文本分类及朴素贝叶斯分类器

本讲内容

- 文本分类的概念及其与IR的关系
- 朴素贝叶斯分类器(朴素贝叶斯)
- 文本分类的评价

提纲

- 文本分类
- 朴素贝叶斯
- 朴素贝叶斯的生成模型
- 朴素贝叶斯理论
- 特征选择
- 文本分类评价

文本分类

- Text Classification或者Text Categorization：给定分类体系(taxonomy)，将一篇文档分到其中一个或者多个类别中的过程。
- 给定文档$d \in X$和一个固定的类别集合$C=\{c_1, c_2, \ldots, c_J\}$，其中$X$表示文档空间(document space)，类别(class)也通常称为类(category)或类标签(label)。
  - 按类别数目：binary vs. multi-class
  - 按每篇文档赋予的标签数目：single label vs. multi label

分类方法: 1. 手工方法

- Web发展的初期，Yahoo使用人工分类方法来组织Yahoo目录，类似工作还有： ODP, PubMed
- 如果是专家来分类精度会非常高
- 如果问题规模和分类团队规模都很小的时候，能否保持分类结果的一致性
- 但是对人工分类进行规模扩展将十分困难，代价昂贵

- → 因此，需要自动分类方法

分类方法: 2. 规则方法

- Google Alerts的例子是基于规则分类的
- 存在一些IDE开发环境来高效撰写非常复杂的规则 (如Verity)
- 通常情况下都是布尔表达式组合 (如Google Alerts)
- 如果规则经过专家长时间的精心调优，精度会非常高
- 建立和维护基于规则的分类系统非常繁琐，开销也大

分类方法: 3. 机器学习方法

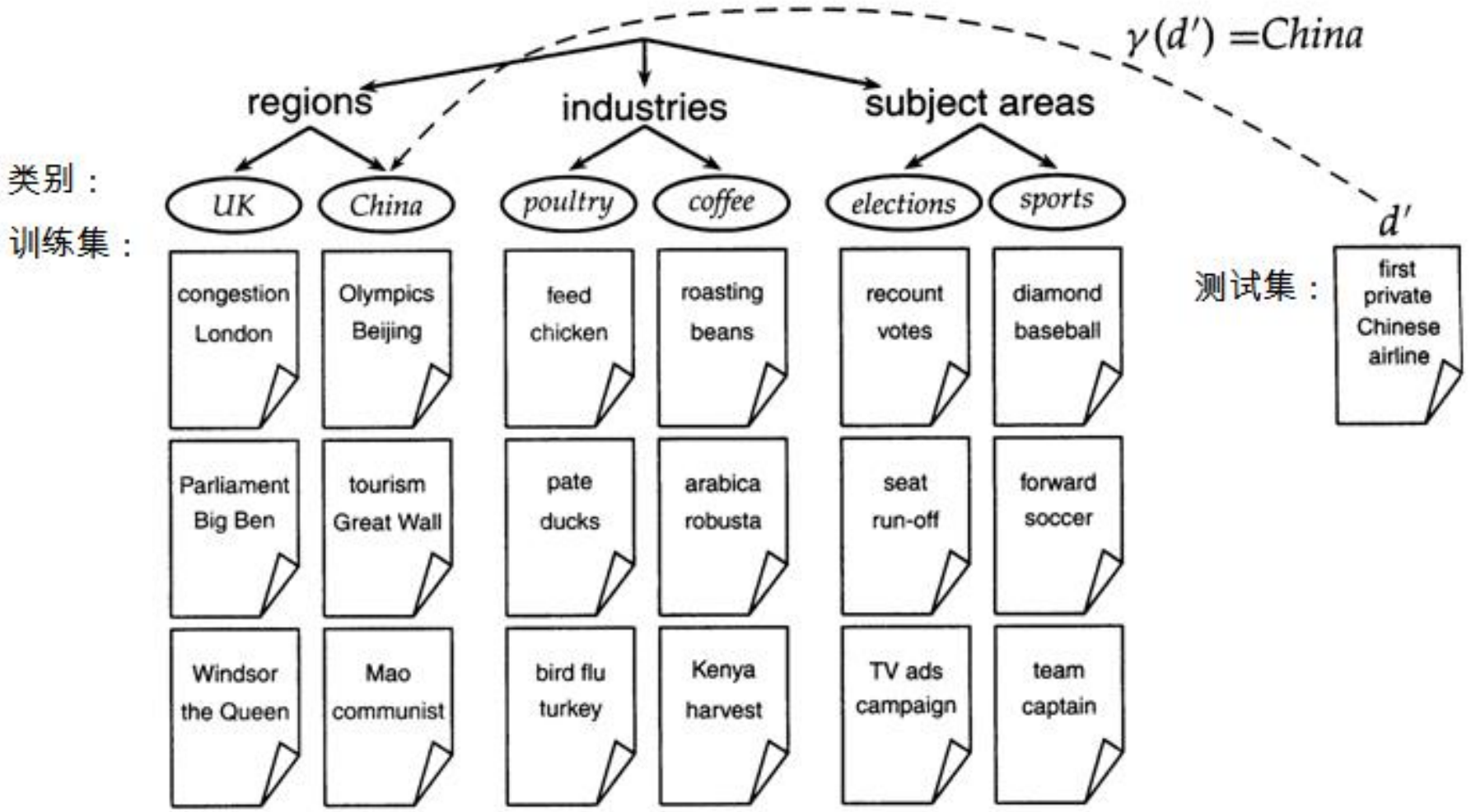
- 文本分类被定义为一个学习问题，这也是本书中的定义，包括
  - (i) 通过有监督的学习，得到分类函数γ，然后将其
  - (ii) 用于对新文档的分类
- 将介绍一系列分类方法: 朴素贝叶斯, Rocchio, kNN, SVM
- 当学习方法基于统计时，这种方法也称为统计文本分类(statistical text classification)。
  - 在统计文本分类中，对于每个类别需要一些好的文档样例(或者称为训练文档)。
  - 由于需要人来标注训练文档，所以对人工分类的需求仍然存在。
  - 这里的标注(labeling)指的是对每篇文档赋予类别标签的过程。

基于学习的文本分类

- 文档空间X
  - 文档都在该空间下表示—通常都是某种高维空间
- 固定的类别集合C = $\{c_1, c_2, \ldots, c_J\}$
  - 类别往往根据应用的需求来人为定义 (如, 相关类 vs. 不相关类)
- 训练集D，文档d用类别c来标记，$<d, c> \in X \times C$
  - 利用学习算法，可以学习一个分类器 γ，将文档映射成类别： $\gamma : X \rightarrow C$
- 文档分类的实现
  - 对于文档$d \in X$，可确定$\gamma(d) \in C$, 即确定d最可能属于的类别$c_i=\gamma(d)$， $c_i \in C$

文本分类中的类别、训练集及测试集

Classes, training set, and test set in text classification



无监督/有监督 学习

- supervised learning 监督学习
  - 利用一组已知类别的样本调整分类器的参数，使其达到所要求性能的过程，也称为监督训练或有教师学习。
- 无监督学习
  - 若所给的学习样本不带有类别信息，就是无监督学习。

搜索引擎中的文本分类应用

- 语言识别 (类别: English vs. French等)
- 垃圾网页的识别 (垃圾网页 vs. 正常网页)
- 是否包含淫秽内容 (色情 vs. 非色情)
- 领域搜索或垂直搜索 – 搜索对象限制在某个垂直领域 (如健康医疗) (属于该领域 vs. 不属于该领域)
- 静态查询 (如，Google Alerts)
- 情感识别: 影评或产品评论是贬还是褒 (褒评 vs. 贬评)

提纲

- 文本分类
- 朴素贝叶斯
- 朴素贝叶斯的生成模型
- 朴素贝叶斯理论
- 特征选择
- 文本分类评价

朴素贝叶斯分类器Naive Bayes text classification

是一个概率分类器

- 文档 d 属于类别 c 的概率计算如下：

独立性假设

$$P(c|d)=\frac{P(c)P(d|c)}{P(d)}\propto P(c)P(d|c)\propto P(c)\prod_{1\le k\le n_d}P(t_k|c)$$

所有文档都是一样的

- t_k是d中的词条，n_d是文档的长度(词条的个数)
- $P(t_k|c)$ 是词项t_k出现在类别c中文档的概率，或类别c生成词项t_k的概率，或度量的是当c是正确类别时t_k 的贡献
- $P(c)$ 是类别c的先验概率
- 如果文档的词项无法提供属于哪个类别的信息，那么直接选择$P(c)$最高的那个类别

具有最大后验概率的类别

- 朴素贝叶斯分类的目标是寻找“最佳”的类别
- 最佳类别是具有最大后验概率(maximum a posteriori - MAP)的类别 c_{map}:

$$c_{map} = \arg\max_{c \in C} \hat{P}(c|d) \propto \arg\max_{c \in C} \hat{P}(c) \prod_{1 \le k \le n_d} \hat{P}(t_k|c)$$

- 由于不知道参数的真实值，所以上述公式中采用了从训练集中得到的估计值 $\hat{P}$ 来代替 P。

- 很多小概率的乘积会导致浮点数下溢出
- 由于 $\log(xy) = \log(x) + \log(y)$, 可以通过取对数将原来的乘积计算变成求和计算
- 由于log是单调函数，因此得分最高的类别不会发生改变
- 因此，实际中常常使用的是：

$$c_{map} = \arg\max_{c \in C}[\log \hat{P}(c) + \sum_{1 \le k \le n_d} \log \hat{P}(t_k \mid c)]$$

朴素贝叶斯分类器

- 分类规则:

$$c_{map} = \arg\max_{c \in C}[\log \hat{P}(c) + \sum_{1 \le k \le n_d} \log \hat{P}(t_k \mid c)]$$

- 简单说明:
  - 每个条件参数$\hat{P}(t_k \mid c)$ 是反映t_k对c的贡献高低的一个权重
  - 先验概率 $\hat{P}(c)$ 是反映类别c的相对频率的一个权重
  - 因此，所有权重的求和反映的是文档属于类别的可能性
  - 选择最具可能性的类别

参数估计 1: 极大似然估计

- 如何从训练数据中估计 $\hat{P}(c)$ 和 $\hat{P}(t_k \mid c)$ ？
- 先验:

$$\hat{P}(c) = \frac{N_c}{N}$$

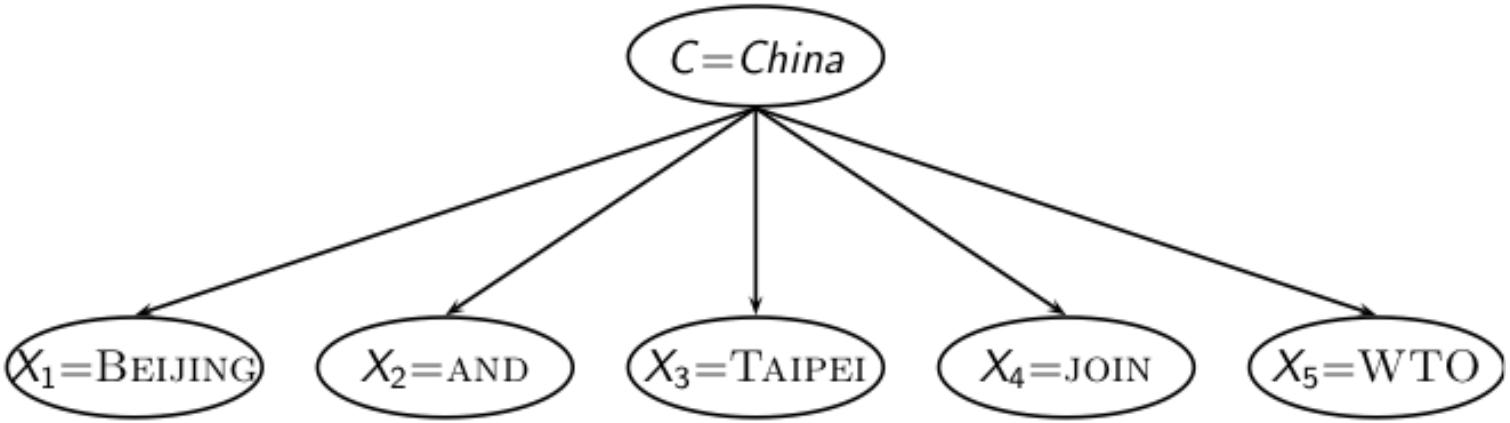
  - N_c: 训练集中类c中的文档数目; N: 训练集中文档总数
- 条件概率:

$$\hat{P}(t_k \mid c) = \frac{T_{ct}}{\sum_{t' \in V} T_{ct'}}$$

  - 引入了位置独立性假设(positional independence assumption)，在该假设下，T_{ct}是t在训练集类c文档中所有位置的出现次数之和。即不同位置上的概率值采用相同的估计办法。比如，如果次t在一篇文档中出现过两次，分别在k_1和k_2的位置上，那么假定

$$\hat{P}(t_{k1} \mid c) = \hat{P}(t_{k2} \mid c)$$

MLE估计中的问题：零概率问题



$$P(China|d) \propto P(China) \cdot P(\text{BEIJING}|China) \cdot P(\text{AND}|China) \cdot P(\text{TAIPEI}|China) \cdot P(\text{JOIN}|China) \cdot P(\text{WTO}|China)$$

- 如果 **WTO** 在训练集中没有出现在类别 ***China*****中:**

$$P(\text{WTO}\,|\,China) = \frac{T_{China,\text{WTO}}}{\sum_{t' \in V} T_{China,t'}} = \frac{0}{\sum_{t' \in V} T_{China,t'}} = 0$$

- → 那么，对于任意包含WTO的文档，$P(China|d) = 0$。

一旦发生零概率，将无法判断类别

避免零概率: 加1平滑

- 平滑前：

$$\hat{P}(t|c) = \frac{T_{ct}}{\sum_{t' \in V} T_{ct'}}$$

- 平滑后: 对每个量都加上1

$$\hat{P}(t|c) = \frac{T_{ct}+1}{\sum_{t' \in V}(T_{ct'}+1)} = \frac{T_{ct}+1}{(\sum_{t' \in V} T_{ct'})+B}$$

- B 是不同的词语个数 (这种情况下词汇表大小 $B = |V|$)
- 加1平滑可以认为是采用均匀分布作为先验分布(每个词项在每个类中出现1次)，然后根据训练数据进行更新得到的结果。

朴素贝叶斯: 训练过程

$$c_{map} \propto \arg\max_{c \in C} \hat{P}(c) \prod_{1 \le k \le n_d} \hat{P}(t_k \mid c) = \arg\max_{c \in C} \frac{N_c}{N} \prod_{1 \le k \le n_d} \frac{T_{ct_k} + 1}{(\sum_{t' \in V} T_{ct'}) + B}$$

TRAINMULTINOMIALNB($\mathbb{C}, \mathbb{D}$)

1 $V \leftarrow$ EXTRACTVOCABULARY($\mathbb{D}$) 词汇表
2 $N \leftarrow$ COUNTDOCS($\mathbb{D}$)
3 **for each** $c \in \mathbb{C}$
4 **do** $N_c \leftarrow$ COUNTDOCSINCLASS($\mathbb{D}, c$)
5 $prior[c] \leftarrow N_c / N$ 类别先验
6 $text_c \leftarrow$ CONCATENATETEXTOFALLDOCSINCLASS($\mathbb{D}, c$)
7 **for each** $t \in V$
8 **do** $T_{ct} \leftarrow$ COUNTTOKENSOFTERM($text_c, t$) 计算词频
9 **for each** $t \in V$
10 **do** $condprob[t][c] \leftarrow \frac{T_{ct}+1}{\sum_{t'}(T_{ct'}+1)}$ 特征|类别 条件概率
11 **return** $V, prior, condprob$

运算量:

计算|C|个 $\hat{p}(c)$

计算|C|·|V|个 $\hat{p}(t_k \mid c)$

朴素贝叶斯: 测试过程

训练过程已得到了估计参数 $\hat{p}(c)$ 和 $\hat{p}(t_k \mid c)$

$$c_{map} = \arg\max_{c \in C}[\log \hat{P}(c) + \sum_{1 \le k \le n_d} \log \hat{P}(t_k \mid c)]$$

测试过程根据 $\hat{p}(c)$ 和 $\hat{p}(t_k \mid c)$ 计算文档 d 的 c_{map}

```
APPLYMULTINOMIALNB(ℂ, V, prior, condprob, d)
1  W ← EXTRACTTOKENSFROMDOC(V, d)
2  for each c ∈ ℂ
3  do score[c] ← log prior[c]
4     for each t ∈ W
5     do score[c] += log condprob[t][c]
6  return arg max_{c∈ℂ} score[c]
```

运算量:

计算|C| 个 $\hat{p}(c)$

计算|C|·|V|个 $\hat{p}(t_k \mid c)$

朴素贝叶斯分类示例

	文档ID	文档中词	属于c=China类?
训练集	1	Chinese Beijing Chinese	Yes
	2	Chinese Chinese Shanghai	Yes
	3	Chinese Macao	Yes
	4	Tokyo Japan Chinese	No
测试集	5	Chinese Chinese Chinese Tokyo Japan	?

Priors

$$\hat{P}(c) = \frac{3}{4}$$

$$\hat{P}(\bar{c}) = \frac{1}{4}$$

	文档ID	文档中词	属于c=China类?
训练集	1	Chinese Beijing Chinese	Yes
	2	Chinese Chinese Shanghai	Yes
	3	Chinese Macao	Yes
	4	Tokyo Japan Chinese	No
测试集	5	Chinese Chinese Chinese Tokyo Japan	?

Conditional probabilities:

$$P(t|c) = \frac{T_{ct}+1}{(\sum_{t' \in V} T_{ct'}) + |V|}$$

$$\hat{P}(Chinese|c) = \frac{5+1}{8+6} = \frac{3}{7} \quad \hat{P}(Tokyo|c) = \hat{P}(Japan|c) = \frac{0+1}{8+6} = \frac{1}{14}$$

$$\hat{P}(Chinese|\bar{c}) = \hat{P}(Tokyo|\bar{c}) = \hat{P}(Japan|\bar{c}) = \frac{1+1}{3+6} = \frac{2}{9}$$

$$\hat{P}(c|d_5) \propto \frac{3}{4}\left(\frac{3}{7}\right)^3 \frac{1}{14}\frac{1}{14} \approx 0.0003 \quad \hat{P}(\bar{c}|d_5) \propto \frac{1}{4}\left(\frac{2}{9}\right)^3 \frac{2}{9}\frac{2}{9} \approx 0.0001$$

测试文档分到$c = China$类，这是因为d_5中起正向作用的CHINESE出现3次的权重高于起反向作用的 JAPAN和TOKYO的权重之和。

朴素贝叶斯的时间复杂度分析

mode	time complexity
training	$\Theta(\|\mathbb{D}\|L_{ave} + \|\mathbb{C}\|\|V\|)$
testing	$\Theta(L_a + \|\mathbb{C}\|M_a) = \Theta(\|\mathbb{C}\|M_a)$

- L_{ave}: 训练文档的平均长度，D: 训练文档，V: 词汇表，C: 类别集合，L_a: 测试文档的平均长度，M_a: 测试文档中不同的词项个数
- $\Theta(|D|L_{ave})$：参数计算所需的预处理复杂度，词汇表的抽取、词项计算等
- $\Theta(|C||V|)$：参数估计的时间复杂度，|C||V|个条件概率、|C|个先验概率
- 因此：朴素贝叶斯 对于训练集的大小和测试文档的大小而言是线性的(相对于测试文档的长度而言)。这是最优的。

小结：Naive Bayes text classification

- 分类目标：找出文档最可能属于的类别。对于NB来说，最可能的类是具有MAP估计值的结果c_{map}:

$$c_{map} = \arg\max_{c \in C} \hat{P}(c|d) \propto \arg\max_{c \in C} \hat{P}(c) \prod_{1 \le k \le n_d} \hat{P}(t_k|c)$$

- 估计参数

$$\hat{P}(c) = \frac{N_c}{N} \qquad \hat{P}(t_k|c) = \frac{T_{ct}}{\sum_{t' \in V} T_{ct'}}$$

- 零概率问题→平滑

$$\hat{P}(t|c) = \frac{T_{ct}+1}{\sum_{t' \in V}(T_{ct'}+1)} = \frac{T_{ct}+1}{(\sum_{t' \in V} T_{ct'}) + B}$$

提纲

- 文本分类
- 朴素贝叶斯
- 朴素贝叶斯的生成模型
- 朴素贝叶斯理论
- 特征选择
- 文本分类评价

NB分类器的生成(Generative)模型

多项分布：有多种取值，比如抛骰子

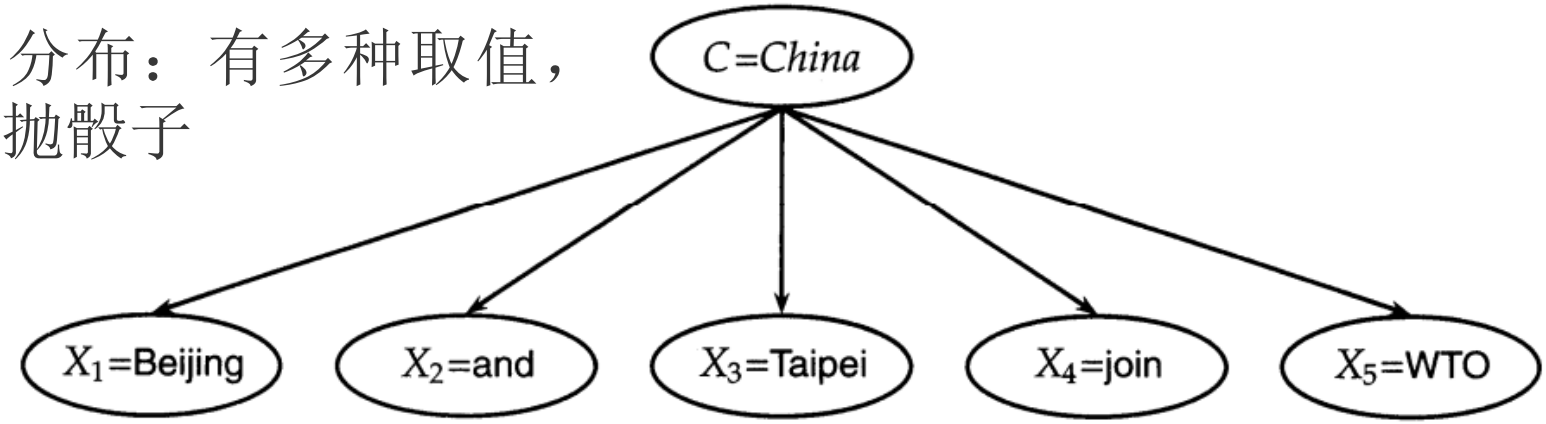


图 13-4　多项式 NB 模型

伯努利(二项)分布：只有2种取值，抛硬币

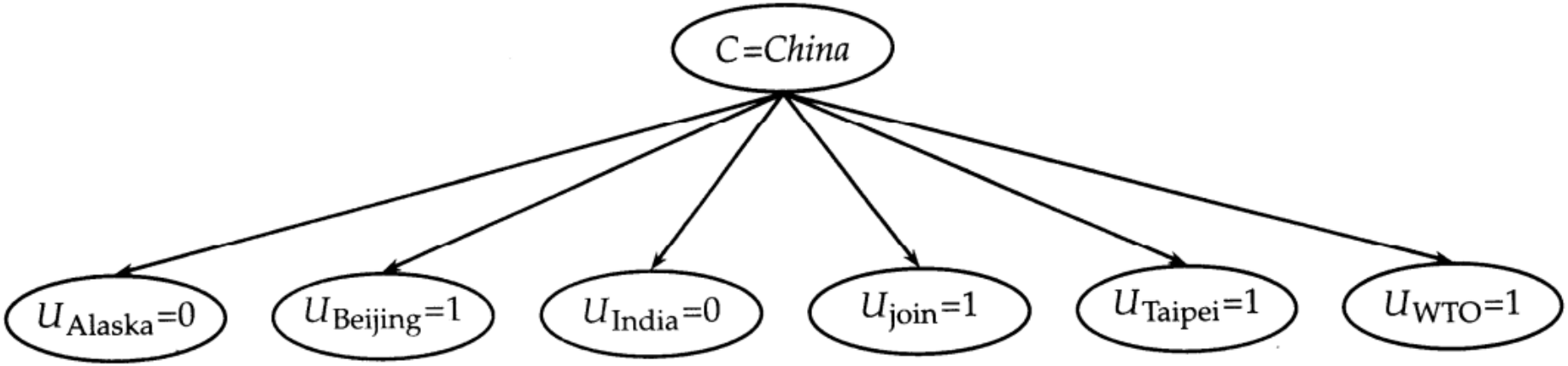


图 13-5　贝努利 NB 模型

Naive Bayes algorithm

- $\hat{P}(t|c)$ 的估计策略不同
  - 贝努利：类c文档中包含t的文档数的比率
  - 多项式：t出现的次数占类c文档中所有词条数目的比率
  - 当对测试文档进行分类时，贝努利模型只考虑词项的出现或不出现(即二值)，并不考虑出现的次数，而多项式模型中则要考虑出现次数
- 未出现词项在分类中的使用不同
  - 多项式模型：不影响分类效果
  - 贝努利模型：计算$P(c|d)$时要以一个因子来参与计算，因为贝努利模型对词项的未出现也要显式建模

Naive Bayes algorithm

- $\hat{P}(t|c)$ 的估计策略不同
- 未出现词项在分类中的使用不同

TrainMultinomialNB($\mathbb{C}$, $\mathbb{D}$) 多项式

1 $V \leftarrow$ ExtractVocabulary($\mathbb{D}$)
2 $N \leftarrow$ CountDocs($\mathbb{D}$)
3 **for each** $c \in \mathbb{C}$
4 **do** $N_c \leftarrow$ CountDocsInClass($\mathbb{D}, c$)
5 $prior[c] \leftarrow N_c/N$
6 $text_c \leftarrow$ ConcatenateTextOfAllDocsInClass($\mathbb{D}, c$)
7 **for each** $t \in V$
8 **do** $T_{ct} \leftarrow$ CountTokensOfTerm($text_c, t$)
9 **for each** $t \in V$
10 **do** $condprob[t][c] \leftarrow \frac{T_{ct}+1}{\sum_{t'}(T_{ct'}+1)}$
11 **return** $V, prior, condprob$

ApplyMultinomialNB($\mathbb{C}, V, prior, condprob, d$)

1 $W \leftarrow$ ExtractTokensFromDoc(V, d)
2 **for each** $c \in \mathbb{C}$
3 **do** $score[c] \leftarrow \log prior[c]$
4 **for each** $t \in W$
5 **do** $score[c] \mathrel{+}= \log condprob[t][c]$
6 **return** $\arg\max_{c \in \mathbb{C}} score[c]$

W文档词汇

TrainBernoulliNB($\mathbb{C}$, $\mathbb{D}$) 贝努利

1 $V \leftarrow$ ExtractVocabulary($\mathbb{D}$)
2 $N \leftarrow$ CountDocs($\mathbb{D}$)
3 **for each** $c \in \mathbb{C}$
4 **do** $N_c \leftarrow$ CountDocsInClass($\mathbb{D}, c$)
5 $prior[c] \leftarrow N_c/N$
6 **for each** $t \in V$
7 **do** $N_{ct} \leftarrow$ CountDocsInClassContainingTerm($\mathbb{D}, c, t$)
8 $condprob[t][c] \leftarrow (N_{ct}+1)/(N_c+2)$
9 **return** $V, prior, condprob$

ApplyBernoulliNB($\mathbb{C}, V, prior, condprob, d$)

1 $V_d \leftarrow$ ExtractTermsFromDoc(V, d)
2 **for each** $c \in \mathbb{C}$
3 **do** $score[c] \leftarrow \log prior[c]$
4 **for each** $t \in V$
5 **do if** $t \in V_d$
6 **then** $score[c] \mathrel{+}= \log condprob[t][c]$
7 **else** $score[c] \mathrel{+}= \log(1 - condprob[t][c])$
8 **return** $\arg\max_{c \in \mathbb{C}} score[c]$

V整个词汇表

基于贝努利模型的NB示例

- 参数的计算：$\hat{P}(c)$ 和 $\hat{P}(t|c)$的估计

	文档ID	文档中词	属于c=China类?
训练集	1	Chinese Beijing Chinese	Yes
	2	Chinese Chinese Shanghai	Yes
	3	Chinese Macao	Yes
	4	Tokyo Japan Chinese	No
测试集	5	Chinese Chinese Chinese Tokyo Japan	?

	文档ID	文档中词	属于c=China类?
训练集	1	Chinese Beijing Chinese	Yes
	2	Chinese Chinese Shanghai	Yes
	3	Chinese Macao	Yes
	4	Tokyo Japan Chinese	No
测试集	5	Chinese Chinese Chinese Tokyo Japan	?

Priors一样

$$\hat{P}(c) = \frac{3}{4}$$

$$\hat{P}(\overline{c}) = \frac{1}{4}$$

	文档ID	文档中词	属于c=China类?
训练集	1	Chinese Beijing Chinese	Yes
	2	Chinese Chinese Shanghai	Yes
	3	Chinese Macao	Yes
	4	Tokyo Japan Chinese	No
测试集	5	Chinese Chinese Chinese Tokyo Japan	?

Conditional probabilities: 类*c*文档中包含*t*的文档数的比率

$$\hat{P}(Chinese|c)=\frac{3+1}{3+2}=\frac{4}{5} \qquad \hat{P}(Tokyo|c)=\hat{P}(Japan|c)=\frac{0+1}{3+2}=\frac{1}{5}$$

$$\hat{P}(Beijing|c)=\hat{P}(Macao|c)=\hat{P}(Shanghai|c)=\frac{1+1}{3+2}=\frac{2}{5}$$

$$\hat{P}(Chinese|\bar{c})=\hat{P}(Tokyo|\bar{c})=\hat{P}(Japan|\bar{c})=\frac{1+1}{1+2}=\frac{2}{3}$$

$$\hat{P}(Beijing|\bar{c})=\hat{P}(Macao|\bar{c})=\hat{P}(Shanghai|\bar{c})=\frac{0+1}{1+2}=\frac{1}{3}$$

有 3 篇文档属于 c 类，1 篇文档属于非 c 类，另外由于对每个词项都只考虑出现与不出现两种情形，分母平滑常数 B 为 2

	文档ID	文档中词	属于c=China类?
训练集	1	Chinese Beijing Chinese	Yes
	2	Chinese Chinese Shanghai	Yes
	3	Chinese Macao	Yes
	4	Tokyo Japan Chinese	No
测试集	5	Chinese Chinese Chinese Tokyo Japan	?

$$\hat{P}(c \mid d_5) \propto \hat{P}(c)\hat{P}(Chinese \mid c)\hat{P}(Japan \mid c)\hat{P}(Tokyo \mid c)$$

$$(1-\hat{P}(Beijing \mid c))(1-\hat{P}(Shanghai \mid c))(1-\hat{P}(Macao \mid c))$$

未出现词

$$= \frac{3}{4}\frac{4}{5}\frac{1}{5}\frac{1}{5}(1-\frac{2}{5})(1-\frac{2}{5})(1-\frac{2}{5}) \approx 0.0005$$

$$\hat{P}(\overline{c} \mid d_5) \propto \frac{1}{4}\frac{2}{3}\frac{2}{3}\frac{2}{3}(1-\frac{1}{3})(1-\frac{1}{3})(1-\frac{1}{3}) \approx 0.022$$

根据上述结果，分类器最终会将测试文档归为非 c 类。当只关注词项出现与否而不考虑词项频率时，Japan 和Tokyo对于c 来说是正向标志特征(2/3 > 1/5)，而Chinese属于c类和非c类的条件概率的差异还不足以影响分类的结果。

小结：朴素贝叶斯分类器的生成模型

- 文本分类的步骤
  - 训练
  - 测试
- 建立 NB 分类器有两种不同的方法
  - Multinomial NB model
  - Bernoulli model
- Naive Bayes algorithm
  - $\hat{P}(t|c)$的估计策略不同
  - 未出现词项在分类中的使用不同

提纲

- 文本分类
- 朴素贝叶斯
- 朴素贝叶斯的生成模型
- 朴素贝叶斯理论
- 特征选择
- 文本分类评价

朴素贝叶斯: 分析

- 接下来对朴素贝叶斯的性质进行更深层次的理解
- 包括形式化地推导出分类规则
- 然后介绍在推导中的假设

朴素贝叶斯规则

给定文档的条件下，希望得到最可能的类别

$$c_{map} = \arg\max_{c \in C} P(c \mid d)$$

应用贝叶斯定律 $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$:

$$c_{map} = \arg\max_{c \in C} \frac{P(d \mid c)P(c)}{P(d)}$$

由于分母$P(d)$对所有类别都一样，因此可以去掉:

$$c_{map} = \arg\max_{c \in C} P(d \mid c)P(c)$$

两种模型的文本生成过程

- 给定类别时文档生成的条件概率计算有所不同
  - 多项式模型 $P(d|c) = P(<t_1, \ldots, t_k, \ldots, t_{nd}>|c)$，$<t_1,\ldots, t_{nd}>$是在$d$ 中出现的词项序列(每个位置都考虑，当然要去掉那些从词汇表中去掉的词，如停用词)
  - 贝努利模型 $P(d|c) = P(<e_1, \ldots, e_i, \ldots, e_M>|c)$，$<e_1,\ldots, e_i, \ldots, e_M>$是一个$M$维的布尔向量，表示每个词项在文档$d$中存在与否
- 两种不同的文档表示方法
  - 第一种方法的文档空间X是所有词项序列的集合
  - 第二种方法的文档空间X是$\{0,1\}^M$

两种生成模型需要估计的参数

- 多项式模型$P(d|c) = P(<t_1, \ldots, t_k, \ldots, t_{nd}>|c)$
  - nd 是文档的长度(词条的个数)
  - $\hat{P}(c)$：$|C|$个
  - $\hat{P}(t|c)$：$M^{nd} \bullet |C|$个 (每个位置)
- 贝努利模型$P(d|c) = P(<e_1, \ldots, e_i, \ldots, e_M>|c)$
  - M是词汇表中所有词项的个数
  - $\hat{P}(c)$： $|C|$个
  - $\hat{P}(t|c)$：$2^M \bullet |C|$个不同的参数，每个参数都是M个e_i取值和一个类别取值的组合
- 多项式模型和贝努利模型具有相同数量级的参数个数。
- 要估计这么多参数，必须需要大量的训练样例。但是，训练集的规模总是有限的，于是出现数据稀疏性(data sparseness)问题
- 由于参数空间巨大，对这些参数进行可靠估计是不可行的

朴素贝叶斯条件独立性假设

为减少参数数目，给出朴素贝叶斯条件独立性假设，即给定类别时，假设特征之间是相互独立的：

$$\text{Multinomianl } P(d \mid c) = P(< t_1,...,t_{n_d} > \mid c) = \prod_{1 \le k \le n_d} P(X_k = t_k \mid c)$$

$$\text{Bernoulli } P(d \mid c) = P(< e_1,...,e_M > \mid c) = \prod_{1 \le i \le M} P(U_i = e_i \mid c)$$

上面公式中引入了两类随机变量X_k和U_i，这样的话两个不同的文本生成模型就更清晰。

- X_k是文档在位置k上的随机变量，$P(X_k = t \mid c)$表示一篇c类文档中词项t 出现在位置k上的概率。
- 随机变量U_i对应词项t_i，当词项在文档中不出现时取0，出现时取1。$P(U_i = 1 \mid c)$表示的是t_i出现在c类文档中的概率，这时可以是在任意位置上出现任意多次，既只考虑二值。

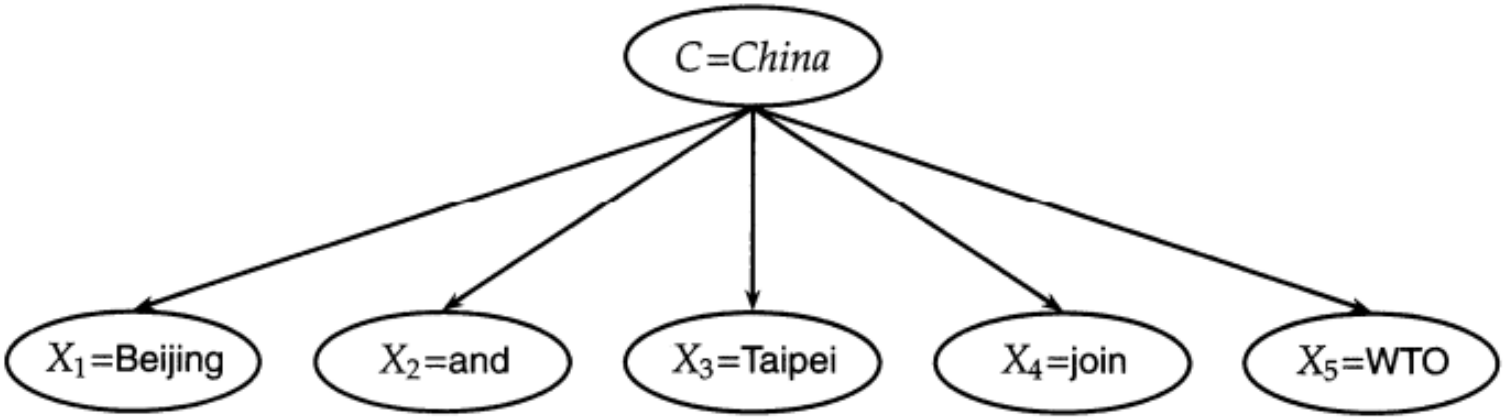


图 13-4　多项式 NB 模型

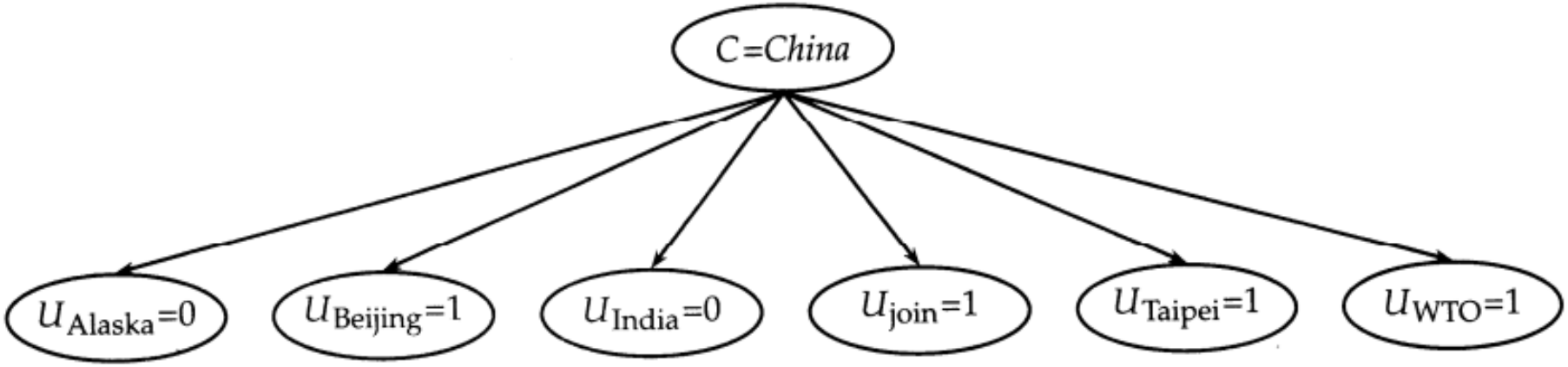


图 13-5　贝努利 NB 模型

朴素贝叶斯的位置独立性假设

- 即使是采用条件独立性假设，但假如在文档中每个位置k上的概率分布不同的话，那么对于多项式模型来说仍然有太多的参数需要估计。
  - 比如，bean出现在coffee类文档的第1个位置和出现在第2个位置的概率是不同的
- 词项在文档中的出现位置本身并不包含任何对分类有用的信息
- 故在多项式模型中引入第二个独立性假设—位置独立性假设(positional independence)，即词项在文档中每个位置的出现概率是一样的，也就是对于任意位置k_1、k_2、词项t和类别c，有

$$\hat{p}(t_{k1}|c) = \hat{p}(t_{k2}|c)$$

$M^{nd}\bullet|C|$　　$2^{M}\bullet|C|$

$\Theta(M\bullet|C|)$

- 基于条件独立性和位置独立性假设，我们只需要估计$\Theta(M\bullet|C|)$个多项式模型下的参数$P(t_k|c)$或贝努利模型下的参数$P(e_i|c)$，其中每个参数对应一个词项和类别的组合。

两个模型的比较

表13-3　多项式模型和贝努利模型的比较

	多项式模型		贝努利模型
事件模型	词条生成模型		文档生成模型
随机变量	$X=t$，当且仅当 t 出现在给定位置		$U_t=1$，当且仅当 t 出现在文档中
文档表示	$d=<t_1,\ldots,t_k,\ldots,t_{nd}>, t_k \in V$		$d=<e_1,\ldots,e_i,\ldots,e_M>, e_i \in \{0,1\}$
参数估计	$\hat{P}(X=t\mid c)$		$\hat{P}(U_i=e\mid c)$
决策规则：最大化	$\hat{P}(c)\prod_{1\le k\le n_d}\hat{P}(X=t_k\mid c)$		$\hat{P}(c)\prod_{t_i\in V}\hat{P}(U_i=e_i\mid c)$
词项多次出现	考虑		不考虑
文档长度	能处理更长文档		最好处理短文档
特征数目	能够处理更多特征		特征数目较少效果更好
词项the的估计	$\hat{P}(X=\text{the}\mid c)\approx 0.05$	计算公式	$\hat{P}(U_{\text{the}}\mid c)\approx 1.0$

“朴素”

- 条件独立性假设声称在给定类别的情况下特征之间相互独立，这对于实际文档中的词项来说几乎不可能成立。
- 多项式模型中还给出了位置独立性假设。而由于贝努利模型中只考虑词项出现或不出现，所以它忽略了所有的位置信息。这种词袋模型忽略了自然语言句子中词序相关的信息
- 所以**NB** 对自然语言的建模做了非常大的简化，从这个意义上讲，如何能保证**NB**方法的分类效果？

朴素贝叶斯方法起作用的原因

- 即使在条件独立性假设严重不成立的情况下，朴素贝叶斯方法依然能够高效地工作。例如

表13-4　正确的参数估计意味着精确的预测，但是精确的预测不一定意味着正确的参数估计

	c_1	c_2	选择的类别
真实概率$P(c \mid d)$	0.6	0.4	c_1
$\hat{P}(c)\prod_{1\le k\le n_d}\hat{P}(t_k \mid c)$（公式（13-13））	0.000 99	0.00 001	
NB估计 $\hat{P}(c \mid d)$	0.99	0.01	c_1

概率化

- 概率$P(c_2|d)$被过低估计(0.01)，而概率$P(c_1|d)$被过高估计 (0.99)。然而，分类决策取决于哪个类别得分最高，并不关注得分本身的精确性。尽管概率估计效果很差，但是NB会给c_1一个很高的分数，因此最后会将d归到正确的类别中
- 分类的目标是预测正确的类别，并不是准确地估计概率
- 准确估计 ⇒ 精确预测，反之并不成立!

提纲

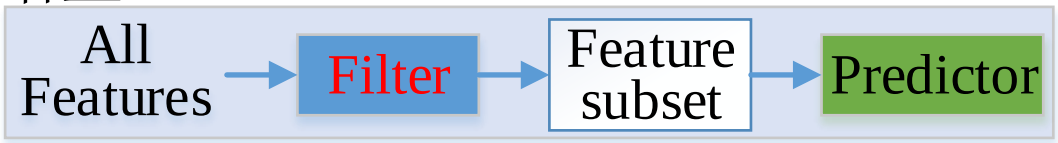
- 文本分类
- 朴素贝叶斯
- 朴素贝叶斯的生成模型
- 朴素贝叶斯理论
- 特征选择
- 文本分类评价

特征选择

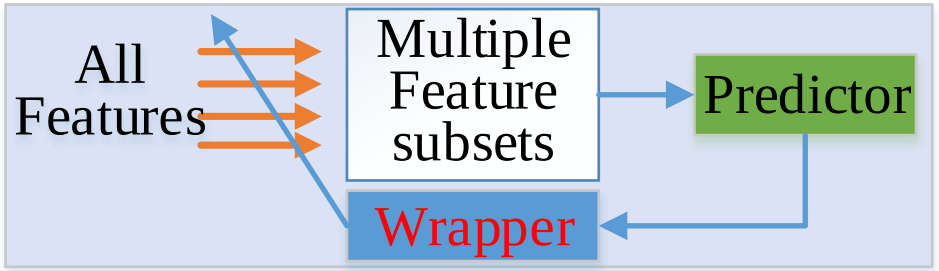
也称特征子集选择(Feature Subset Selection)，或属性选择(Attribute Selection)。指从已有的多个特征中选择特征子集使得学习器的性能最优化。

- 两个功能
  - 减少特征数量、降维，使模型泛化能力更强，减少过拟合
  - 增强对特征和特征值之间的理解
- 主要方法
  - 过滤式Filter
  - 包裹式Wrapper
  - 嵌入式Embedding

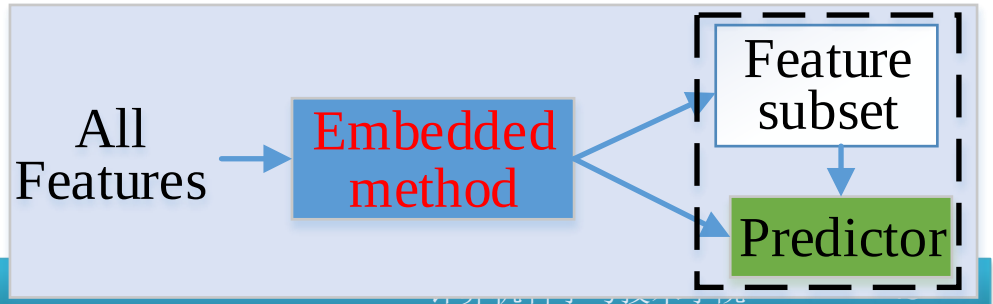
- 过滤式Filter
  - 首先对数据集进行特征选择，然后再训练学习器。两者独立
  - 互信息，卡方χ^2，信息增益



- 包裹式Wrapper
  - 直接把学习器的性能作为特征子集的评价准则。即特征选择的目的就是为给定的学习器选择最有利其性能的特征子集。
  - 完全搜索，启发式搜索



- 嵌入式Embedding
  - 前两者特征选择和学习器训练有明显的分别。
  - 嵌入式特征选择是将特征选择过程与学习器训练融为一体，两者在同一个优化过程中完成，即在学习器训练过程中自动进行特征选择。
  - Lasso，Ridge，深度学习



分类特征选择

- 文本分类中，通常要将文本表示在一个高维空间下，每一维对应一个词项。
- **特征选择**是从训练集出现的词项中选出一部分子集的过程。在文本分类过程仅仅使用这个子集作为特征。
- 特征选择有两个主要目的
  - 通过减少有效词汇的空间来提高分类器训练和应用的效率。这对于除NB之外其他的训练开销较大的分类器来说尤为重要。
  - 去除噪音特征，提高分类精度。

- 噪音特征(noise feature)指那些加入后反而会增加新数据上的分类错误率的特征。
  - 假定某个罕见词项(如arachnocentric)对某个类别(如China)不提供任何信息，但训练集中所有的arachnocentric恰好都出现在China类，那么学习后产生的分类器会将包含arachnocentric的测试文档误分到China类中去。
- 这种由于训练集的偶然性导出的不正确的泛化结果称为过学习(overfitting)。

特征选择算法

- 给定类别c，对词汇表中的每个词项t，计算效用指标$A(t,c)$，然后从中选择k个具有最高值的词项作为最后的特征，其它的词项被忽略。

```
SELECTFEATURES(𝔻, c, k)
1  V ← EXTRACTVOCABULARY(𝔻)
2  L ← []
3  for each t ∈ V
4  do A(t, c) ← COMPUTEFEATUREUTILITY(𝔻, t, c)
5     APPEND(L, ⟨A(t, c), t⟩)
6  return FEATURESWITHLARGESTVALUES(L, k)
```

图 13-6　选择 k 个最佳特征的基本特征选择算法

不同的特征选择方法

- 特征选择方法主要基于其所使用特征效用指标来定义。
- 特征效用指标
  - 频率法 – 选择高频词项
  - 互信息(Mutual information) – 选择具有最高互信息的词项
  - 卡方χ^2(Chi-square)

互信息(Mutual information)

- $A(t,c)$采用词项t和类别c的期望互信息(*Expected Mutual Information*)来计算
- MI给出的是词项所包含的有关类别的信息量及类别包含的有关词项的信息量
- 定义：

$$I(U;C)=\sum_{e_t\in\{1,0\}}\sum_{e_c\in\{1,0\}}P(U=e_t,C=e_c)\log_2\frac{P(U=e_t,C=e_c)}{P(U=e_t)P(C=e_c)}$$

  - U是一个二值随机变量，当文档包含词项t时，取值为e_t=1，否则取值为e_t=0。
  - C也是一个二值随机变量，当文档属于类别c时，它取值为e_c=1，否则取值为e_c=0。

χ^2卡方统计量

- χ^2统计量常常用于检测两个事件的独立性。
  - 两个事件 A 和 B 独立，是指两个事件 A、B的概率满足 $P(AB)=P(A)P(B)$或者 $P(A|B)=P(A)$且$P(B|A)=P(B)$。
- 在特征选择中，两个事件分别是指词项的出现和类别的出现。
- 度量两者独立性的缺乏程度，χ^2越大，独立性越小，相关性越大($N=A+B+C+D$)

$$\chi^2(t,c)=\frac{N(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)}$$

	c	~c
t	A	B
~t	C	D

提纲

- 文本分类
- 朴素贝叶斯
- 朴素贝叶斯的生成模型
- 朴素贝叶斯理论
- 特征选择
- 文本分类评价

分类评价

- 评价必须基于测试数据进行，而且该测试数据是与训练数据完全独立的 (通常两者样本之间无交集)
- 很容易通过训练在训练集上达到很高的性能 (比如记忆所有的测试集合)
- 常用指标：正确率、召回率、F_1值、分类精确率(classification accuracy)等
- 当对具有多个分类器的文档集进行处理时，往往需要计算出一个融合了每个分类器指标的综合指标。为实现这个目的，通常有宏平均和微平均两种做法：
  - 宏平均(macro averaging)是在类别之间求平均值，
  - 微平均(micro averaging)则是将每篇文档在每个类别上的判定放入一个缓冲池，然后基于这个缓冲池计算效果指标。

微平均 vs. 宏平均

- 对于一个类得到评价指标F_1
- 但是希望得到在所有类别上的综合性能
- 宏平均(Macroaveraging)
  - 对类别集合C 中的每个类都计算一个F_1 值
  - 对C个结果求平均Average these C numbers
- 微平均(Microaveraging)
  - 对类别集合C 中的每个类都计算TP、FP和FN
  - 将C中的这些数字累加
  - 基于累加的TP、FP、FN计算P、R和F_1

表13-8　宏平均和微平均的计算

	类别1			类别2			缓冲表	
	实际 yes	实际 no		实际 yes	实际 no		实际 yes	实际 no
判定yes	10	10	判定 yes	90	10	判定yes	100	20
判定 no	10	970	判定 no	10	890	判定 no	20	1860

注：“实际”表示实际上属于该类，“判定”表示的是分类器的判定情况。下例中，宏平均正确率为$[10/(10+10)+90/(10+90)]/2=(0.5+0.9)/2=0.7$，而微平均正确率为$100/(100+20)\approx 0.83$。

宏平均和微平均的适用范围

- 两者的计算结果可能会相差很大。宏平均对每个类别同等对待，而微平均则对每篇文档的判定结果同等对待。
- 由于F1值忽略判断正确的负例，所以它的大小主要由判断正确的正例数目所决定，所以在微平均计算中大类起支配作用。
  - 上例中，系统的微平均正确率(0.83)更接近c_2类的正确率(0.9)，而与c_1类的正确率(0.5)相差较大，这是因为c_2的大小是c_1的5 倍。
  - 因此，微平均实际上是文档集中大类上的一个效果度量指标。如果要度量小类上的效果，往往需要计算宏平均指标。

本讲要点

- 什么是文本分类？**Taxonomies and Classification**
- 什么是朴素贝叶斯分类器？

$$c_{map} = \arg\max_{c \in C} \hat{P}(c \mid d) \propto \arg\max \hat{P}(c) \prod_{1 \le k \le n_d} \hat{P}(t_k \mid c)$$

- 朴素贝叶斯分类器的生成模型
- 朴素贝叶斯分类器的性质
  - 条件独立性假设 & 位置独立性假设
- 特征选择：互信息、$\chi 2$ 统计量、词项频率
- 文本分类的评价：宏平均和微平均