



南京大學
NANJING UNIVERSITY



自然语言处理

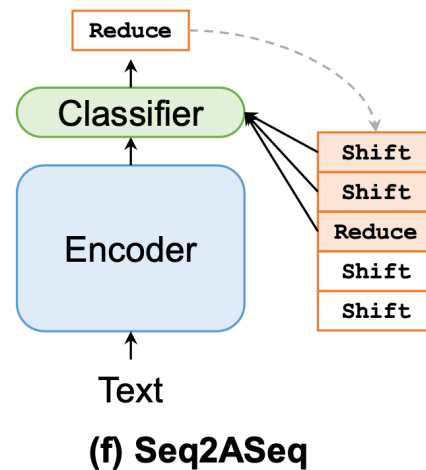
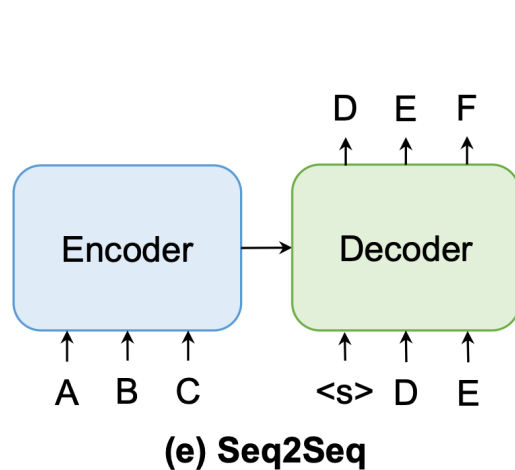
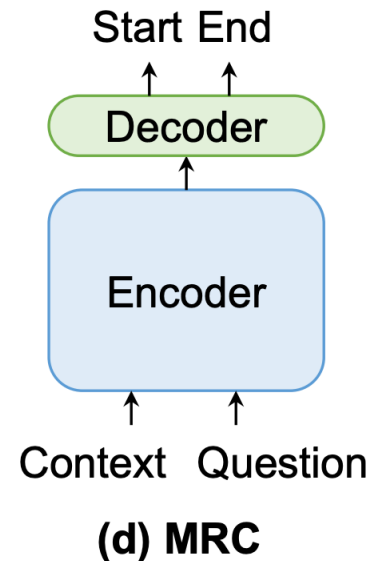
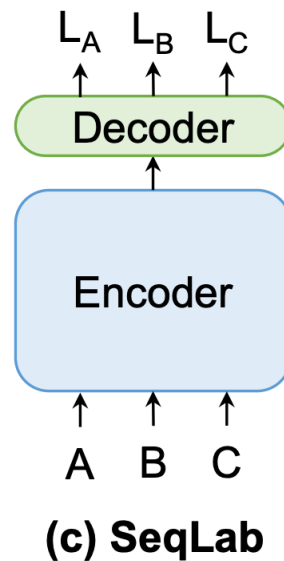
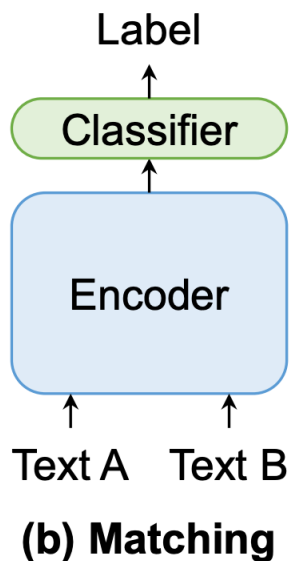
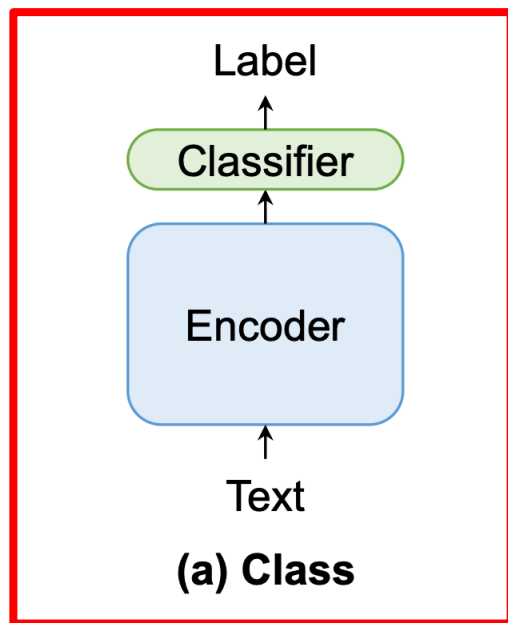
文本分类

吴震

南京大学人工智能学院
南京大学自然语言处理研究组

2023年3月

自然语言处理中典型的任务形式



- 背景知识
- 基于统计学习的文本分类
 - Naïve Bayes
 - Perception
 - Logistic Regression
- 基于深度学习的文本分类



01



背景知识

BACKGROUND

分类 (CLASSIFICATION)

- 对输入进行自动的决策，并归到对应的类别中
 - Document → category
 - Image of digit → digit
 - Image of object → object type
 - Query + webpages → best match
 - Symptoms → diagnosis
 -

- 将文本归类到预定义的某一个或某几个语义标签中
 - 多类别文本分类 (multi-class) : 每个文本只有一个类别标签
 - 多标签文本分类 (multi-label) : 每个文本可以有多个类别标签
- 形式上, 分类器 f 将输入文本 x (样本空间 X) 映射为标签 y (标签空间 Y)
 - 样本空间 X : 所有样本构成的集合
 - 标签空间 Y : 所有标签构成的集合
 - x : 单个文本
 - y : 文本 x 对应的标签 (一个或多个)

$$\operatorname{argmax}_y P(y|x)$$

- 情感分类

Google Product Search



HP Officejet 6500A Plus e-All-in-One Color Ink-jet - Fax / copier / printer / scanner

\$89 online, \$100 nearby ★★★★★ 377 reviews

September 2010 - Printer - HP - Inkjet - Office - Copier - Color - Scanner - Fax - 250 sh

Reviews

Summary - Based on 377 reviews



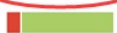
What people are saying

ease of use



"This was very easy to setup to four computers."

value



"Appreciate good quality at a fair price."

setup



"Overall pretty easy setup."

customer service



"I DO like honest tech support people."

size



"Pretty Paper weight."

mode



"Photos were fair on the high quality mode."

colors



"Full color prints came out with great quality."

● 新闻分类

課前小複習—認識新聞分類


政治	旅遊
社會	科技
娛樂	教育
運動	健康
國際	生活


模型项目 文本分类

重磅历史决议的几个细节	474万 🔥🔥	-文化
直播中 狮子座流星雨	466万 🔥	-文化
幼师称希望大连疫情越多越好 被行拘	453万	-社会
欧莱雅客服:李佳琦说低价就是低价吗	409万	-社会
江苏海域发生5.0级地震 上海有震感	419万	-地理

第十八期 | 新闻资讯自动分类

● 垃圾邮件过滤

Domain Focus: example.com ( change focus)

My Account Domain Center FAQ  Sign Out

Messages

- Quarantined Email
- Delivery Failures
- Email Statistics

Settings

- Email Addresses
- MX Guarddog Servers
- Your Email Servers
- Aggression
- Language
- Timezone




Filters


- Blacklisting
- Whitelisting





































Domains / Licensing

- Domain Management
- Invoices
- Earn Tokens
- Earn Tokens x2
- Buy Tokens

Email Addresses Catch-All LDAP Sync cPanel Sync

 Add New Email  Bulk Add Email  Add New Alias

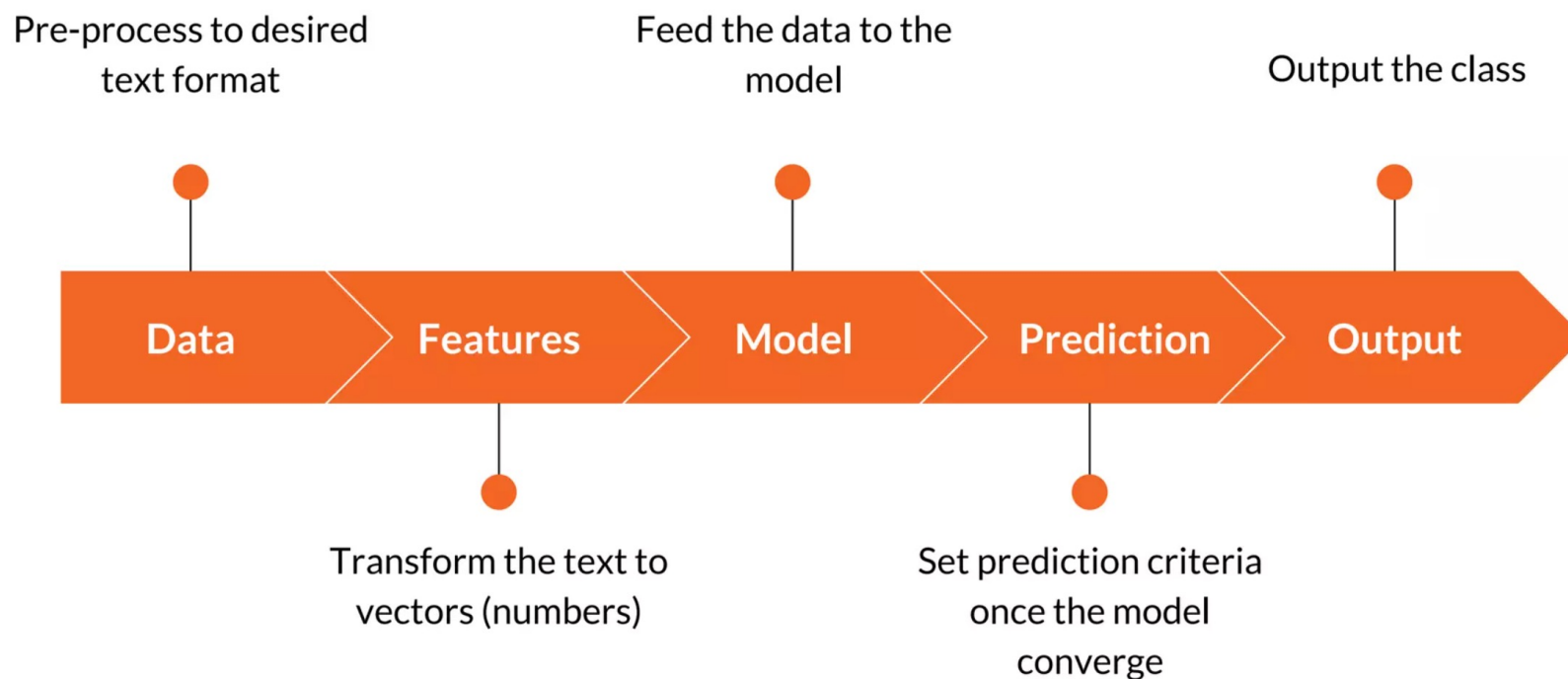
<< < 1 > >> >>> email search <<< 

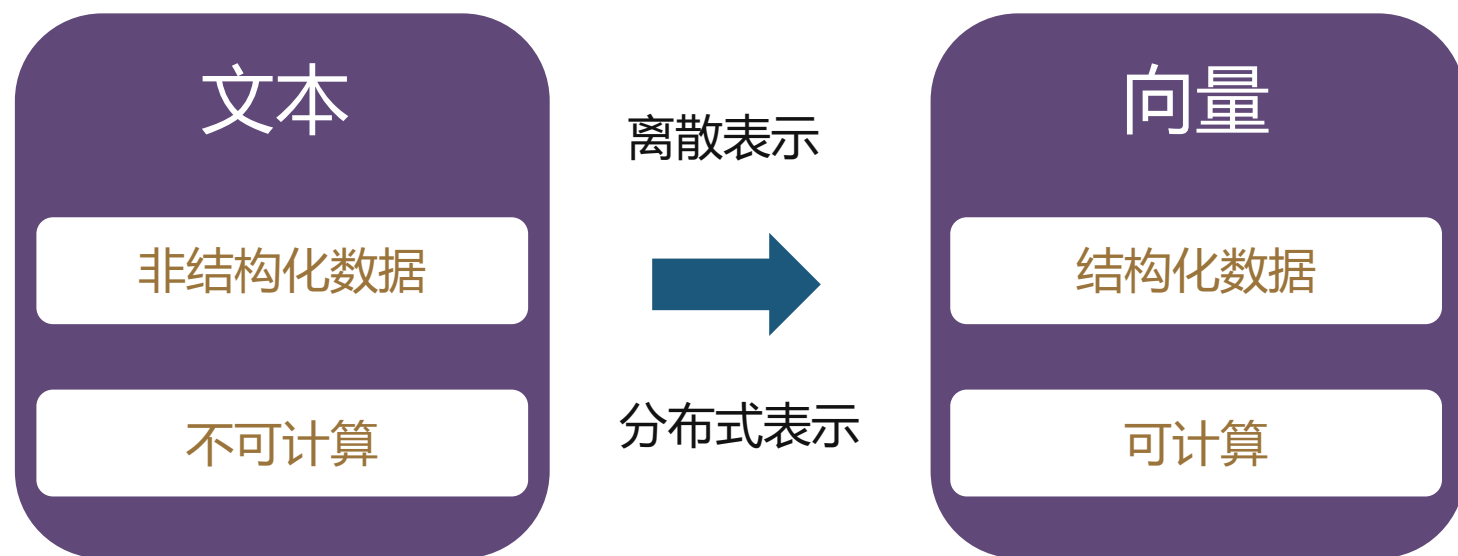
Email Address	Name	Date Added
   charles@example.com		Jan 30, 2016 
   chris@example.com		Jan 30, 2016 
   info@example.com		Jan 30, 2016 
   postmaster@example.com	Frank Lamb	Jan 30, 2016 
   products@example.com		Jan 30, 2016 
   returns@example.com		Jan 30, 2016 
   sales@example.com		Jan 30, 2016 
   sandi@example.com		Jan 30, 2016 
   santa@example.com		Jan 30, 2016 

<< < 1 > >>

- 主要步骤

- 特征表示 (Feature representation)
- 建模 (Modeling)
- 训练 (Training)
- 推理 (Inference)





- 单词表示：One-hot编码

- 将所有的单词构成一个词表，给每个词编码一个索引，根据索引进行one-hot表示

句子1：我/有/一个/苹果
句子2：我/明天/去/一个/地方
句子3：你/到/一个/地方
句子4：我/有/我/最爱的/地方



我	1	0	0	0	0	0	0	0	0	0
有	0	1	0	0	0	0	0	0	0	0
...
最爱的	0	0	0	0	0	0	0	0	0	1

- 词袋模型 (Bag-of-Words)

The Bag of Words Representation

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!

15



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
...	...

- 词袋模型

- 用一个词表维度的向量表示文本，如果文本中包含某个单词就置为1，否则置为0

句子1：我/有/一个/苹果
句子2：我/明天/去/一个/地方
句子3：你/到/一个/地方
句子4：我/有/我/最爱的/地方



	我	有	一个	苹果	明天	去	地方	你	到	最爱的
句子1	1	1	1	1	0	0	0	0	0	0
句子2	1	0	1	0	1	1	1	0	0	0
句子3	0	0	1	0	0	0	1	1	1	0
句子4	2	1	0	0	0	0	1	0	0	1



02

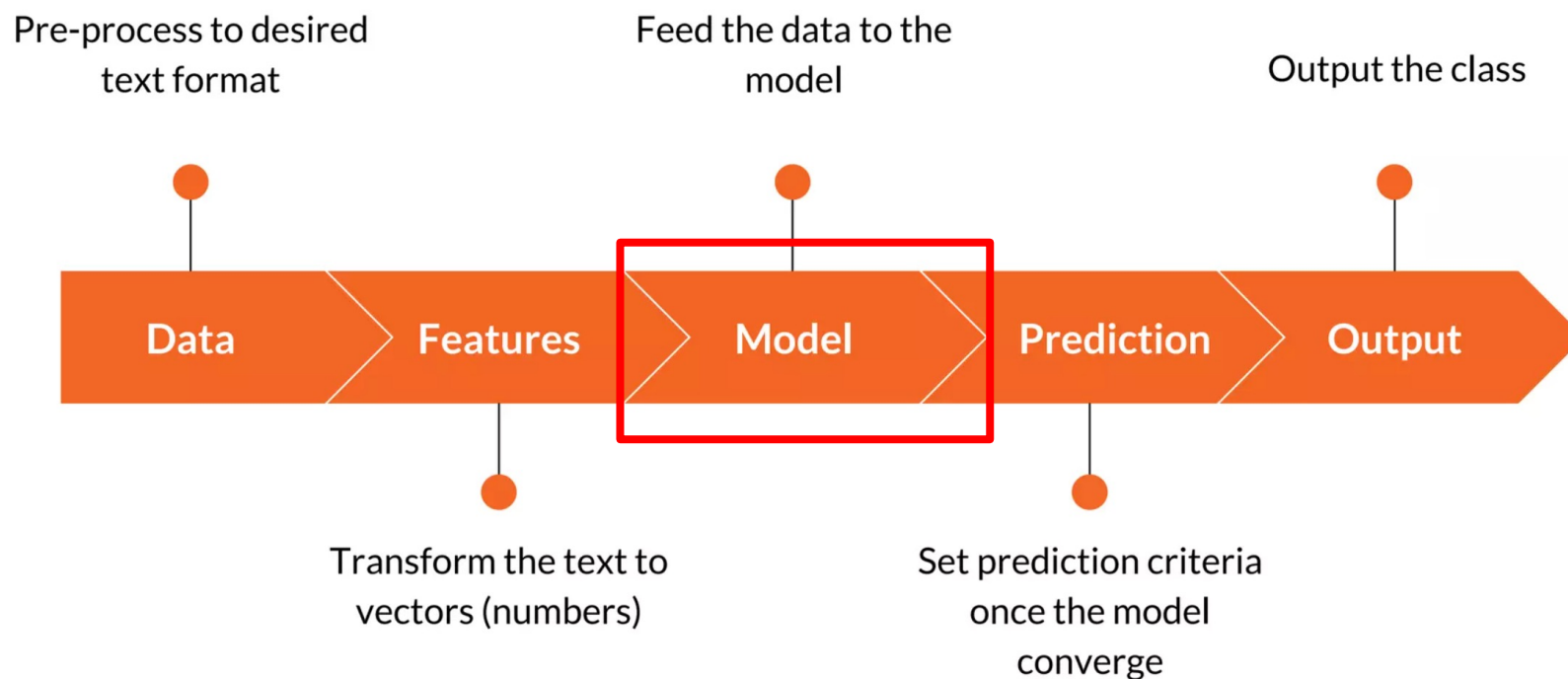


基于统计学习的文本分类

STATISTICAL LEARNING-BASED TEXT CLASSIFICATION

- 主要步骤

- 特征表示 (Feature representation)
- 建模 (Modeling)
- 训练 (Training)
- 推理 (Inference)



- 符号定义

- 文本 $x = \{w_1, w_2, \dots, w_n\}, \quad x \in X$
- 标签 $y \in Y, \quad Y = \{c_1, c_2, \dots, c_m\}$

- 任务目标

$$\operatorname{argmax}_y P(y|x)$$

朴素贝叶斯模型 (NAÏVE BAYES)



- 一个概率模型
- 一个生成式模型
- 具有“朴素”假设
- 适用于离散分布
- 广泛应用于文本分类、自然语言处理和模式识别

朴素贝叶斯模型 (NAÏVE BAYES)



- 贝叶斯公式

$$\begin{aligned}\operatorname{argmax}_y P(y|x) &= \operatorname{argmax}_y \frac{P(x, y)}{P(x)} \\ &= \operatorname{argmax}_y \frac{P(x|y)P(y)}{P(x)} \\ &= \operatorname{argmax}_y P(x|y)P(y)\end{aligned}$$

建模联合概率，生成式模型

$$\operatorname{argmax}_y P(y|x) = \operatorname{argmax}_y \boxed{P(x, y)} = \operatorname{argmax}_y P(x|y)P(y)$$

朴素贝叶斯模型 (NAÏVE BAYES)

- 朴素假设

- 特征之间相互独立，即任意两个词出现的概率互不影响

$$P(x, y) = P(x|y)P(y)$$

$$= P(w_1, w_2, \dots, w_n|y)P(y)$$

朴素假设

$$= \prod_{i=1}^n P(w_i|y)P(y)$$

不准确的形式，帮助理解

如何计算？

和具体的概率文档表示模型有关

伯努利文档模型 (BERNOULLI DOCUMENT MODEL)

- 表示文本时只考虑单词是否出现，不考虑出现次数
- 词表 V 中包含 $|V|$ 个词
- d_t 表示单词 w_t 是否在文本 x 中出现，出现则为 1，不出现则为 0

$$P(x, y) = P(x|y)P(y)$$

$$= P(y) \prod_{t=1}^{|V|} (d_t P(w_t|y) + (1 - d_t)(1 - P(w_t|y)))$$

$$P(x, c_k) = P(c_k) \prod_{t=1}^{|V|} (d_t P(w_t|c_k) + (1 - d_t)(1 - P(w_t|c_k)))$$

- 已标注的数据集 D
 - N : 数据集 D 中的文档总数
 - N_k : 数据集 D 中标签为 c_k 的文档数目
 - $n_k(w_t)$: 标签为 c_k 的文档中, 包含单词 w_t 的文档数目
- 模型参数
 - $P(w_t|c_k)$: 给定类别为 c_k 的条件下, 单词 w_t 出现的概率
 - $P(c_k)$: 类别 c_k 的先验概率

$$P(x, c_k) = P(c_k) \prod_{t=1}^{|V|} (d_t P(w_t|c_k) + (1 - d_t)(1 - P(w_t|c_k)))$$

伯努利文档模型 (BERNOULLI DOCUMENT MODEL)



- 已标注的数据集 D
 - N : 数据集 D 中的文档总数
 - N_k : 数据集 D 中标签为 c_k 的文档数目
 - $n_k(w_t)$: 标签为 c_k 的文档中, 包含单词 w_t 的文档数目
- 模型训练 (参数估计)

$$\hat{P}(w_t|c_k) = \frac{n_k(w_t)}{N_k}$$

$$\hat{P}(c_k) = \frac{N_k}{N}$$

- 推理

- 利用训练好的模型对无标签的文本进行文本分类

$$\begin{aligned}\operatorname{argmax}_{c_k} P(c_k|x) &= \operatorname{argmax}_{c_k} P(x|c_k)P(c_k) \\ &= \operatorname{argmax}_{c_k} P(c_k) \prod_{t=1}^{|V|} (d_t P(w_t|c_k) + (1 - d_t)(1 - P(w_t|c_k)))\end{aligned}$$

伯努利文档模型 (BERNOULLI DOCUMENT MODEL)

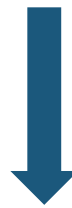
- 零概率

$$\hat{P}(w_t|c_k) = \frac{n_k(w_t)}{N_k}$$



$$\hat{P}(w_t|c_k) = \frac{n_k(w_t) + 1}{N_k + 2}$$

$$\hat{P}(c_k) = \frac{N_k}{N}$$



$$\hat{P}(c_k) = \frac{N_k + 1}{N + |Y|}$$

拉普拉斯平滑

伯努利文档模型 (BERNOULLI DOCUMENT MODEL)



- 遗留问题
 - 表示文本时只考虑单词是否出现，不考虑出现次数
 - 忽略了词频对文本分类的重要性

- 已标注的数据集 D

- N : 数据集 D 中的文档总数

- N_k : 数据集 D 中标签为 c_k 的文档数目

- n_{it} : 单词 w_t 在第 i 个文档中出现的次数

- n_i : 第 i 个文档包含的总单词数

- z_{ik} : 如果第 i 个文档的标签为 c_k , 则 $z_{ik} = 1$; 否则 $z_{ik} = 0$

- 模型参数

- $P(w_t|c_k)$: 给定类别为 c_k 的条件下 , 单词 w_t 出现的概率

- $P(c_k)$: 类别 c_k 的先验概率

$$P(x, c_k) = P(c_k) \frac{n_i!}{\prod_{t=1}^{|V|} n_{it}!} \prod_{t=1}^{|V|} P(w_t|c_k)^{n_{it}}$$

- 已标注的数据集 D
 - N : 数据集 D 中的文档总数
 - N_k : 数据集 D 中标签为 c_k 的文档数目
 - n_{it} : 单词 w_t 在第 i 个文档中出现的次数
 - z_{ik} : 如果第 i 个文档的标签为 c_k , 则 $z_{ik} = 1$; 否则 $z_{ik} = 0$
- 模型训练 (参数估计)

$$\hat{P}(w_t|c_k) = \frac{\sum_{i=1}^N n_{it} z_{ik}}{\sum_{j=1}^{|V|} \sum_{i=1}^N n_{ij} z_{ik}}$$

$$\hat{P}(c_k) = \frac{N_k}{N}$$

- 推理

- 利用训练好的模型对无标签的文本进行文本分类

$$\begin{aligned}\operatorname{argmax}_{c_k} P(c_k|x) &= \operatorname{argmax}_{c_k} P(x|c_k)P(c_k) \\ &= \operatorname{argmax}_{c_k} P(c_k) \frac{n_i!}{\prod_{t=1}^{|V|} n_{it}!} \prod_{t=1}^{|V|} P(w_t|c_k)^{n_{it}} \\ &= \operatorname{argmax}_{c_k} P(c_k) \prod_{t=1}^{|V|} P(w_t|c_k)^{n_{it}}\end{aligned}$$

多项式文档模型 (MULTINOMIAL DOCUMENT MODEL)



- 零概率

$$\hat{P}(w_t|c_k) = \frac{n_k(w_t)}{N_k}$$



$$\hat{P}(w_t|c_k) = \frac{n_k(w_t) + 1}{N_k + |V|}$$

$$\hat{P}(c_k) = \frac{N_k}{N}$$



$$\hat{P}(c_k) = \frac{N_k + 1}{N + |Y|}$$

拉普拉斯平滑

朴素贝叶斯-文本分类

- 训练数据

ID	Text	Label
d_{tr1}	Chinese Beijing Chinese	C
d_{tr2}	Chinese Chinese Shanghai	C
d_{tr3}	Chinese Macao	C
d_{tr4}	Tokyo Japan Chinese	J

- 测试数据

ID	Text
d_{te1}	Chinese Chinese Chinese Tokyo Japan
d_{te2}	Tokyo Tokyo Japan Shanghai

- 类别标签
 - $c1 = C$
 - $c2 = J$
- 特征向量
 - $t1 = \text{Beijing}$
 - $t2 = \text{Chinese}$
 - $t3 = \text{Japan}$
 - $t4 = \text{Macao}$
 - $t5 = \text{Shanghai}$
 - $t6 = \text{Tokyo}$

朴素贝叶斯-文本分类(伯努利文档模型)

- 训练

		Doc	t1	t2	t3	t4	t5	t6
Document Frequency	c1	3	1	3	0	1	1	0
	c2	1	0	1	1	0	0	1
Probability	c1	3/4	2/5	$(3+1)/(3+2)=4/5$	1/5	2/5	2/5	1/5
	c2	1/4	1/3	$(1+1)/(1+2)=2/3$	2/3	1/3	1/3	2/3

- 预测

	Un-normalized	Normalized
$P(c1 d_{te1})$	$(3/4)*(1-2/5)*4/5*1/5*(1-2/5)*(1-2/5)*1/5=0.005184$	0.1911
$P(c2 d_{te1})$	$(1/4)*(1-1/3)*2/3*2/3*(1-1/3)*(1-1/3)*2/3=0.02195$	0.8089
$P(c1 d_{te2})$	$(3/4)*(1-2/5)*(1-3/5)*1/5*(1-2/5)*2/5*1/5=0.001728$	0.2395
$P(c2 d_{te2})$	$(1/4)*(1-1/3)*(1-2/3)*2/3*(1-1/3)*1/3*2/3=0.005487$	0.7605

朴素贝叶斯-文本分类(多项式文档模型)

- 训练

		Doc	t1	t2	t3	t4	t5	t6
Term Frequency	c1	3	1	5	0	1	1	0
	c2	1	0	1	1	0	0	1
Probability	c1	3/4	2/14	$(5+1)/(1+5+1+1+6)=6/14$	1/14	2/14	2/14	1/14
	c2	1/4	1/9	$(1+1)/(1+1+1+6)=2/9$	2/9	1/9	1/9	2/9

- 预测

	Un-normalized	Normalized
$P(c1 d_{te1})$	$(3/4)*(6/14)^3*(1/14)*(1/14)=0.0030121$	0.689757
$P(c2 d_{te1})$	$(1/4)*(2/9)^3*(2/9)*(2/9)=0.0013548$	0.310243
$P(c1 d_{te2})$	$(3/4)*(1/14)^2*(1/14)*(2/14)$	0.113547
$P(c2 d_{te2})$	$(1/4)*(2/9)^2*(2/9)*(1/9)$	0.886453

感知机模型 (PERCEPTRON)



- 1957年，由弗兰克·罗森布拉特发明
- 用于监督学习的分类算法
- 一种线性分类算法
- 为人工神经网络奠定了基础



感知机模型 (PERCEPTRON)

- 模型建模 (二分类)
 - v 为文本 x 的特征表示
 - ω 为特征表示 v 的权重向量
 - \hat{y} 为文本 x 的预测标签

$$\hat{y} = \begin{cases} 1 & \text{if } \omega^T v \geq 0 \\ 0 & \text{if } \omega^T v < 0 \end{cases}$$

- 损失函数

$$\begin{aligned} J &= \sum_{x_i \in C_0} \omega^T v_i - \sum_{x_j \in C_1} \omega^T v_j \\ &= \sum_{i=1}^N ((1 - y_i) \hat{y}_i - y_i (1 - \hat{y}_i)) \omega^T v_i \\ &= \sum_{i=1}^N (\hat{y}_i - y_i) \omega^T v_i \end{aligned}$$

- 参数更新

$$\begin{aligned}\omega &:= \omega + \alpha(y - \hat{y})v \\ &= \begin{cases} \omega + \alpha v, & \text{if } y = 1 \text{ and } \hat{y} = 0 \\ \omega - \alpha v, & \text{if } y = 0 \text{ and } \hat{y} = 1 \\ \omega, & \text{others} \end{cases}\end{aligned}$$

感知机模型 (PERCEPTRON)



- 代码样例

```
threshold = 0.5
learning_rate = 0.1
weights = [0,0,0]
training_set = [((1, 0, 0) , 1), ((1, 0, 1) , 1) , ((1, 1, 0), 1), ((1, 1, 1) , 0)]
def dot_product (values, weights):
    return sum(value * weight for value, weight in zip(values, weights))

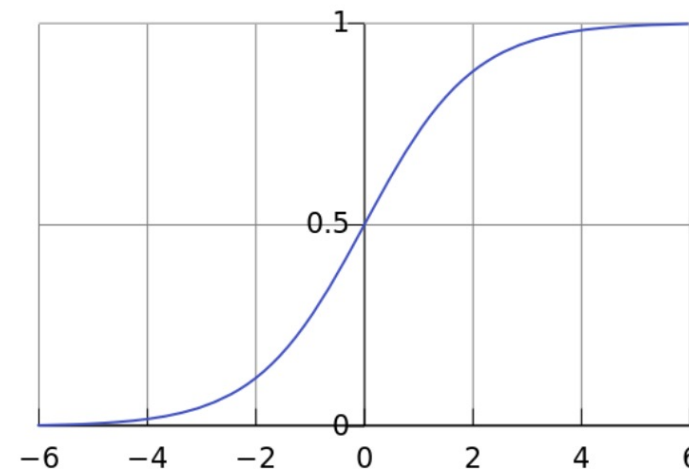
while True:
    print('-'*60)
    error_count = 0
    for input_vector, desired_output in training_set:
        print (weights)
        result = dot_product(input_vector, weights) > threshold
        error = desired_output - result
        if error != 0:
            error_count += 1
            for index, value in enumerate(input_vector):
                weights[index] += learning_rate * error * value
    if error_count == 0:
        break
```

- 逻辑回归是一种二分类模型
- 逻辑回归是一种线性分类模型
- 用一个非线性激活函数（ Sigmoid函数 ）来模拟后验概率

- Sigmoid 函数

$$\delta(z) = \frac{1}{1 + e^{-z}}$$

$$\frac{d\delta(z)}{dz} = \delta(z) (1 - \delta(z))$$



- 模型建模

$$P(y = 1|x; \theta) = \delta(\omega^T v) = \frac{1}{1 + e^{-\omega^T v}}$$

$$P(y = 0|x; \theta) = 1 - P(y = 1|x; \theta)$$

简洁版

$$P(y|x; \theta) = \left(\frac{1}{1 + e^{-\omega^T v}}\right)^y \left(1 - \frac{1}{1 + e^{-\omega^T v}}\right)^{1-y}$$

- 似然函数

$$\begin{aligned} L(\theta) &= \prod_{i=1}^N P(y_i | x_i; \theta) \\ &= \prod_{i=1}^N \left(\frac{1}{1 + e^{-\omega^T v_i}} \right)^{y_i} \left(1 - \frac{1}{1 + e^{-\omega^T v_i}} \right)^{1-y_i} \end{aligned}$$



南京大學
NANJING UNIVERSITY

Thank you !
Q&A

