**【数据预处理】**

1. 去除所有表情符号 -> 首先**弄清楚表情符号的编码方式**
2. 删除所有空格
3. 删除所有标点符号（除了**“+”**,待定）
4. 删除所有特殊符号
5. 转换成简体
6. 检测到全角的数字，直接屏蔽（必须在删除标点符号之后），提示玩家“消息不符合发言规则”

如“正规渠道50送１００万元宝+ⅥP+银币+１３６＋５０８＋３２２到付。ctBWKn“”

以上1~4点基于以下理由：

1. 有的垃圾信息会故意以空格、标点符号、特殊符号、表情等来隔断可能被检测到的关键垃圾词汇；
2. 由于手误等原因，会有空格、表情等无意义字符插入到正常的文本中，删除这些有利于提取特征；
3. 聊天信息的长度一般非常短，因此要尽量提取到尽可能多的特征再进行筛选，避免漏掉一些可能会对分类有影响的特征。所以删去以上符号，使文本连在一起后再进行特征提取会更有利。

**本项目可使用：**

（1）特征单元：Bigram与分词相结合，两个特征集进行合并。若两个特征集中有相同的特征，次数不累加。

（2）特征选取方法：

* 用文本频率DF >= 2进行截断
* Chi-square/信息增益IG/互信息MI等方法选取前k个最具代表性的特征

（3）特征向量权重：使用TF-IDF表示；权重向量要进行规格化

**【文献中的经验】->可考虑**

* 将所有的空格、标点符号、特殊字符都删除
* **增加一个特征：文本中的数字串个数 或 文本中是否有数字串**。 考虑到垃圾信息中可能会含有qq号、群号等等，这个特征有一定价值。
* 注意到很多垃圾消息中包含“+”号，是否可以**将“+”号的数量也作为一个特征**呢。如果这么做，那么数据预处理的时候就要保留“+”号了。

例子：“50元给50万元宝+1亿银+V12+橙装+橙将Q272-387-060先货iN”

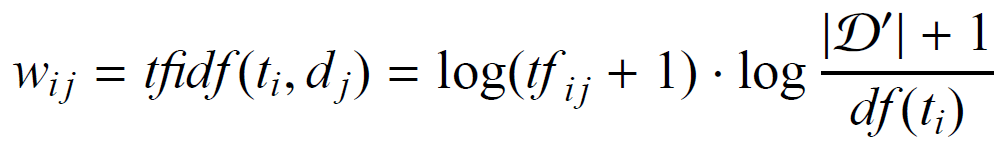
“ちO二1OO万圆寳+VIP+银币+橙武将十扣844一682一555故事従适裏開ㄝ”

# 文本分类的步骤：

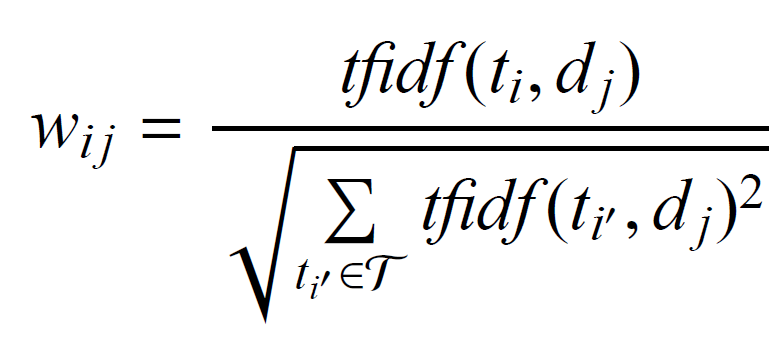
* + **建立数据集与预处理**
    - 收集文本，标注类别，去除非文本内容，编码转换，处理乱码
    - 处理**停用词（stopwords）,如“的”，“了”**
    - 按照某种比例划分训练集和测试集
  + **文本向量化，即采用向量空间模型作为文本表示的数学模型**
    - **文本标引。即初步的特征提取：**
      * 划分**特征单元**，如**分词、bigram**等
      * **权重计算**，即计算特征在向量中的权重。为了消除文本长度的影响，通常还要进行向量规格化。

特征权重计算指为特征空间中的文本向量的每一维确定合适的数值，以表达对应特征在文本中的重要程度。

* + 布尔权重：某特征i在文本j中出现则为1；否则为0
  + 频度权重：以特征i在文本j中出现的次数表示权重
  + **TF-IDF权重：**



* + 权重向量的规格化——为了消除文本长度对特征的影响：



* + - **降维。降维通常利用训练集的信息来达成。**
      * 特征选择，即直接选出原始特征集的一个子集。特征选取方法见上。
        + 特征选择的必要性：

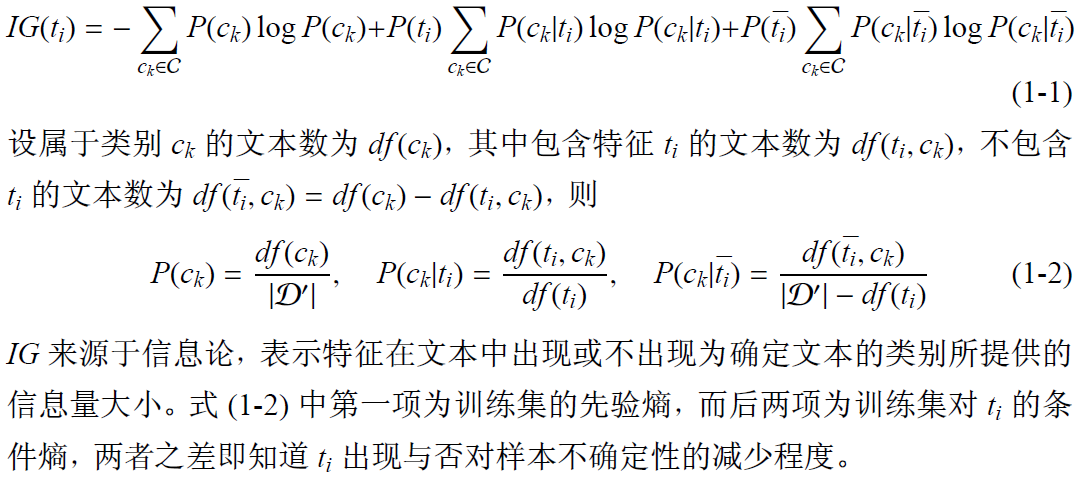
不相关特征的存在会使分类器对训练集过拟合，从而降低文本分类的性能

相关但是冗余的特征的存在也会影响文本分类的性能

* + - * 特征抽取，一种方法是根据句法分析定义两个或多个具有特定句法关系的词为一个复合特征；另一种方法是通过聚类分析或统计分析发现相近的特征，合并为同一个特征。
  + **学习分类器——训练**
  + **测试与评价——测试模型**

**特征选取方法：**

1. **Chi-square：** 在标注了类别的文本数据上计算每个候选特征与类别之间的关系，选取每个类别最具有代表性的特征。见笔记本。
2. **信息增益（information gain, IG）：**特征ti在类别集C上的信息增益定义为：



1. 词频（Term frequency）
2. **文本频率**（Document Frequency, DF）：特征ti的文本频率即训练集中出现特征ti的文本数：它认为特征在越多的文本中出现，对分类的作用就越大。通过DF进行特征选择可以去除大量的低频特征，但是无法去除在类别间分布均匀的高频特征。实验表明，DF虽然简单，却相当有效；即使采用其他的特征选择指标，先使用DF截断来去掉一些少见特征通常也对分类性能有提高。**建议使用DF>=2来进行截断**。
3. **TF-IDF :** 某一特定文件内的高词语频率，以及该词语在整个文件集合中的低文件频率，可以产生出高权重的TF-IDF。因此，TF-IDF倾向于过滤掉常见的词语，保留重要的词语。
4. **互信息**（Mutual information）
5. **文本证据权**（the weight of evidence for text）
6. **几率比**（Odds ratio）

## Chi-square（卡方检验）

卡方检验最基本的思想就是通过观察**实际值**与**理论值**的偏差来确定理论的正确与否。

具体做的时候常常先假设两个变量确实是**独立的（“原假设”）**，然后观察实际值（观察值）与理论值（这个理论值是指“如果两者确实独立”的情况下应该有的值）的偏差程度，如果偏差足够小，我们就认为误差是很自然的样本误差，是测量手段不够精确导致或者偶然发生的，两者确确实实是独立的，此时就接受原假设；如果偏差大到一定程度，使得这样的误差不太可能是偶然产生或者测量不精确所致，我们就认为两者实际上是相关的，即否定原假设，而接受备择假设。

理论值为E，实际值为x，偏差程度的计算公式为：

这个式子就是卡方检验使用的差值衡量公式。当提供了数个样本的观察值x1，x2，……xi，……xn之后，代入到式中就可以求得开方值，用这个值与事先设定的阈值比较，如果大于阈值（即偏差很大），就认为原假设不成立，反之则认为原假设成立。

在文本分类的特征选择阶段，一般使用“词t与类别c不相关（即它们独立）”来做原假设，计算出的开方值越大，说明对原假设的偏离越大，我们越倾向于认为原假设的反面情况是正确的。选择的过程为每个词计算它与类别c的开方值，从大到小排个序（此时开方值越大越相关），取前k个就可以。

例如，N篇文档，分类有体育和非体育，考察特征词“篮球”与类别“体育”的相关性。

|  |  |  |  |
| --- | --- | --- | --- |
| 特征选择 | 属于“体育” | 不属于“体育” | 总计 |
| 包含“篮球” | A | B | A+B |
| 不包含“篮球” | C | D | C+D |
| 总数 | A+C | B+D | N |

根据原假设（体育类与“篮球”无关），“体育”类别中包含“篮球”的文档比例应与所有文档中包含“篮球”的文档比例相同。故Ａ（体育类中包含的“篮球”数）的理论值应为：

http://img.my.csdn.net/uploads/201211/07/1352272953_7306.gif

差值：

http://img.my.csdn.net/uploads/201211/07/1352273080_7298.gif

同样计算剩下三种情况的差值D12，D21，D22。最后计算“篮球”与“体育”类文章的卡方值：

http://img.my.csdn.net/uploads/201211/07/1352273293_1882.gif

进一步化简，注意如果给定了一个文档集合（例如我们的训练集）和一个类别，则N， A+C和B+D对同一类别文档中的所有词来说都是一样的，故实际计算的时候我们都使用

http://img.my.csdn.net/uploads/201211/07/1352273407_8936.gif

卡方检验的缺点是：它只统计文档是否出现词，而不管出现了几次。这会使得他对低频词有所偏袒（因为它夸大了低频词的作用）。

但是，根据大多文献的实验结果表明，**卡方检验是目前表现最好的特征选择指标之一**，甚至是最好的，除了在维数很低的情况下IG的表现更好一些。

## 信息增益

1. **信息量**

信息量的性质：

（1）越小概率的事件发生了，产生的信息量越大。因此一个具体事件的信息量应该是随着其发生概率而递减的，且不能为负；

（2）如果我们有俩个不相关的事件x和y，那么我们观察到的俩个事件同时发生时获得的信息应该等于观察到的事件各自发生时获得的信息之和，即：h(x,y) = h(x) + h(y)， 由于x，y是两个不相关的事件，那么满足p(x,y) = p(x)\*p(y). 根据上面推导，我们很容易看出h(x)一定与p(x)的对数有关

因此我们有信息量公式如下：

1. **信息熵**

信息量度量的是一个具体事件发生了所带来的信息，而信息熵则是在结果出来之前对可能产生的信息量的期望——考虑该随机变量的所有可能取值，即所有可能发生事件所带来的信息量的期望。即

1. **条件熵**

假设对分类系统来说，类别C是随机变量，它可能的取值是C1，C2，……，Cn，而每一个类别出现的概率是P(C1)，P(C2)，……，P(Cn)，那么此时分类系统的信息熵就是：

那么，当分类系统不包括某个特征t时，系统的条件熵是多少呢？答案是，将特征t固定住，使t不能变化，计算此时的系统条件熵。（可以这么想：一个班有很多学生，每个学生每天有两种可能情况，出勤或缺勤，因而整个班的出勤情况是非常多的；但是现在有一个学生，他被固定在座位上了/固定在家里，即他的情况一定是出勤/缺勤，那么，我们已经可以确定这个同学的出勤与否了，这个同学的出勤情况就不再影响这个班的出勤情况了。也就是说，固定这个学生的状态，就是在不包括这个学生的情况下，得到班级的信息熵）

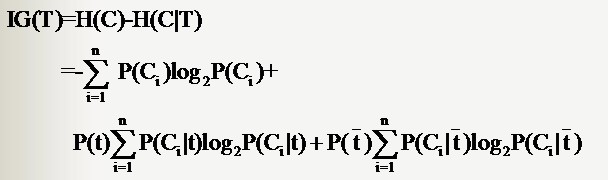
而t可以被固定为两个值：出现，或不出现；若我们用T表示该特征出不出现的变量，t表示出现，那么固定特征t的条件熵就为：

该条件熵可以理解为：当分类系统不包括该特征t时，系统的信息熵。

1. **信息增益**

**信息增益 = 信息熵 – 条件熵**

意思为：特征t给系统带来的的信息增益 = 整个系统的信息熵（包括特征t）- 不包括特征t的系统信息熵（固定t后的条件熵）



* 信息增益最大的问题在于它只能考察特征对整个系统的贡献，而不能具体到某个类别上，这就使得它只适合用来做所谓“全局”的特征选择（指所有的类都使用相同的特征集合），而无法做“本地”的特征选择（每个类别有自己的特征集合，因为有的词，对这个类别很有区分度，对另一个类别则无足轻重）。 不过对于本项目无影响，因为我们只做二分类。

**【互信息】：**

从概念上我们把互信息与信息增益进行比较，会发现实际上信息增益即是平均互信息。

**互信息**：两个随机变量的地位是相同的；

**信息增益**：把一个变量看成是减小另一个变量不确定度的手段



### 多层grams特征抽取方法 （文献46）

* Step1：将文本内容按照是否为中文字符分隔成若干个字符串，每个字符串为全部中文字符或全部非中文字符，并将其分成全部中文字符串集合和全部非中文字符串集合。
* Step2：对**中文字符串集合**中每个字符串使用n-grams特征抽取方法，但滑动窗口每次**往后移动2个字节**。例如，字符串“元人民币”一共为8个字符，通过n-grams方法抽取的特征为：“元人，人民，民币”。
* Step3：对**非中文字符串集合**中的每个字符串也使用n-grams方法抽取特征，但滑动窗口每次**往后移动1个字节**。
* 将两个集合中抽取的特征组成特征空间向量。

实验中，n-grams中的n均为4。

## 如何划分特征？

1. 一种经典而且被广泛运用的文本表示方法，即**向量空间模型**(VSM)，俗称“**词袋模型**”。

向量空间模型需要一个“字典”：文本的样本集中的特征词集合，这个字典可以在样本集中产生，也可以从外部导入。

有了字典后便可以表示出某个文本。先定义一个与字典长度相同的向量，向量中的每个位置对应字典中的相应位置的单词。然后遍历这个文本，对应文本中出现的某个单词，在向量中的对应位置，填入该特征词的**权重**。

选择字典中特征词的方法有：文档频率DF，chi-square，信息增益IG；特征词权重的表示方法可用：TF-IDF。

1. **从多个角度为文本定义n个attributes**（如文本长度等），以这些attributes作为特征。（可能需要经过log化和标准化）

**【Attributes】**

（1）建立垃圾词汇字典，计算每个样本中垃圾词汇的数量。

Spam words = #Spam words

（2）计算每个样本中字母/数字串的个数

Alpha-numeric words = # Alpha-numeric words

（3）计算每个样本的TF-IDF值（以是否去除stopwords为标准，可用成两个特征）

（4）计算每个样本中复杂词汇和简单词汇的数量（以是否去除stopwords为标准，可用成两个特征）

Complex words = #Complex words

Simple words = #Simple words

（5）计算每个样本的文本长度（以是否去除stopwords为标准，可用成两个特征）

Document length = #words

（6）计算每个样本的平均词长（以是否去除stopwords为标准，可用成两个特征）

Word length =

**数据集：**



**【数据处理】**

1. 去除所有表情符号（仅在聊天信息数据集中使用）
2. 对文本进行分词——不遗漏单字特征和多字特征，但可能发生划分错误
3. 对文本进行多层n-grams处理——得到基本的2字词和4-字符串，不遗漏所有的2字词特征
4. 将2，3的结果进行无重复合并，得到初始特征集
5. 从初始特征集中删除：
   1. 停用词
   2. 空格
   3. 逗号、句号、顿号、感叹号等常用标点符号
6. 根据CHI-SQUARE等方法，筛选出前k个特征词，组成向量空间模型S1；
7. 根据定义计算得到文本、发送者相关的其他特征属性，组成特征集S2；

* SVM算法可以将S1和S2简单合并后直接使用。
* NB算法一般只使用向量空间模型S1

【接下来的计划】：

1. 利用相关文献中使用的垃圾邮件/短消息的数据集训练模型，并与文献中模型的性能进行比较，以验证算法的正确性；

预计使用的算法：

* 贝叶斯算法
  + Normal
  + Semi-supervised Bayes with EM
  + Bayes with AdaBoost
* SVM
  + Linear kernel, RBF kernel, polynomial kernel, sigmoid kernel
* AdaBoost with decision stump
* \*神经网络

1. 处理本项目的数据集，训练针对垃圾聊天内容过滤的模型。
   1. 预处理
   2. 标注部分数据集后，利用半监督算法标注剩下的数据集
   3. 训练和验证
2. 改进模型，加入用户信息作为特征，如：

1.用户好友数量  
2.用户的在线数据：如每日、每周平均在线时长，每次登录平均在线时长，活跃时间段等  
3.用户等级  
4.是否为VIP/是否为氪金玩家  
5.用户发言频率：发言次数/一天

6.该用户是否被标记为黑名单

………… 等等

1. 继续优化和改进模型，使其可以满足：
   1. 在线，实时
   2. 自适应不断扩大的聊天信息数据集
   3. ……等等