外，我们还考虑过爬取其他网站的评论，但是由于其他网站的评论部分采用了ajax加密，因此我们在网易云中实现的爬虫方案并不能爬取到数据，于是我们了解了通过ajax异步加载的方式获取数据的方法，在这个过程中，客户端浏览器会异步发送一个新的请求给服务器，可以通过分析这个异步访问请求，获得浏览器请求资源的真实url，然后解析这个url爬取数据，同时这种方式获取的是json格式的数据，更易于处理。

实现过程中我们还遇到了很多细节问题，上面是比较典型的例子，这些细节问题如html界面、javascript脚本语言、css样式调试等，最后通过查资料反复调试实现了整个系统的调通。

# 参考文献

[1] Wang H, Can D, Kazemzadeh A, et al. A system for real-time Twitter sentiment analysis of 2012 U.S. Presidential election cycle[C]// ACL 2012 System Demonstrations. 2012:115-120.

[2] Apoorv Agarwal. Sentiment Analysis of Twitter Data// R Pssonneau - 2011

[3] Jiang L, Yu M, Zhou M, et al. Target-dependent Twitter sentiment classification[C]// Meeting of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2011:151-160.

[4] Riloff E, Shepherd J. A Corpus-Based Approach for Building Semantic Lexicons[J]. Computer Science, 2012:117--124.

[5] Hatzivassiloglou V, Mckeown K R. Predicting the semantic orientation of adjectives[C]// Eighth Conference on European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, 1997:174-181.

[6] Turney P D, Littman M L. Measuring praise and criticism:Inference of semantic orientation from association[J]. Acm Transactions on Information Systems, 2003, 21(4):315-346.

[7] Tsou B K Y, Yuen R W M, Kwong O Y, et al. Polarity Classification of Celebrity Coverage in the Chinese Press[J]. 2005.

[8] 徐琳宏, 林鸿飞. 基于语义特征和本体的语篇情感计算[J]. 计算机研究与发展, 2007, 44(s2):356-360.

[9] 李钝, 曹付元, 曹元大,等. 基于短语模式的文本情感分类研究[J]. 计算机科学, 2008, 35(4):132-134.

[10] 闻彬, 何婷婷, 罗乐,等. 基于语义理解的文本情感分类方法研究[J]. 计算机科学, 2010, 37(6):261-264.

[11] 赵妍妍, 秦兵, 车万翔,等. 基于句法路径的情感评价单元识别[J]. 软件学报, 2011, 22(5):887-898.

[12] 王素格, 李德玉, 魏英杰. 基于赋权粗糙隶属度的文本情感分类方法[J]. 计算机研究与发展, 2011, 48(5):855-861.

[13] Wawre S V, Deshmukh S N. Sentiment classification using machine learning techniques[J]. Int. J. Sci. Res, 2016, 5(4): 1-3.

[14] Bo P, Lee L. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales[J]. 2005:115-124.

[15] Whitelaw C, Garg N, Argamon S. Using appraisal groups for sentiment analysis[J]. 2005, 9(1):625-631.

[16] Boiy E, Moens M F. A machine learning approach to sentiment analysis in multilingual Web texts[J]. Information Retrieval, 2009, 12(5):526-558.

[17] 唐慧丰, 谭松波, 程学旗. 基于监督学习的中文情感分类技术比较研究[J]. 中文信息学报, 2007, 21(6):88-94.

[18] 夏火松, 陶敏, 王一,等. 停用词表对基于SVM的中文文本情感分类的影响[J]. 情报学报, 2011, 30(4):347-352.

[19] 乔向杰, 王志良, 王万森. 基于OCC模型的E-learning系统情感建模[J]. 计算机科学, 2010, 37(5):214-218.

[20] Dual Sentiment Analysis: Consideri Two Sides of One Review. Rui Xia, Feng Xu, Chengqing Zong, Qianmu Li, Yong Qi, and Tao Li

[21] 利用Python实现中文情感极性分析 atlee http://www.jianshu.com/p/233da896226a

[22] 基于网络信息文本倾向性分析的领域应用研究 俞飞 华东师范大学