Natural Language Processing I

6. 朴素贝叶斯

WU Xiaokun 吴晓堃

xkun.wu [at] gmail

2022/04/01

文本分类任务

垃圾邮件

这是不是垃圾邮件?

> 你想不过你的日收入入过万呢!那请马上加入我们的游戏吧! www.zxxzg.com 公平公证! 新手进来可领取首次登陆礼金。91x怯

文本风格

谁写了下面这段文字?

> 看来有很多人说，王二不存在。这件事叫人困惑的原因就在这里。
> 大家都说存在的东西一定不存在，这是因为眼前的一切都是骗局。
> 大家都说不存在的东西一定存在，比如王二，假如他不存在，这个名字是从哪里来的?

文本风格

谁写了下面这段文字?

> 看来有很多人说，王二不存在。这件事叫人困惑的原因就在这里。
> 大家都说存在的东西一定不存在，这是因为眼前的一切都是骗局。
> 大家都说不存在的东西一定存在，比如王二，假如他不存在，这个名字是从哪里来的?

> *写出《黄金时代》之前，我从未觉得自己写得好。–王小波*

正负面评价

情感分类：酒店评价

- “前台态度非常好！早餐很丰富，房间很干净。”
- “结果大失所望，灯光昏暗，空间狭小，房间有霉味。”

正负面评价

情感分类：酒店评价

- “前台态度非常好！早餐很丰富，房间很干净。”
- “结果大失所望，灯光昏暗，空间狭小，房间有霉味。”

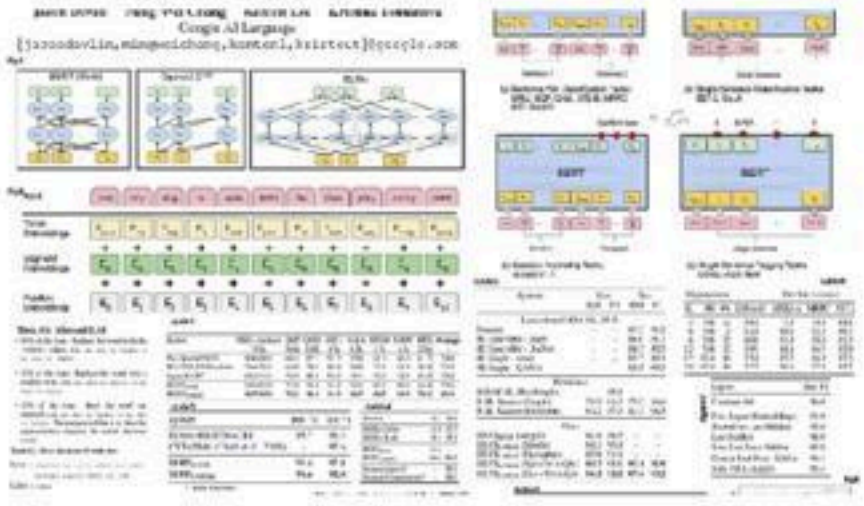
注意：“情感分类”也可以是客观评价

该生使用深度学习的方法，实现了一个文本情感分类系统。很好的完成了任务书规定的工作量，除按时完成外文翻译外，并能阅读一些自选资料，设计合理，有较强的实践动手能力，成果具有实际应用意义，论文结构合理，论述层次清晰，论文符合规范化要求。在答辩过程中能够对系统进行分析并得出合理正确的结论。

该生使用Unity3D框架完成了一个小游戏。在答辩过程中对问题不能正确回答。

学科分类

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding



学科分类与代码

520	计算机科学技术
52020	人工智能
5202020	自然语言处理

文本分类定义

输入

- 文档：d；固定的类别集合：$C = \{c_1, c_2, .., c_m\}$

输出

- 预测类别：$c \in C$

方法：预定义规则

领域专家根据**特征**集合给出的**规则**

- 垃圾邮件特征：黑名单 OR (“大礼包” AND “注册”)
- 例如：数模型，分支判定

方法：预定义规则

领域专家根据**特征**集合给出的**规则**

- 垃圾邮件特征：黑名单 OR (“大礼包” AND “注册”)
- 例如：数模型，分支判定

精度可以很高

- 如果“专家”确实对领域非常熟悉
- 人工经验：构建及维护成本高

方法：监督学习

输入

- 文档：d；固定的类别集合：$C = \{c_1, c_2, .., c_m\}$
- **标注好的训练集**：$(d_1, c_1), .., (d_n, c_n)$

输出

- **分类器**：$\gamma : d \to c$

回顾：**机器学习范式**

- 自动化（经验）规则提取：替代特征工程
- 能够预测新文档

朴素 Bayes 分类器

朴素贝叶斯

朴素贝叶斯 naïve （天真幼稚的） Bayes

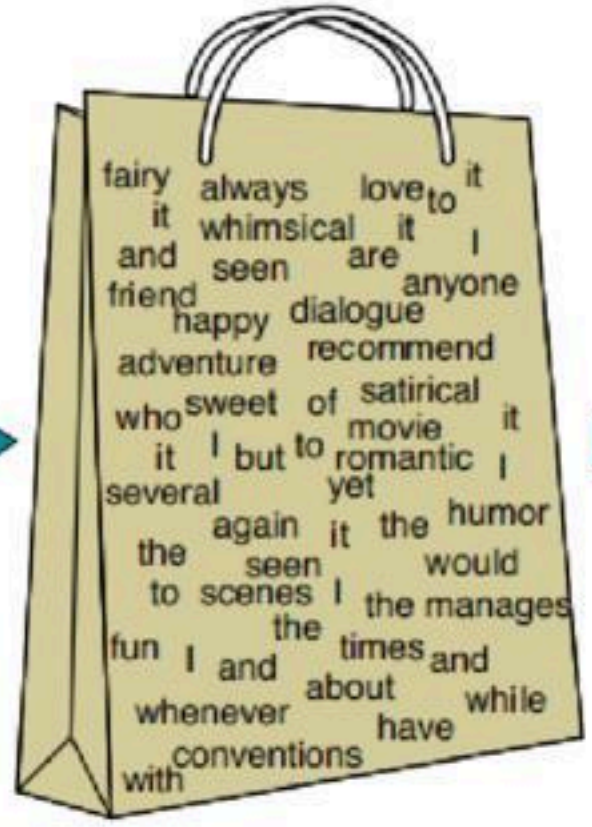
- 模型简单：假设相对朴素
- 理论依据：Bayes 规则

数据表示也很简单

- **词袋 bag-of-words**

词袋

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!



it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1
...	..

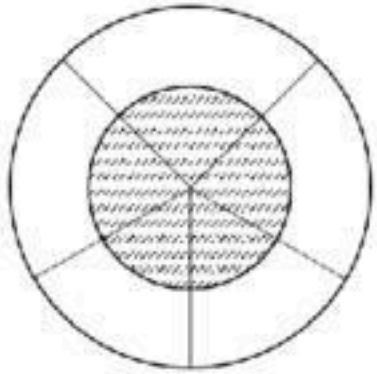
- 无序的词汇集合
- 基本统计信息：**频率**

Bayes 规则

[Bayes 1763] **Bayes 推断 inference**

- 全概率公式

$$P(AB) = P(A|B)P(B) = P(B|A)P(A)$$

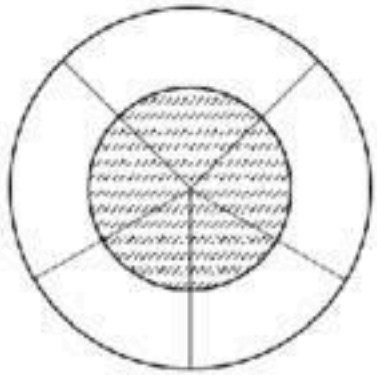


Bayes 规则

[Bayes 1763] **Bayes 推断 inference**

- 全概率公式

$$P(AB) = P(A|B)P(B) = P(B|A)P(A)$$



- Bayes 规则：即条件概率公式

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

- [Mosteller 1964] 应用于文本分类问题

文档分类估计

给定文档d，估计其类别c作为输出；但优化目标是概率

$$\begin{aligned}\hat{c} &= \arg\max_{c \in C} P(c|d) \\ &= \arg\max_{c \in C} \frac{P(d|c)P(c)}{P(d)}\end{aligned}$$

文档分类估计

给定文档d，**估计**其类别c作为输出；但优化**目标**是概率

$$\hat{c} = \arg\max_{c \in C} P(c|d)$$
$$= \arg\max_{c \in C} \frac{P(d|c)P(c)}{P(d)}$$

注意：$P(d)$不随文档类别c变化，即**没有参数依赖关系**

- $P(d)$仅取决于文档的分布；计算不同c时d不变

$$\hat{c} = \arg\max_{c \in C} P(d|c)P(c)$$

最大后验概率

最大后验 Maximum A Posterior (MAP) 概率

- A posteriori, Latin for “from the latter”

$$\hat{c}_{MAP} = \arg\max_{c \in C} \overbrace{P(d|c)}^{\text{likelihood}} \overbrace{P(c)}^{\text{prior}}$$

最大后验概率

最大后验 Maximum A Posterior (MAP) 概率

- A posteriori, Latin for “from the latter”

$$\hat{c}_{MAP} = \arg\max_{c \in C} \overbrace{P(d|c)}^{\text{likelihood}} \overbrace{P(c)}^{\text{prior}}$$

- **先验 prior 概率**：对被估计量的经验假设，与输出一致
  - 通常是因为还没有观测到（足够多的）数据

最大后验概率

最大后验 Maximum A Posterior (MAP) 概率

- A posteriori, Latin for “from the latter”

$$\hat{c}_{MAP} = \arg\max_{c \in C} \overbrace{P(d|c)}^{\text{likelihood}} \overbrace{P(c)}^{\text{prior}}$$

- **先验 prior 概率**：对被估计量的经验假设，与输出一致
  - 通常是因为还没有观测到（足够多的）数据
- **似然 likelihood**：“可能性”的一种较文艺的说法
  - 并非“概率”：给定样本d，参数（实数）c（相对其他参数选择）的可能性

解决似然的理论问题

似然是一种经验（而非客观）规律的描述，无法识别“偶发事件”

$$P(w_n|w_{n-1}) = \frac{\Gamma(w_{n-1}w_n)}{\sum_w \Gamma(w_{n-1}w)} = \frac{\Gamma(w_{n-1}w_n)}{\Gamma(w_{n-1})}$$

$$P(\text{下雨}|\text{吴老师来上课}) = \frac{\Gamma(\text{下雨}, \text{吴老师来上课})}{\Gamma(\text{吴老师来上课})}$$

$$= \frac{9}{12} = .75$$

解决似然的理论问题

似然是一种**经验（而非客观）规律**的描述，无法识别“偶发事件”

$$P(w_n|w_{n-1}) = \frac{\Gamma(w_{n-1}w_n)}{\sum_w \Gamma(w_{n-1}w)} = \frac{\Gamma(w_{n-1}w_n)}{\Gamma(w_{n-1})}$$

$$P(\text{下雨}|\text{吴老师来上课}) = \frac{\Gamma(\text{下雨}, \text{吴老师来上课})}{\Gamma(\text{吴老师来上课})}$$

$$= \frac{9}{12} = .75$$

先验概率的重要性：先验是对**客观规律的经验总结**

$$\hat{P}(\text{下雨}) \propto \overbrace{P(\text{吴老师来上课}|\text{下雨})}^{\text{likelihood}} \overbrace{P(\text{下雨})}^{\text{prior}}$$

MAP 概率举例

例如吸烟导致肺癌的概率

$$\hat{c}_{MAP} = \arg\max_{c \in C} \overbrace{P(d|c)}^{\text{likelihood}} \overbrace{P(c)}^{\text{prior}}$$

- 没拿到检测报告前，只能凭经验给出一个相对普遍的关联性
- 检测报告会**对先验概率做出修正**，最终导致后验概率的改变

生成模型

朴素 Bayes 通常被认为是一种**生成模型 generative model**

- 模型在公式（右侧）上隐含着生成数据的逻辑假设

$$\hat{c}_{MAP} = \arg\max_{c \in C} P(d|c)P(c)$$

1. 首先从$P(c)$中采样出类别
2. 然后从$P(d|c)$中采样出文本

生成模型

朴素 Bayes 通常被认为是一种**生成模型 generative model**

- 模型在公式（右侧）上隐含着生成数据的逻辑假设

$$\hat{c}_{MAP} = \arg\max_{c \in C} P(d|c)P(c)$$

1. 首先从$P(c)$中采样出类别
2. 然后从$P(d|c)$中采样出文本

但直接生成文档过于复杂：文档空间几乎是无限维

- 转而生成文档的**有限特征描述**

文档特征

文档可以看成由一系列**特征 feature** 构成

$$\hat{c}_{MAP} = \arg\max_{c \in C} \overbrace{P(f_1, f_2, .., f_n|c)}^{\text{likelihood}} \overbrace{P(c)}^{\text{prior}}$$

- 特征可以理解为数据中**有效信息的压缩形式**，及其（几何意义上的）变换
  - 最简单的特征就是（字/词：频率）的向量表示

文档特征

文档可以看成由一系列**特征 feature** 构成

$$\hat{c}_{MAP} = \arg\max_{c \in C} \overbrace{P(f_1, f_2, .., f_n|c)}^{\text{likelihood}} \overbrace{P(c)}^{\text{prior}}$$

- 特征可以理解为数据中**有效信息的压缩形式**，及其（几何意义上的）变换
  - 最简单的特征就是（字/词：频率）的向量表示

问题：计算量仍然过大

- 特征集合所有可能的组合：$O(|X|^n|C|)$

两个简化假设

词袋表示假设：特征（比如词）只考虑类别，不考虑序列位置关系

两个简化假设

词袋表示假设：特征（比如词）只考虑类别，不考虑序列位置关系

朴素 Bayes假设：特征之间条件独立，不考虑条件（参数）的相关性

$$P(f_1, f_2, .., f_n|c) = P(f_1|c) \cdot P(f_2|c) \cdot .. \cdot P(f_n|c)$$

两个简化假设

词袋表示假设：特征（比如词）只考虑类别，不考虑序列位置关系

朴素 Bayes假设：特征之间条件独立，不考虑条件（参数）的相关性

$$P(f_1, f_2, .., f_n|c) = P(f_1|c) \cdot P(f_2|c) \cdot .. \cdot P(f_n|c)$$

朴素 Bayes **分类器**

$$\begin{aligned}\hat{c}_{NB} &= P(f_1, f_2, .., f_n|c)P(c) \\ &= \arg\max_{c \in C} P(c) \prod_{f \in F} P(f|c)\end{aligned}$$

NB分类器应用：文本

文本由字符串构成：$d = w_1 w_2 .. w_n$

- 遍历文本中的每个词

$$\hat{c}_{NB} = \arg\max_{c \in C} P(c) \prod_i P(w_i|c)$$

NB分类器应用：文本

文本由字符串构成：$d = w_1 w_2 .. w_n$

- **遍历**文本中的每个词

$$\hat{c}_{NB} = \arg\max_{c \in C} P(c) \prod_i P(w_i|c)$$

问题：大量概率值相乘可能导致**浮点数下溢**

- 概率取值范围：$[0, 1]$
- 词数很多的文本非常常见

对数空间计算

$$\begin{aligned}\hat{c}_{NB} &= \arg\max_{c \in C} P(c) \prod_i P(w_i|c) \\ &= \arg\max_{c \in C} \log \left(P(c) \prod_i P(w_i|c) \right) \\ &= \arg\max_{c \in C} \log P(c) + \sum_i \log P(w_i|c)\end{aligned}$$

- 取对数不改变最大值：单调增函数**保持序关系**
- NB分类器是**线性模型**：（对数空间里）输入的线性组合

训练朴素 Bayes 分类器

计算还缺什么?

$$\hat{c}_{NB} = \arg\max_{c \in C} \log P(c) + \sum_{f \in F} \log P(f|c)$$

- 如何估计$P(c)$和$P(f|c)$?

计算还缺什么?

$$\hat{c}_{NB} = \arg\max_{c \in C} \log P(c) + \sum_{f \in F} \log P(f|c)$$

- 如何估计$P(c)$和$P(f|c)$?

简化假设：特征f_i就是文档中的词w_i在词袋中**是否出现**

- 特征的向量表示f：长度为词袋容量

计算还缺什么?

$$\hat{c}_{NB} = \arg\max_{c \in C} \log P(c) + \sum_{f \in F} \log P(f|c)$$

- 如何估计$P(c)$和$P(f|c)$?

简化假设：特征f_i就是文档中的词w_i在词袋中**是否出现**

- 特征的向量表示f：长度为词袋容量
- 因此改为估计$P(w_i|c)$

回顾：最大似然估计

MLE：简单使用数据频率作为概率估计值

- 经典统计学中的**频率学派 frequentist**

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

- N_c：训练集中标注为c的文档数量；N_{doc}：文档总数

回顾：最大似然估计

MLE：简单使用数据频率作为概率估计值

- 经典统计学中的**频率学派 frequentist**

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

- N_c：训练集中标注为c的文档数量；N_{doc}：文档总数

$$\hat{P}(w_i|c) = \frac{\Gamma(w_i, c)}{\sum_{w \in V} \Gamma(w, c)}$$

- $\Gamma(w_i, c)$：在类别c中w_i出现的次数；V：所有类别的词汇总表

最大似然的问题（之一）

（**数据不足**问题）如果训练集中某个类别缺少一个词

- 例如：正负面两类评价中，正面的评价里没有“还行”，但负面的评价出现了

$$\hat{P}(\text{“还行”}|+) = \frac{\Gamma(\text{“还行”}, +)}{\sum_{w \in V} \Gamma(w, +)} = 0$$

最大似然的问题（之一）

（**数据不足**问题）如果训练集中某个类别缺少一个词

- 例如：正负面两类评价中，正面的评价里没有“还行”，但负面的评价出现了

$$\hat{P}(\text{“还行”}|+) = \frac{\Gamma(\text{“还行”}, +)}{\sum_{w \in V} \Gamma(w, +)} = 0$$

那么MAP概率必定为0，相当于**一票否决**

$$\hat{c}_{MAP} = \arg\max_{c \in C} \hat{P}(c) \prod_i \hat{P}(w_i|c)$$

- 无论这个文档中其他内容写得多好，只要出现“还行”两个字就被判定为负面

加一平滑

$$\hat{P}(w_i|c) = \frac{\Gamma(w_i, c) + 1}{\sum_{w \in V}(\Gamma(w, c) + 1)}$$
$$= \frac{\Gamma(w_i, c) + 1}{\sum_{w \in V}\Gamma(w, c) + |V|}$$

- **朴素 Bayes 文本分类任务**通常使用加一平滑
  - 尽管其他语言模型大多选用更复杂的平滑算法，如KN

加一平滑

$$\hat{P}(w_i|c) = \frac{\Gamma(w_i, c) + 1}{\sum_{w \in V}(\Gamma(w, c) + 1)}$$
$$= \frac{\Gamma(w_i, c) + 1}{\sum_{w \in V}\Gamma(w, c) + |V|}$$

- **朴素 Bayes 文本分类任务**通常使用加一平滑
  - 尽管其他语言模型大多选用更复杂的平滑算法，如KN

再次强调：V是所有类别的词汇总表，否则无法**归一化**

未知词汇

如何处理**未知**词汇？具体问题、具体分析

- 出现在测试集，但训练集和词汇表里没有

需要对未知词汇的分布建模吗？即统计每个类别中未知词数

未知词汇

如何处理**未知**词汇？具体问题、具体分析

- 出现在测试集，但训练集和词汇表里没有

需要对未知词汇的分布建模吗？即统计每个类别中未知词数

实际应用（情感分析）：**直接忽略掉**

- 从测试文本中删除，也就不做概率估计

未知词汇

如何处理**未知**词汇？具体问题、具体分析

- 出现在测试集，但训练集和词汇表里没有

需要对未知词汇的分布建模吗？即统计每个类别中未知词数

实际应用（情感分析）：**直接忽略掉**

- 从测试文本中删除，也就不做概率估计

为什么不对未知词汇建模？没意义

- 通常来说，决定文档类别的是**关键字**
- 知道哪个类未知词汇多通常不解决问题

停用詞

停用詞 stop words：非常高频但对语言影响很小，主要是虚词

- 冠词：`the`, `a`
- 叹词：啊, 呢, 乎

一些任务会**忽略**停用词

- 删除频率最高的10到50个词
- 删除停用词黑名单里面的词

停用詞

停用詞 stop words：非常高频但对语言影响很小，主要是虚词

- 冠词：`the`, `a`
- 叹词：啊, 呢, 乎

一些任务会**忽略**停用词

- 删除频率最高的10到50个词
- 删除停用词黑名单里面的词

但**文本分类任务通常保留**停用词

- 很多语气词可以表达情感因素

情感分析

模型修正

对情感相关的任务：

- 词**是否出现 occurrence** 比词频 frequency 更重要
  - “非常好”比5个“还算好”要更好

模型修正

对情感相关的任务：

- 词**是否出现 occurrence** 比词频 frequency 更重要
  - “非常好”比5个“还算好”要更好

二元NB分类器

- 词的计数截取为$\{0, 1\}$

实验：情感分析

否定词

否定词：否定语义

```
I really like this movie
I really don't like this movie
```

- 主要否定的是**动词支配的分句**

否定词

否定词：否定语义

```
I really like this movie
I really don't like this movie
```

- 主要否定的是**动词支配的分句**

[Pang 2002] 将**分句中每个词**都替换成否定形式

```
didn't like this movie , but I
didn't NOT_like NOT_this NOT_movie , but I
```

- 因此，词汇表扩充为2倍

否定词

否定词：否定语义

```
I really like this movie
I really don't like this movie
```

- 主要否定的是**动词支配的分句**

[Pang 2002] 将**分句中每个词**都替换成否定形式

```
didn't like this movie , but I
didn't NOT_like NOT_this NOT_movie , but I
```

- 因此，词汇表扩充为2倍
- 以标点作为（分句）终止点
  - 更好的方法：同时训练分句器

情感词典

训练数据量不足？引入预标注的**常用情感词典**

- [Stone 1966] General Inquirer：相当于添加专家系统规则
- [Pennebaker 2007] LIWC

例如：《客服人员标准礼貌用语》、《毕业设计（论文）评分标准》

情感词典

训练数据量不足？引入预标注的**常用情感词典**

- [Stone 1966] General Inquirer：相当于添加专家系统规则
- [Pennebaker 2007] LIWC

例如：《客服人员标准礼貌用语》、《毕业设计（论文）评分标准》

使用方法：将每个使用的词视作一个**额外特征**

- 特征的向量表示 f：长度为词袋容量 + 词袋用词数
- 训练集的代表性越差，词典词汇就越重要

小结

- 名为“naïve”，但并不简单，译为朴素
- 参数少：运行速度非常快，存储要求低
  - 不容易过拟合：数据量少时很有效
- 如果条件独立假设确实成立，则是最优解
- 通常作为很可靠的基准实现

NB作为语言模型

NB与语言模型

NB分类器可以使用任何形式的特征

- 词，URL，神经网络中的特征
- 本节只考虑用**词作为特征**

NB与语言模型

NB分类器可以使用任何形式的特征

- 词，URL，神经网络中的特征
- 本节只考虑用**词作为特征**

类别上的一元语法（首先考虑似然）

- 每个类别按一元语法建模：$P(w|c)$
  - 注意：类别c是固定的，是讨论范畴（而非参数）

NB与语言模型

NB分类器可以使用任何形式的特征

- 词，URL，神经网络中的特征
- 本节只考虑用**词作为特征**

类别上的一元语法（首先考虑似然）

- 每个类别按一元语法建模：$P(w|c)$
  - 注意：类别c是固定的，是讨论范畴（而非参数）
- 句子看成词的集合

$$P(s|c) = \prod_i P(w_i|c)$$

似然计算示例

w	P(w|+)	P(w|-)
I	0.1	0.2
love	0.1	0.001
this	0.01	0.01
fun	0.05	0.005
film	0.1	0.1

$$P(\text{“I love this fun film”}|+) = 0.1 \times 0.1 \times 0.01 \times 0.05 \times 0.1 = 5 \cdot 10^{-7}$$

$$P(\text{“I love this fun film”}|-) = 0.2 \times 0.001 \times 0.01 \times 0.005 \times 0.1 = 1 \cdot 10^{-9}$$

- 这里可以看出对数空间计算的重要性：防止**下溢出**

似然计算示例

w	P(w|+)	P(w|-)
I	0.1	0.2
love	0.1	0.001
this	0.01	0.01
fun	0.05	0.005
film	0.1	0.1

$$P(\text{“I love this fun film”}|+) = 0.1 \times 0.1 \times 0.01 \times 0.05 \times 0.1 = 5 \cdot 10^{-7}$$

$$P(\text{“I love this fun film”}|-) = 0.2 \times 0.001 \times 0.01 \times 0.005 \times 0.1 = 1 \cdot 10^{-9}$$

- 这里可以看出对数空间计算的重要性：防止**下溢出**

注意：MAP概率计算还需要考虑**先验**（即类别的概率），即$P(+), P(-)$

评测：精确率，召回率，F度量

回顾：两类误差

假阳 False positives (Type I errors)

- 查出错误信息
- 提高准确度 accuracy、精度 precision

假阴 False negatives (Type II errors)

- 没有查出正确信息
- 提高覆盖率 coverage、召回率 recall

回顾：两类误差

假阳 False positives (Type I errors)

- 查出错误信息
- 提高准确度 accuracy、精度 precision

假阴 False negatives (Type II errors)

- 没有查出正确信息
- 提高覆盖率 coverage、召回率 recall

评测：模型的预测性能，即预测正确率

混淆矩阵

混淆矩阵 confusion matrix：预测与标注对比的可视化

	T	F
P	True Positive (TP)	False Positive (FP)
N	False Negative (FN)	True Negative (TN)

- TP和TN是两种正确的预测

混淆矩阵

混淆矩阵 confusion matrix：预测与标注对比的可视化

	T	F
P	True Positive (TP)	False Positive (FP)
N	False Negative (FN)	True Negative (TN)

- TP和TN是两种正确的预测

评测：计算预测的正确率

- 一定与**正确的预测**（TP和TN）有关

精确度、召回率

精确度 precision：预测为+的结果中，正确（实际确实是）的百分比

召回率 recall：标注为T的数据中，正确检测出来（预测阳性）的百分比

	T	F	
P	TP	FP	$P = \frac{TP}{TP+FP}$
N	FN	TN	
	$R = \frac{TP}{TP+FN}$		

注意：两者都只关注TP，而**把容易带来干扰的TN排除**了

F度量

F度量 F-measure：精确度、召回率的加权调和平均

$$F_\alpha = \frac{1}{\alpha \frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{\beta^2 + 1}{\frac{1}{P} + \beta^2 \frac{1}{R}}$$

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

- $\beta > 1$：强调召回率；$\beta < 1$：强调精确度

F度量

F度量 F-measure：精确度、召回率的加权调和平均

$$F_\alpha = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{\beta^2+1}{\frac{1}{P} + \beta^2\frac{1}{R}}$$

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

- $\beta > 1$：强调召回率；$\beta < 1$：强调精确度

最常用的是F_1度量：$\beta = 1$

$$F_1 = \frac{2PR}{P+R}$$

三类举例

邮件分类：紧急、普通、垃圾

	T_1	T_2	T_3
P_1	8	10	1
P_2	5	60	50
P_3	3	30	200

如何评测?

多个类：混淆矩阵

前面两个类看成："属于第一个类"和"不属于第一个类"

- 也可以看成："属于第一个类"和"属于**非第一个类**"

	T_1	T_2
P_1	T_1P_1	T_2P_1
P_2	T_1P_2	T_2P_2

多个类：混淆矩阵

前面两个类看成："属于第一个类"和"不属于第一个类"

- 也可以看成："属于第一个类"和"属于**非第一个类**"

	T_1	T_2
P_1	T_1P_1	T_2P_1
P_2	T_1P_2	T_2P_2

推广到多个类：

	T_1	T_2	...	T_C
P_1	T_1P_1	T_2P_1	...	T_CP_1
P_2	T_1P_2	T_2P_2	...	T_CP_2
...			...	
P_C	T_1P_C	T_2P_C	...	T_CP_C

多个类：度量

	T_1	T_2	...	T_C	
P_1	T_1P_1	T_2P_1	...	T_CP_1	$P_{i1}=\frac{T_iP_1}{\sum_i T_iP_1}$
P_2	T_1P_2	T_2P_2	...	T_CP_2	
...			...		
P_C	T_1P_C	T_2P_C	...	T_CP_C	$P_{iC}=\frac{T_iP_C}{\sum_i T_iP_C}$
	$R_{1j}=\frac{T_1P_j}{\sum_j T_1P_j}$			$R_{Cj}=\frac{T_CP_j}{\sum_j T_CP_j}$	$A=\frac{\sum_i T_iP_i}{\sum_{ij} T_iP_j}$

- 总共$2C+1$个度量

多个类：度量

	T_1	T_2	...	T_C	
P_1	T_1P_1	T_2P_1	...	T_CP_1	$P_{i1}=\frac{T_iP_1}{\sum_i T_iP_1}$
P_2	T_1P_2	T_2P_2	...	T_CP_2	
...			...		
P_C	T_1P_C	T_2P_C	...	T_CP_C	$P_{iC}=\frac{T_iP_C}{\sum_i T_iP_C}$
	$R_{1j}=\frac{T_1P_j}{\sum_j T_1P_j}$			$R_{Cj}=\frac{T_CP_j}{\sum_j T_CP_j}$	$A=\frac{\sum_i T_iP_i}{\sum_{ij} T_iP_j}$

- 总共$2C+1$个度量

问题：如何综合考虑多个精确度或召回率？

宏平均、微平均

	T_1	T_2	…	T_C	
P_1	T_1P_1	T_2P_1	…	T_CP_1	$P_{i1}=\frac{T_iP_1}{\sum_i T_iP_1}$
P_2	T_1P_2	T_2P_2	…	T_CP_2	
…			…		
P_C	T_1P_C	T_2P_C	…	T_CP_C	$P_{iC}=\frac{T_iP_C}{\sum_i T_iP_C}$
	$R_{1j}=\frac{T_1P_j}{\sum_j T_1P_j}$			$R_{Cj}=\frac{T_CP_j}{\sum_j T_CP_j}$	$A=\frac{\sum_i T_iP_i}{\sum_{ij} T_iP_j}$

宏平均 macroaveraging：对每个类分别计算度量，然后算**类间平均**

- 对角线**每个元素**作为TP，分别构造（二类）混淆矩阵

宏平均、微平均

	T_1	T_2	…	T_C	
P_1	T_1P_1	T_2P_1	…	T_CP_1	$P_{i1}=\frac{T_iP_1}{\sum_i T_iP_1}$
P_2	T_1P_2	T_2P_2	…	T_CP_2	
…			…		
P_C	T_1P_C	T_2P_C	…	T_CP_C	$P_{iC}=\frac{T_iP_C}{\sum_i T_iP_C}$
	$R_{1j}=\frac{T_1P_j}{\sum_j T_1P_j}$			$R_{Cj}=\frac{T_CP_j}{\sum_j T_CP_j}$	$A=\frac{\sum_i T_iP_i}{\sum_{ij} T_iP_j}$

宏平均 macroaveraging：对每个类分别计算度量，然后算**类间平均**

- 对角线**每个元素**作为TP，分别构造（二类）混淆矩阵

微平均 microaveraging：将所有数据收集到同一混淆矩阵，然后计算度量

- 对角线元素**加总**作为TP；将宏平均构造的矩阵**加总**得到（二类）混淆矩阵

三类举例：宏、微平均

	T_1	T_2	T_3
P_1	8	10	1
P_2	5	60	50
P_3	3	30	200

	T_1	F_1
P_1	8	11
N_1	8	340

	T_2	F_2
P_2	60	55
N_2	40	212

	T_3	F_3
P_3	200	33
N_3	51	83

	T	F
P	268	99
N	99	635

$P_1 = \frac{8}{8+11} = .42$

$P_2 = \frac{60}{60+55} = .52$

$P_3 = \frac{200}{200+33} = .86$

$P_{micro} = \frac{268}{268+99} = .73$

$P_{macro} = \frac{.42+.52+.86}{3} = .60$

三类举例：宏、微平均

	T_1	T_2	T_3
P_1	8	10	1
P_2	5	60	50
P_3	3	30	200

	T_1	F_1
P_1	8	11
N_1	8	340

	T_2	F_2
P_2	60	55
N_2	40	212

	T_3	F_3
P_3	200	33
N_3	51	83

	T	F
P	268	99
N	99	635

$P_1 = \frac{8}{8+11} = .42$

$P_2 = \frac{60}{60+55} = .52$

$P_3 = \frac{200}{200+33} = .86$

$P_{micro} = \frac{268}{268+99} = .73$

$P_{macro} = \frac{.42+.52+.86}{3} = .60$

- 微平均容易**被频率高的类主导**（此例中的T_3P_3）：计数不加区分

交叉验证

避免过拟合：信息泄露，即间接使用测试集调参

例如4折交叉验证的数据划分：

=	=	=	V
V	=	=	=
=	V	=	=
=	=	V	=

Review

本章内容

文本分类任务。朴素 Bayes 分类器。朴素 Bayes 分类器用于情感分析。朴素 Bayes 作为语言模型。评测朴素 Bayes 分类器。生成、判别模型。

重点：Bayes 规则、最大后验估计、朴素 Bayes 分类器；评测朴素 Bayes 分类器；生成、判别模型的区别。

难点：最大似然、最大后验估计的理论对比。

学习目标

- 理解文本分类任务的监督学习定义。
- 理解 Bayes 规则、最大后验估计。
- 理解朴素 Bayes 分类器的假设、构造、计算方法。
- 理解多类评测的计算方法：宏平均、微平均
- 理解生成、判别模型的区别。

问题

简述文本分类任务的监督学习定义。

简述应用 Bayes 规则对文档分类问题的最大后验估计。

简述朴素 Bayes 分类器的两个假设、构造与计算方法。

简述多类评测的计算方法：宏平均、微平均。

简述生成、判别模型的区别。