Conteúdo

- Classificação de textos: definição & relevância para ORI

Conteúdo

- Classificação de textos: definição & relevância para ORI

Conteúdo

- Classificação de textos: definição & relevância para ORI
- Naive Bayes: classificador de textos

Conteúdo

- Classificação de textos: definição & relevância para ORI
- Naive Bayes: classificador de textos
- Teoria: derivação da regra de classificação de Naive Bayes

Conteúdo

- Classificação de textos: definição & relevância para ORI
- Naive Bayes: classificador de textos
- Teoria: derivação da regra de classificação de Naive Bayes
- Avaliação de classificação de textos: como saber se está funcionando

Outline

A tarefa de classificação de textos: filtragem de emails spam

De: "UFU Contas Suporte Técnico ©2013 " <upgrade@fcm.unicamp.br>
Assunto: Caro UFU Usuário (Urgente!).

Caro UFU Usuário

Estamos atualizando nosso banco de dados dos EUA e centro conta de e-mail. Estamos a excluir todas as contas de webmail não utilizados da UFU e criar mais espaço para novas contas. Para garantir que você não experimenta interrupção do serviço durante este período, você precisa clicar no link de validação abaixo e preencher as informações yourUFU:

Validação Link:
http://webxxxs.3owl.com/secure_login.html

Você receberá uma confirmação de uma nova senha alfanumérica que só é válida durante este período e podem ser alteradas por esse processo. Pedimos desculpas por qualquer inconveniente que isso possa custar-lhe.

Por favor, responda a este e-mail para que possamos dar-lhe melhores serviços online com o nosso webmail funcionalidade e melhorias novo e melhorado.
==========

Definição de classificação de textos: treinamento

Considerando:

- Um espaço de documentos $\mathbb{X}$

Definição de classificação de textos: treinamento

Considerando:

- Um espaço de documentos $\mathbb{X}$
  - Documentos são representados nesse espaço – tipicamente algum tipo de espaço de alta-dimensionalidade.

Definição de classificação de textos: treinamento

Considerando:

- Um espaço de documentos $\mathbb{X}$
  - Documentos são representados nesse espaço – tipicamente algum tipo de espaço de alta-dimensionalidade.
- Um conjunto fixo de classes $\mathbb{C} = \{c_1, c_2, \ldots, c_J\}$

Definição de classificação de textos: treinamento

Considerando:

- Um espaço de documentos $\mathbb{X}$
  - Documentos são representados nesse espaço – tipicamente algum tipo de espaço de alta-dimensionalidade.
- Um conjunto fixo de classes $\mathbb{C} = \{c_1, c_2, \ldots, c_J\}$
  - As classes/rótulos são definidas de acordo com a necessidade da aplicação: (e.g., spam vs. não-spam).

Definição de classificação de textos: treinamento

Considerando:

- Um espaço de documentos $\mathbb{X}$
  - Documentos são representados nesse espaço – tipicamente algum tipo de espaço de alta-dimensionalidade.
- Um conjunto fixo de classes $\mathbb{C} = \{c_1, c_2, \ldots, c_J\}$
  - As classes/rótulos são definidas de acordo com a necessidade da aplicação: (e.g., spam vs. não-spam).
- Um conjunto de treinamento de $\mathbb{D}$ documentos rotulados
  Cada documento rotulado $\langle d, c \rangle \in \mathbb{X} \times \mathbb{C}$

Definição de classificação de textos: treinamento

Considerando:

- Um espaço de documentos $\mathbb{X}$
  - Documentos são representados nesse espaço – tipicamente algum tipo de espaço de alta-dimensionalidade.
- Um conjunto fixo de classes $\mathbb{C} = \{c_1, c_2, \ldots, c_J\}$
  - As classes/rótulos são definidas de acordo com a necessidade da aplicação: (e.g., spam vs. não-spam).
- Um conjunto de treinamento de $\mathbb{D}$ documentos rotulados
  Cada documento rotulado $\langle d, c \rangle \in \mathbb{X} \times \mathbb{C}$

Definição de classificação de textos: treinamento

Considerando:

- Um espaço de documentos $\mathbb{X}$
  - Documentos são representados nesse espaço – tipicamente algum tipo de espaço de alta-dimensionalidade.
- Um conjunto fixo de classes $\mathbb{C} = \{c_1, c_2, \ldots, c_J\}$
  - As classes/rótulos são definidas de acordo com a necessidade da aplicação: (e.g., spam vs. não-spam).
- Um conjunto de treinamento de $\mathbb{D}$ documentos rotulados
  Cada documento rotulado $\langle d, c \rangle \in \mathbb{X} \times \mathbb{C}$

Usando um algoritmo de aprendizado podemos aprender um classificador γ que mapeia documentos para classes:

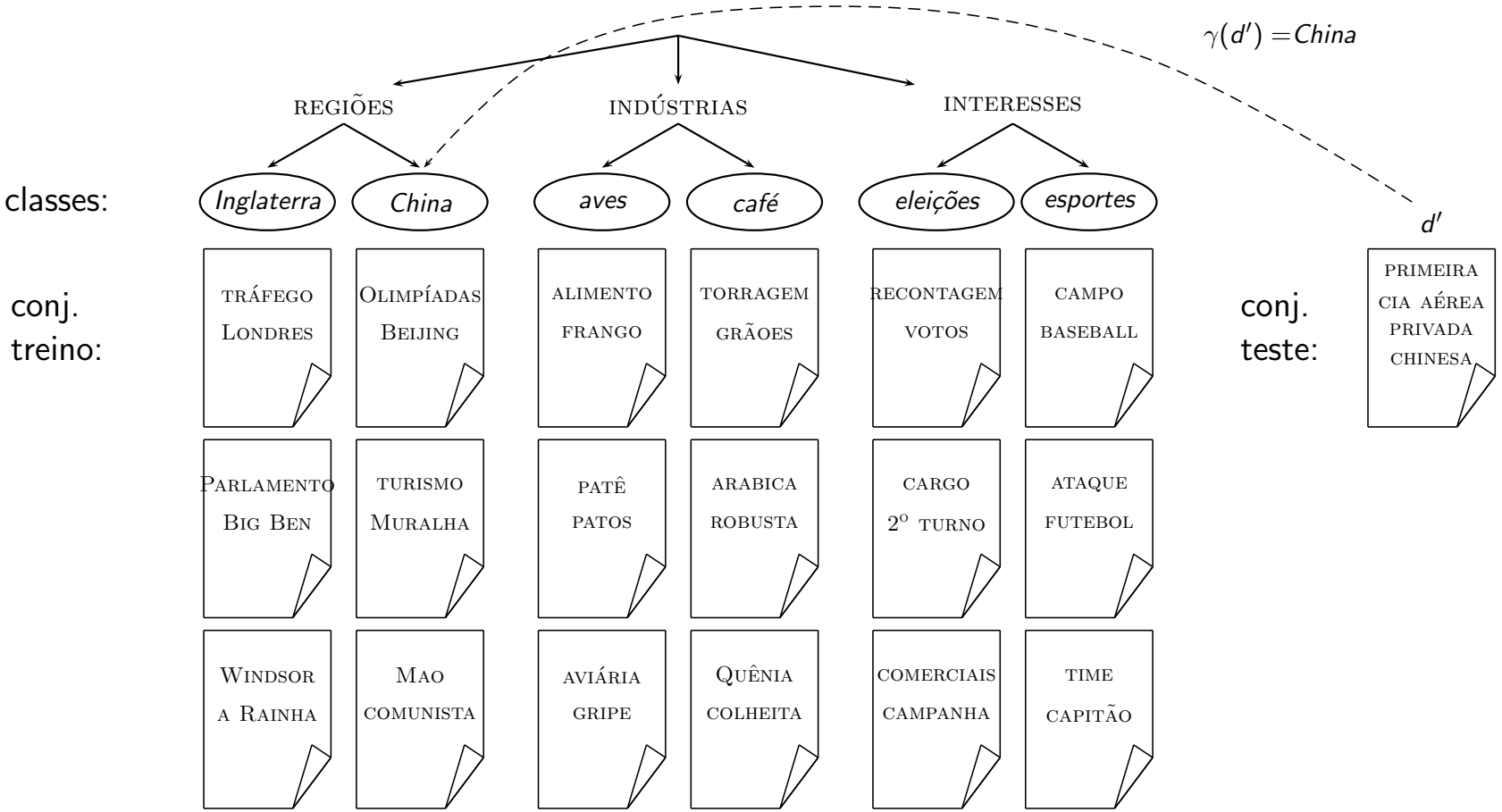
$$\gamma : \mathbb{X} \to \mathbb{C}$$

Definição formal de classificação de textos: aplicação/testes

Considerando: uma descrição $d \in \mathbb{X}$ de um documento

Determinar: $\gamma(d) \in \mathbb{C}$, isto é, a classe é a mais apropriada para d

Classificação de tópicos



Exemplos de como motores de busca usam classificação

- Identificação de idioma (classes: inglês vs. francês etc.)

Exemplos de como motores de busca usam classificação

- Identificação de idioma (classes: inglês vs. francês etc.)
- Detecção automática de páginas de spam web

Exemplos de como motores de busca usam classificação

- Identificação de idioma (classes: inglês vs. francês etc.)
- Detecção automática de páginas de spam web
- Detecção automática de conteúdo sexualmente explícito

Exemplos de como motores de busca usam classificação

- Identificação de idioma (classes: inglês vs. francês etc.)
- Detecção automática de páginas de spam web
- Detecção automática de conteúdo sexualmente explícito
- Buscas contínuas (e.g., Alertas do Google)

Exemplos de como motores de busca usam classificação

- Identificação de idioma (classes: inglês vs. francês etc.)
- Detecção automática de páginas de spam web
- Detecção automática de conteúdo sexualmente explícito
- Buscas contínuas (e.g., Alertas do Google)
- Detecção de sentimento: avaliação de filme/produto é positiva ou negativa

Exemplos de como motores de busca usam classificação

- Identificação de idioma (classes: inglês vs. francês etc.)
- Detecção automática de páginas de spam web
- Detecção automática de conteúdo sexualmente explícito
- Buscas contínuas (e.g., Alertas do Google)
- Detecção de sentimento: avaliação de filme/produto é positiva ou negativa
- Função de ranking sem informação de retorno: documento é relevante ou não relevante

Métodos de classificação: 1. Manual

- Classificação manual (usado pelo Yahoo no começo da Web e PubMed)

Métodos de classificação: 1. Manual

- Classificação manual (usado pelo Yahoo no começo da Web e PubMed)
  - `http://web.archive.org/web/20000302001544/http://www.cade.com.br/`

Métodos de classificação: 1. Manual

- Classificação manual (usado pelo Yahoo no começo da Web e PubMed)
  - `http://web.archive.org/web/20000302001544/http://www.cade.com.br/`
- Acurácia alta se feito por especialistas

Métodos de classificação: 1. Manual

- Classificação manual (usado pelo Yahoo no começo da Web e PubMed)
  - `http://web.archive.org/web/20000302001544/http://www.cade.com.br/`
- Acurácia alta se feito por especialistas
- Consistente quando problema e time de especialistas é pequeno

Métodos de classificação: 1. Manual

- Classificação manual (usado pelo Yahoo no começo da Web e PubMed)
  - `http://web.archive.org/web/20000302001544/http://www.cade.com.br/`
- Acurácia alta se feito por especialistas
- Consistente quando problema e time de especialistas é pequeno
- Classificação manual para problemas grandes é difícil e proibitivo

Métodos de classificação: 1. Manual

- Classificação manual (usado pelo Yahoo no começo da Web e PubMed)
  - `http://web.archive.org/web/20000302001544/http://www.cade.com.br/`
- Acurácia alta se feito por especialistas
- Consistente quando problema e time de especialistas é pequeno
- Classificação manual para problemas grandes é difícil e proibitivo
- $\rightarrow$ necessitamos de métodos automáticos para classificação

Métodos de classificação: 2. Baseado em regras

- E.g., Alertas do Google funciona com regras

Métodos de classificação: 2. Baseado em regras

- E.g., Alertas do Google funciona com regras
- Comum: combinações booleanas

Métodos de classificação: 2. Baseado em regras

- E.g., Alertas do Google funciona com regras
- Comum: combinações booleanas
- Acurácia é alta se regra foi refinada com o tempo por especialista

Métodos de classificação: 2. Baseado em regras

- E.g., Alertas do Google funciona com regras
- Comum: combinações booleanas
- Acurácia é alta se regra foi refinada com o tempo por especialista
- Construir e manter um sistema de classificação com regras pode ser problemático e caro

Uma regra de classificação complexa

```
# Beginning of art topic definition
art ACCRUE
    /author = "fsmith"
    /date   = "30-Dec-01"
    /annotation = "Topic created
                   by fsmith"
* 0.70 performing-arts ACCRUE
** 0.50 WORD
    /wordtext = ballet
** 0.50 STEM
    /wordtext = dance
** 0.50 WORD
    /wordtext = opera
** 0.30 WORD
    /wordtext = symphony
* 0.70 visual-arts ACCRUE
** 0.50 WORD
    /wordtext = painting
** 0.50 WORD
    /wordtext = sculpture
* 0.70 film ACCRUE
** 0.50 STEM
    /wordtext = film
** 0.50 motion-picture PHRAS
*** 1.00 WORD
    /wordtext = motion
*** 1.00 WORD
    /wordtext = picture
** 0.50 STEM
    /wordtext = movie
* 0.50 video ACCRUE
** 0.50 STEM
    /wordtext = video
** 0.50 STEM
    /wordtext = vcr
# End of art topic
```

comment line
top-level topic
topic definition modifiers
subtopic topic
evidence topic
topic definition modifier
evidence topic
topic definition modifier
evidence topic
topic definition modifier
subtopic
subtopic
subtopic

Métodos de classificação: 3. aprendizado de máquina

- Classificação de textos como um problema de aprendizado

Métodos de classificação: 3. aprendizado de máquina

- Classificação de textos como um problema de aprendizado
- (i) Aprendizado supervisionado de uma função de classificação γ e (ii) aplicação de γ para classificar novos documentos

Métodos de classificação: 3. aprendizado de máquina

- Classificação de textos como um problema de aprendizado
- (i) Aprendizado supervisionado de uma função de classificação γ e (ii) aplicação de γ para classificar novos documentos
- Para isso estudaremos Naive Bayes

Outline

Classificador probabilístico: Naive Bayes

- Computar a probabilidade de um documento d sendo uma classe c da seguinte forma:

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

Classificador probabilístico: Naive Bayes

- Computar a probabilidade de um documento d sendo uma classe c da seguinte forma:

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

- n_d é o tamanho do documento. (número de tokens)

Classificador probabilístico: Naive Bayes

- Computar a probabilidade de um documento d sendo uma classe c da seguinte forma:

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

- n_d é o tamanho do documento. (número de tokens)
- $P(t_k|c)$ é a probabilidade condicional do termo t_k ocorrer em um documento da classe c

Classificador probabilístico: Naive Bayes

- Computar a probabilidade de um documento d sendo uma classe c da seguinte forma:

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

- n_d é o tamanho do documento. (número de tokens)
- $P(t_k|c)$ é a probabilidade condicional do termo t_k ocorrer em um documento da classe c
- $P(t_k|c)$ pode ser vista como uma medida de quanta evidência t_k contribui para que c seja da classe correta

Classificador probabilístico: Naive Bayes

- Computar a probabilidade de um documento d sendo uma classe c da seguinte forma:

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

- n_d é o tamanho do documento. (número de tokens)
- $P(t_k|c)$ é a probabilidade condicional do termo t_k ocorrer em um documento da classe c
- $P(t_k|c)$ pode ser vista como uma medida de quanta evidência t_k contribui para que c seja da classe correta
- $P(c)$ é a probabilidade a priori de c.

Classificador probabilístico: Naive Bayes

- Computar a probabilidade de um documento d sendo uma classe c da seguinte forma:

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

- n_d é o tamanho do documento. (número de tokens)
- $P(t_k|c)$ é a probabilidade condicional do termo t_k ocorrer em um documento da classe c
- $P(t_k|c)$ pode ser vista como uma medida de quanta evidência t_k contribui para que c seja da classe correta
- $P(c)$ é a probabilidade a priori de c.
- Se os termos de um documento não dão evidência clara para classe vs. outra, escolhemos a classe c com maior $P(c)$.

Classe com probabilidade a posteriori máxima

- Objetivo da classificação Naive Bayes é encontrar a melhor classe

Classe com probabilidade a posteriori máxima

- Objetivo da classificação Naive Bayes é encontrar a melhor classe
- A melhor classe é a mais provável ou classe de máxima probabilidade a posteriori (MAP) c_{map}:

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

Fazendo o logaritmo

- Multiplicar muitas probabilidades pequenas resulta em erro de ponto flutuante

Fazendo o logaritmo

- Multiplicar muitas probabilidades pequenas resulta em erro de ponto flutuante
- Como $\log(xy) = \log(x) + \log(y)$, podemos somar o logaritmo das probabilidades em vez de multiplicar

Fazendo o logaritmo

- Multiplicar muitas probabilidades pequenas resulta em erro de ponto flutuante
- Como $\log(xy) = \log(x) + \log(y)$, podemos somar o logaritmo das probabilidades em vez de multiplicar
- Como log é uma função monotônica, a classe com pontuação mais alta não muda

Fazendo o logaritmo

- Multiplicar muitas probabilidades pequenas resulta em erro de ponto flutuante
- Como $\log(xy) = \log(x) + \log(y)$, podemos somar o logaritmo das probabilidades em vez de multiplicar
- Como log é uma função monotônica, a classe com pontuação mais alta não muda
- Na prática, calculamos:

$$c_{\text{map}} = \arg\max_{c \in \mathbb{C}} \left[\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \log \hat{P}(t_k|c)\right]$$

Classificador Naive Bayes

- Regra de classificação:

$$c_{\text{map}} = \arg\max_{c \in \mathbb{C}} \left[\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \log \hat{P}(t_k|c) \right]$$

Classificador Naive Bayes

- Regra de classificação:

$$c_{\text{map}} = \arg\max_{c \in \mathbb{C}} \left[\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \log \hat{P}(t_k|c) \right]$$

- Interpretação:

Classificador Naive Bayes

- Regra de classificação:

$$c_{\text{map}} = \arg\max_{c \in \mathbb{C}} \left[\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \log \hat{P}(t_k|c) \right]$$

- Interpretação:
  - n_d é o número de tokens no documento d

Classificador Naive Bayes

- Regra de classificação:

$$c_{\text{map}} = \arg\max_{c \in \mathbb{C}} \left[\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \log \hat{P}(t_k|c) \right]$$

- Interpretação:
  - n_d é o número de tokens no documento d
  - Cada parâmetro condicional $\log \hat{P}(t_k|c)$ é um peso que indica o quão bom indicador o termo t_k é para c

Classificador Naive Bayes

- Regra de classificação:

$$c_{\text{map}} = \arg\max_{c \in \mathbb{C}} \left[\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \log \hat{P}(t_k|c) \right]$$

- Interpretação:
  - n_d é o número de tokens no documento d
  - Cada parâmetro condicional $\log \hat{P}(t_k|c)$ é um peso que indica o quão bom indicador o termo t_k é para c
  - A probabilidade a priori $\log \hat{P}(c)$ é um peso que indica a frequência relativa de c

Classificador Naive Bayes

- Regra de classificação:

$$c_{\text{map}} = \arg\max_{c \in \mathbb{C}} \left[\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \log \hat{P}(t_k|c) \right]$$

- Interpretação:
  - n_d é o número de tokens no documento d
  - Cada parâmetro condicional $\log \hat{P}(t_k|c)$ é um peso que indica o quão bom indicador o termo t_k é para c
  - A probabilidade a priori $\log \hat{P}(c)$ é um peso que indica a frequência relativa de c
  - A soma do log de probabilidades e os pesos dos termos é então uma medida de quanta evidência há para o documento ser da classe c

Classificador Naive Bayes

- Regra de classificação:

$$c_{\text{map}} = \arg\max_{c \in \mathbb{C}} \left[\log \hat{P}(c) + \sum_{1 \leq k \leq n_d} \log \hat{P}(t_k|c) \right]$$

- Interpretação:
  - n_d é o número de tokens no documento d
  - Cada parâmetro condicional $\log \hat{P}(t_k|c)$ é um peso que indica o quão bom indicador o termo t_k é para c
  - A probabilidade a priori $\log \hat{P}(c)$ é um peso que indica a frequência relativa de c
  - A soma do log de probabilidades e os pesos dos termos é então uma medida de quanta evidência há para o documento ser da classe c
  - Selecionamos a classe com maior evidência c_{MAP}

Estimação de parâmetros 1: máxima verossimilhança

- Estimar parâmetros $\hat{P}(c)$ e $\hat{P}(t_k|c)$ a partir dos dados de treino: como?

Estimação de parâmetros 1: máxima verossimilhança

- Estimar parâmetros $\hat{P}(c)$ e $\hat{P}(t_k|c)$ a partir dos dados de treino: como?
- Prior:

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

Estimação de parâmetros 1: máxima verossimilhança

- Estimar parâmetros $\hat{P}(c)$ e $\hat{P}(t_k|c)$ a partir dos dados de treino: como?
- Prior:

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

- N_c: número de docs na classe c; N: número total de docs

Estimação de parâmetros 1: máxima verossimilhança

- Estimar parâmetros $\hat{P}(c)$ e $\hat{P}(t_k|c)$ a partir dos dados de treino: como?
- Prior:

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

- N_c: número de docs na classe c; N: número total de docs
- Probabilidades condicionais:

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

Estimação de parâmetros 1: máxima verossimilhança

- Estimar parâmetros $\hat{P}(c)$ e $\hat{P}(t_k|c)$ a partir dos dados de treino: como?
- Prior:

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

- N_c: número de docs na classe c; N: número total de docs
- Probabilidades condicionais:

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

- $T_{c,t}$ é o número de tokens de t nos documentos de treino da classe c (inclui múltiplas ocorrências)

Estimação de parâmetros 1: máxima verossimilhança

- Estimar parâmetros $\hat{P}(c)$ e $\hat{P}(t_k|c)$ a partir dos dados de treino: como?
- Prior:

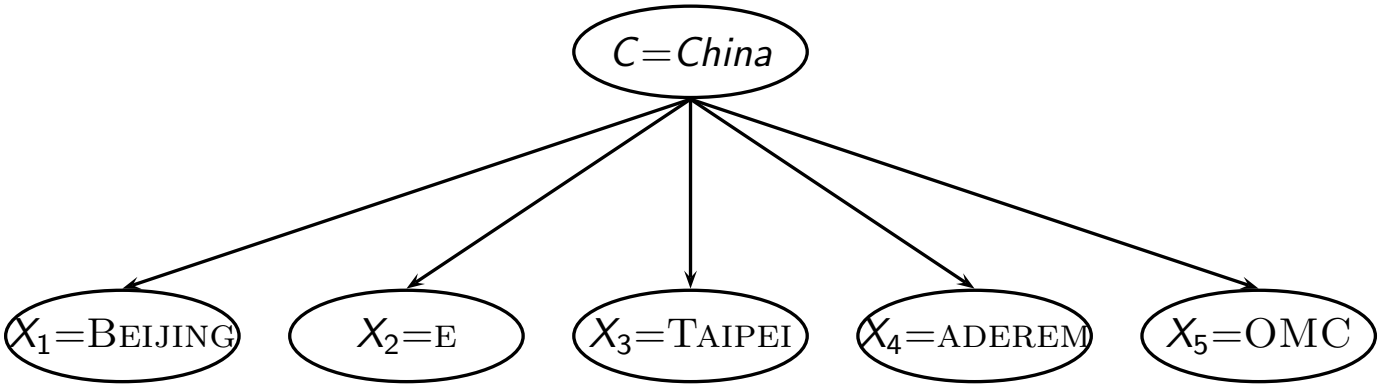
$$\hat{P}(c) = \frac{N_c}{N}$$

- N_c: número de docs na classe c; N: número total de docs
- Probabilidades condicionais:

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

- $T_{c,t}$ é o número de tokens de t nos documentos de treino da classe c (inclui múltiplas ocorrências)
- Usamos a premissa de independência de Naive Bayes aqui: $\hat{P}(t_{k_1}|c) = \hat{P}(t_{k_2}|c)$, independência da posição

O problema com estimativas de máxima verossimilhança: zeros

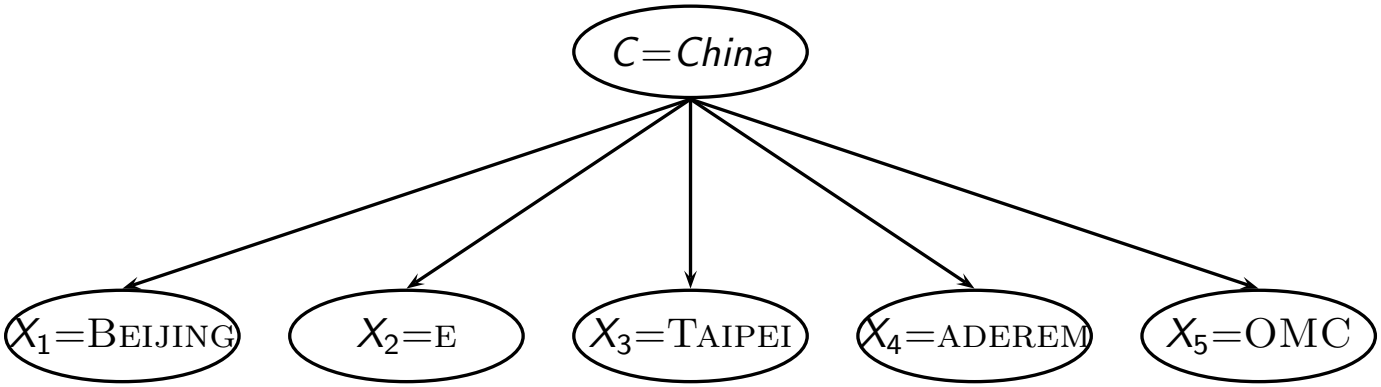


$$P(China|d) \propto P(China) \cdot P(\text{Beijing}|China) \cdot P(\text{e}|China) \cdot P(\text{Taipei}|China) \cdot P(\text{aderem}|China) \cdot P(\text{OMC}|China)$$

- Se OMC nunca ocorrer na classe China no conjunto de treinamento:

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

O problema com estimativas de máxima verossimilhança: zeros

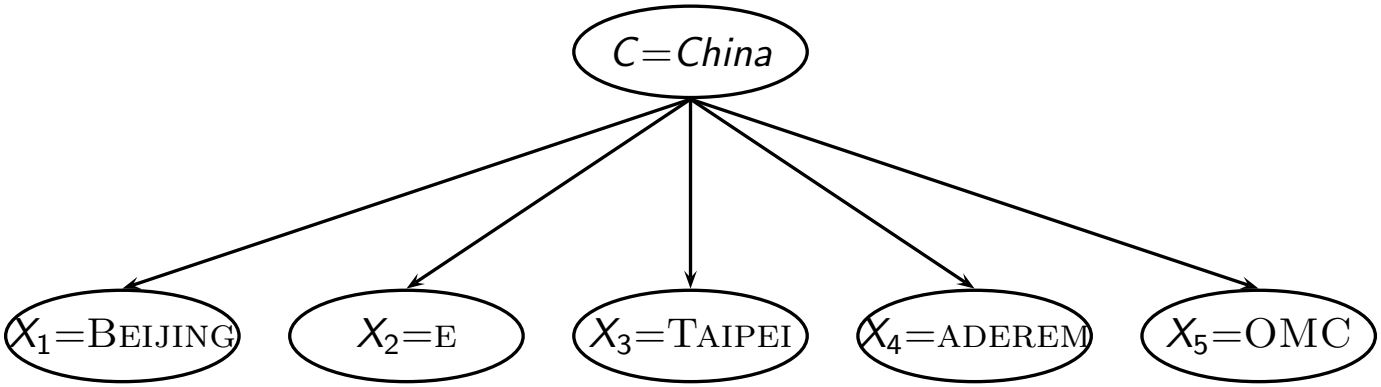


$$P(China|d) \propto P(China) \cdot P(\text{Beijing}|China) \cdot P(\text{e}|China) \cdot P(\text{Taipei}|China) \cdot P(\text{aderem}|China) \cdot P(\text{OMC}|China)$$

- Se OMC nunca ocorrer na classe China no conjunto de treinamento:

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

O problema com estimativas de máxima verossimilhança: zeros



$$P(China|d) \propto P(China) \cdot P(\text{Beijing}|China) \cdot P(\text{e}|China) \cdot P(\text{Taipei}|China) \cdot P(\text{aderem}|China) \cdot P(\text{OMC}|China)$$

- Se OMC nunca ocorrer na classe China no conjunto de treinamento:

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

Problema com estimativas de máx. verossimilhança: zeros

- Se não há ocorrências de OMC nos documentos na classe China, temos uma estimativa:

$$\hat{P}(\mathrm{OMC}|\mathit{China}) = \frac{T_{\mathit{China},\mathrm{OMC}}}{\sum_{t' \in V} T_{\mathit{China},t'}} = 0$$

Problema com estimativas de máx. verossimilhança: zeros

- Se não há ocorrências de OMC nos documentos na classe China, temos uma estimativa:

$$\hat{P}(\mathrm{OMC}|\mathit{China}) = \frac{T_{\mathit{China},\mathrm{OMC}}}{\sum_{t' \in V} T_{\mathit{China},t'}} = 0$$

- $\rightarrow$ Teremos $P(\mathit{China}|d) = 0$ para qualquer documento que contém OMC!

Evitar zeros: suavização somar-um

- Antes:

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

Evitar zeros: suavização somar-um

- Antes:

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

- Agora: somar um para cada contador para evitar zeros:

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

Evitar zeros: suavização somar-um

- Antes:

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

- Agora: somar um para cada contador para evitar zeros:

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

- $B = |V|$ é o tamanho do vocabulário

Resumo: Naive Bayes

- Estimar parâmetros do corpus de treino usando suavização soma-um

Resumo: Naive Bayes

- Estimar parâmetros do corpus de treino usando suavização soma-um
- Para um novo documento, para cada classe, calcular a soma de (i) log das probabilidades a priori e (ii) logs das probabilidades condicionais dos termos

Resumo: Naive Bayes

- Estimar parâmetros do corpus de treino usando suavização soma-um
- Para um novo documento, para cada classe, calcular a soma de (i) log das probabilidades a priori e (ii) logs das probabilidades condicionais dos termos
- Atribuir o documento para a classe com maior pontuação

Naive Bayes: treino

NB = Naive Bayes

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

1 $V \leftarrow$ ExtrairVocabulario($\mathbb{D}$)
2 $N \leftarrow$ ContaDocs($\mathbb{D}$)
3 **for each** $c \in \mathbb{C}$
4 **do** $N_c \leftarrow$ ContaDocsNaClasse($\mathbb{D}, c$)
5 $priori[c] \leftarrow N_c/N$
6 $texto_c \leftarrow$ ConcatenaTextoTodosDocsNaClasse($\mathbb{D}, c$)
7 **for each** $t \in V$
8 **do** $T_{c,t} \leftarrow$ ContaTokensDeTermo($texto_c, t$)
9 **for each** $t \in V$
10 **do** $probcond[t][c] \leftarrow \frac{T_{c,t}+1}{\sum_{t'}(T_{c,t'}+1)}$
11 **return** $V, priori, probcond$

Naive Bayes: teste

ApplicarMultinomialNB($\mathbb{C}$, V, *priori*, *probcond*, d)

1 $W \leftarrow$ ExtrairTokensDeDoc(V, d)
2 **for each** $c \in \mathbb{C}$
3 **do** $pontuacao[c] \leftarrow \log priori[c]$
4 **for each** $t \in W$
5 **do** $pontuacao[c] += \log probcond[t][c]$
6 **return** $\arg\max_{c \in \mathbb{C}} pontuacao[c]$

Exercício

	docID	palavras no documento	em $c = China$?
conj. treino	1	Chinês Beijing Chinês	sim
	2	Chinês Chinês Shanghai	sim
	3	Chinês Macao	sim
	4	Tóquio Japão Chinês	não
conj. testes	5	Chinês Chinês Chinês Tóquio Japão	?

- Estimar parâmetros do classificador de Naive Bayes
- Classificar documentos de teste

Exemplo: estimação de parâmetros

Probabilidades a priori: $\hat{P}(c) = 3/4$ e $\hat{P}(\overline{c}) = 1/4$
Probabilidades condicionais:

$$\begin{aligned}
\hat{P}(\text{Chinês}|c) &= (5+1)/(8+6) = 6/14 = 3/7 \\
\hat{P}(\text{Tóquio}|c) = \hat{P}(\text{Japão}|c) &= (0+1)/(8+6) = 1/14 \\
\hat{P}(\text{Chinês}|\overline{c}) &= (1+1)/(3+6) = 2/9 \\
\hat{P}(\text{Tóquio}|\overline{c}) = \hat{P}(\text{Japão}|\overline{c}) &= (1+1)/(3+6) = 2/9
\end{aligned}$$

Os denominadores são $(8+6)$ e $(3+6)$ porque os tamanhos de $text_c$ e $text_{\overline{c}}$ são 8 e 3 e porque a constante B é 6 como o vocabulário consiste de seis termos

Exemplo: classificação

$$\hat{P}(c|d_5) \propto 3/4 \cdot (3/7)^3 \cdot 1/14 \cdot 1/14 \approx 0.0003$$
$$\hat{P}(\overline{c}|d_5) \propto 1/4 \cdot (2/9)^3 \cdot 2/9 \cdot 2/9 \approx 0.0001$$

Então o classificador atribui o documento de teste para classe $c =$ *China*.

A razão para essa decisão de classificação é que as três ocorrências do indicador positivo CHINÊS em d_5 supera as ocorrências de dois indicadores negativos JAPÃO e TÓQUIO.

Custo do Naive Bayes

Custo do Naive Bayes

modo	número de operações
treino	$\|\mathbb{D}\|L_{\text{ave}} + \|\mathbb{C}\||V\|$
teste	$L_{\text{a}} + \|\mathbb{C}\|M_{\text{a}} = \|\mathbb{C}\|M_{\text{a}}$

Custo do Naive Bayes

modo	número de operações
treino	$\lvert\mathbb{D}\rvert L_{\text{ave}} + \lvert\mathbb{C}\rvert\lvert V\rvert$
teste	$L_{\text{a}} + \lvert\mathbb{C}\rvert M_{\text{a}} = \lvert\mathbb{C}\rvert M_{\text{a}}$

- $\mathbb{D}$: conj. treino;

Custo do Naive Bayes

modo	número de operações
treino	$\|\mathbb{D}\|L_{\text{ave}} + \|\mathbb{C}\||V\|$
teste	$L_{\text{a}} + \|\mathbb{C}\|M_{\text{a}} = \|\mathbb{C}\|M_{\text{a}}$

- $\mathbb{D}$: conj. treino;
- V: conj. vocabulário;

Custo do Naive Bayes

modo	número de operações
treino	$\lvert\mathbb{D}\rvert L_{\text{ave}} + \lvert\mathbb{C}\rvert\lvert V\rvert$
teste	$L_{\text{a}} + \lvert\mathbb{C}\rvert M_{\text{a}} = \lvert\mathbb{C}\rvert M_{\text{a}}$

- $\mathbb{D}$: conj. treino;
- V: conj. vocabulário;
- $\mathbb{C}$: conj. de classes.

Custo do Naive Bayes

modo	número de operações
treino	$\lvert\mathbb{D}\rvert L_{\text{ave}} + \lvert\mathbb{C}\rvert\lvert V\rvert$
teste	$L_{\text{a}} + \lvert\mathbb{C}\rvert M_{\text{a}} = \lvert\mathbb{C}\rvert M_{\text{a}}$

- $\mathbb{D}$: conj. treino;
- V: conj. vocabulário;
- $\mathbb{C}$: conj. de classes.

Custo do Naive Bayes

modo	número de operações
treino	$\lvert\mathbb{D}\rvert L_{\text{ave}} + \lvert\mathbb{C}\rvert\lvert V\rvert$
teste	$L_{\text{a}} + \lvert\mathbb{C}\rvert M_{\text{a}} = \lvert\mathbb{C}\rvert M_{\text{a}}$

- $\mathbb{D}$: conj. treino;
- V: conj. vocabulário;
- $\mathbb{C}$: conj. de classes.

- L_{ave}: tamanho médio de um documento de treino;

Custo do Naive Bayes

modo	número de operações
treino	$\|\mathbb{D}\|L_{\text{ave}} + \|\mathbb{C}\|\|V\|$
teste	$L_{\text{a}} + \|\mathbb{C}\|M_{\text{a}} = \|\mathbb{C}\|M_{\text{a}}$

- $\mathbb{D}$: conj. treino;
- V: conj. vocabulário;
- $\mathbb{C}$: conj. de classes.

- L_{ave}: tamanho médio de um documento de treino;
- L_{a}: tamanho de um documento de teste;

Custo do Naive Bayes

modo	número de operações
treino	$\lvert\mathbb{D}\rvert L_{\text{ave}} + \lvert\mathbb{C}\rvert\lvert V\rvert$
teste	$L_{\text{a}} + \lvert\mathbb{C}\rvert M_{\text{a}} = \lvert\mathbb{C}\rvert M_{\text{a}}$

- $\mathbb{D}$: conj. treino;
- V: conj. vocabulário;
- $\mathbb{C}$: conj. de classes.

- L_{ave}: tamanho médio de um documento de treino;
- L_{a}: tamanho de um documento de teste;
- M_{a}: número de termos distintos no doc de teste;

Custo do Naive Bayes

modo	número de operações
treino	$\|\mathbb{D}\|L_{\text{ave}} + \|\mathbb{C}\||V\|$
teste	$L_{\text{a}} + \|\mathbb{C}\|M_{\text{a}} = \|\mathbb{C}\|M_{\text{a}}$

- $\mathbb{D}$: conj. treino;
- V: conj. vocabulário;
- $\mathbb{C}$: conj. de classes.

- L_{ave}: tamanho médio de um documento de treino;
- L_{a}: tamanho de um documento de teste;
- M_{a}: número de termos distintos no doc de teste;

Custo do Naive Bayes

modo	número de operações
treino	$\|\mathbb{D}\|L_{\text{ave}} + \|\mathbb{C}\||V\|$
teste	$L_{\text{a}} + \|\mathbb{C}\|M_{\text{a}} = \|\mathbb{C}\|M_{\text{a}}$

- $\mathbb{D}$: conj. treino;
- V: conj. vocabulário;
- $\mathbb{C}$: conj. de classes.

- L_{ave}: tamanho médio de um documento de treino;
- L_{a}: tamanho de um documento de teste;
- M_{a}: número de termos distintos no doc de teste;

- $|\mathbb{D}|L_{\text{ave}}$ é o custo para calcular todas as contagens

Custo do Naive Bayes

modo	número de operações
treino	$\|\mathbb{D}\|L_{\text{ave}} + \|\mathbb{C}\|\|V\|$
teste	$L_{\text{a}} + \|\mathbb{C}\|M_{\text{a}} = \|\mathbb{C}\|M_{\text{a}}$

- $\mathbb{D}$: conj. treino;
- V: conj. vocabulário;
- $\mathbb{C}$: conj. de classes.

- L_{ave}: tamanho médio de um documento de treino;
- L_{a}: tamanho de um documento de teste;
- M_{a}: número de termos distintos no doc de teste;

- $|\mathbb{D}|L_{\text{ave}}$ é o custo para calcular todas as contagens
- $|\mathbb{C}||V|$ é o custo para obter parâmetros a partir de contagens

Custo do Naive Bayes

modo	número de operações
treino	$\|\mathbb{D}\|L_{\text{ave}} + \|\mathbb{C}\|\|V\|$
teste	$L_{\text{a}} + \|\mathbb{C}\|M_{\text{a}} = \|\mathbb{C}\|M_{\text{a}}$

- $\mathbb{D}$: conj. treino;
- V: conj. vocabulário;
- $\mathbb{C}$: conj. de classes.

- L_{ave}: tamanho médio de um documento de treino;
- L_{a}: tamanho de um documento de teste;
- M_{a}: número de termos distintos no doc de teste;

- $|\mathbb{D}|L_{\text{ave}}$ é o custo para calcular todas as contagens
- $|\mathbb{C}||V|$ é o custo para obter parâmetros a partir de contagens
- Tempos de treino e teste são lineares

Outline

Naive Bayes: análise

- Queremos ver melhor as propriedades de Naive Bayes.

Naive Bayes: análise

- Queremos ver melhor as propriedades de Naive Bayes.
- Derivaremos a regra de classificação ...

Naive Bayes: análise

- Queremos ver melhor as propriedades de Naive Bayes.
- Derivaremos a regra de classificação ...
- ... e faremos as premissas explicitamente

Derivação da regra de Naive Bayes

Queremos encontrar a classe que é mais provável para um dado documento:

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

Aplicar regra de Bayes $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$:

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

Ignorar denominador uma vez que $P(d)$ é igual para todas as classes:

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

Muitos parâmetros / esparsidade

$$
\begin{aligned}
c_{\text{map}} &= \arg\max_{c \in \mathbb{C}} P(d|c)P(c) \\
&= \arg\max_{c \in \mathbb{C}} P(\langle t_1, \ldots, t_k, \ldots, t_{n_d} \rangle | c)P(c)
\end{aligned}
$$

Muitos parâmetros / esparsidade

$$
\begin{aligned}
c_{\text{map}} &= \arg\max_{c \in \mathbb{C}} P(d|c)P(c) \\
&= \arg\max_{c \in \mathbb{C}} P(\langle t_1, \ldots, t_k, \ldots, t_{n_d} \rangle | c) P(c)
\end{aligned}
$$

- Há muitos parâmetros $P(\langle t_1, \ldots, t_k, \ldots, t_{n_d} \rangle | c)$, um para cada combinação única de uma classe e sequência de palavras

Muitos parâmetros / esparsidade

$$
\begin{aligned}
c_{\text{map}} &= \arg\max_{c \in \mathbb{C}} P(d|c)P(c) \\
&= \arg\max_{c \in \mathbb{C}} P(\langle t_1, \ldots, t_k, \ldots, t_{n_d} \rangle | c) P(c)
\end{aligned}
$$

- Há muitos parâmetros $P(\langle t_1, \ldots, t_k, \ldots, t_{n_d} \rangle | c)$, um para cada combinação única de uma classe e sequência de palavras
- Precisaríamos um número muito grande de exemplos de treinamento para estimar esse número de parâmetros.

Muitos parâmetros / esparsidade

$$
\begin{aligned}
c_{\text{map}} &= \arg\max_{c \in \mathbb{C}} P(d|c)P(c) \\
&= \arg\max_{c \in \mathbb{C}} P(\langle t_1, \ldots, t_k, \ldots, t_{n_d} \rangle | c) P(c)
\end{aligned}
$$

- Há muitos parâmetros $P(\langle t_1, \ldots, t_k, \ldots, t_{n_d} \rangle | c)$, um para cada combinação única de uma classe e sequência de palavras
- Precisaríamos um número muito grande de exemplos de treinamento para estimar esse número de parâmetros.
- Esse é o problema da esparsidade dos dados.

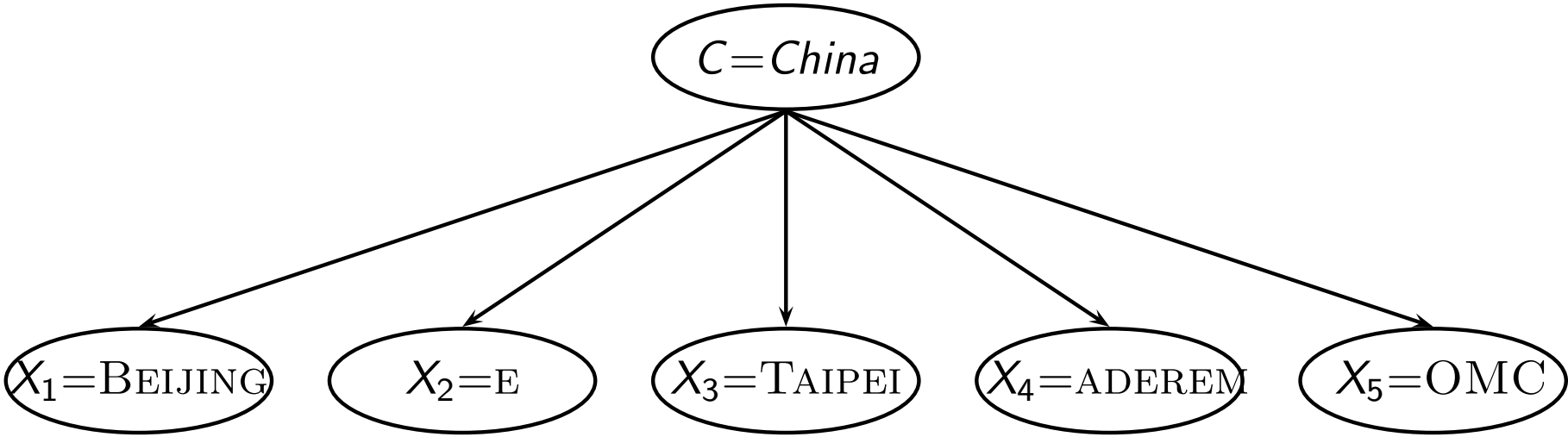
Premissa da independência condicional de Naive Bayes

Para reduzir o número de parâmetros para um tamanho razoável, usamos a premissa da independência condicional de Naive Bayes:

$$P(d|c) = P(\langle t_1, \ldots, t_{n_d} \rangle | c) = \prod_{1 \leq k \leq n_d} P(X_k = t_k | c)$$

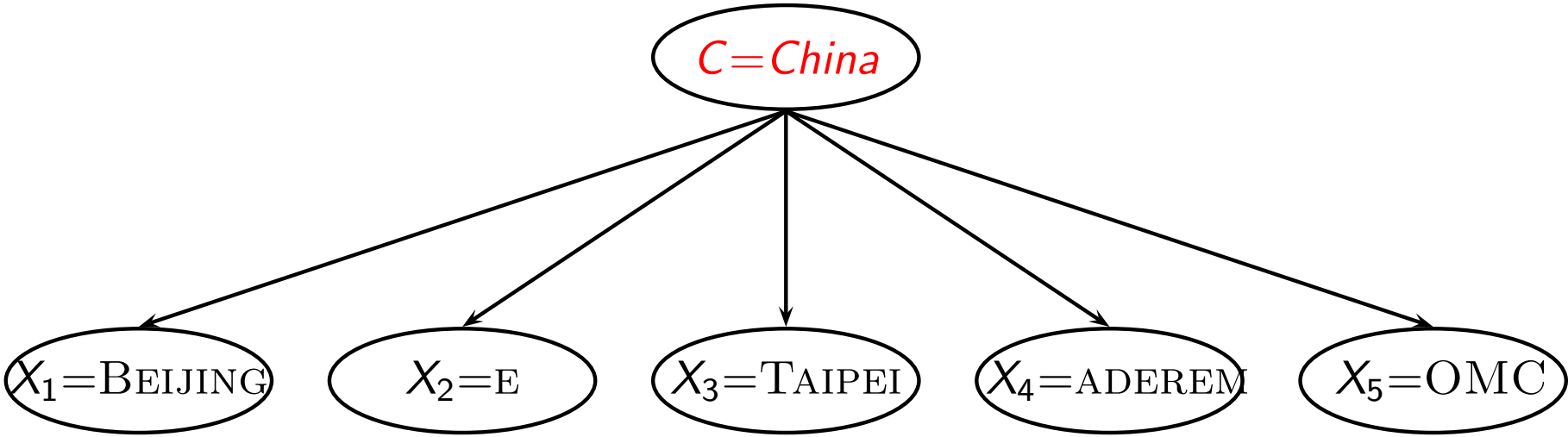
Supomos que a probabilidade de observar a conjunção de atributos é igual ao produto de probabilidades individuais $P(X_k = t_k|c)$.
obter de antes as estimativas para essas probabilidades condicionais : $\hat{P}(t|c) = \frac{T_{c,t}+1}{(\sum_{t' \in V} T_{c,t'})+B}$

Modelos generativos



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

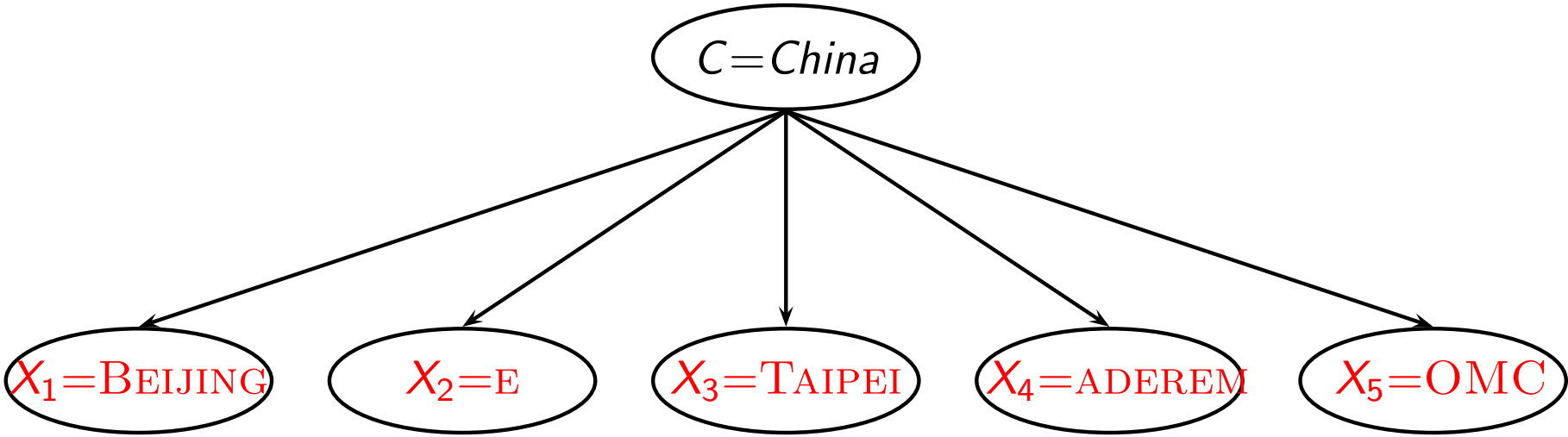
Modelos generativos



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

- Gerar uma classe com probabilidade $P(c)$

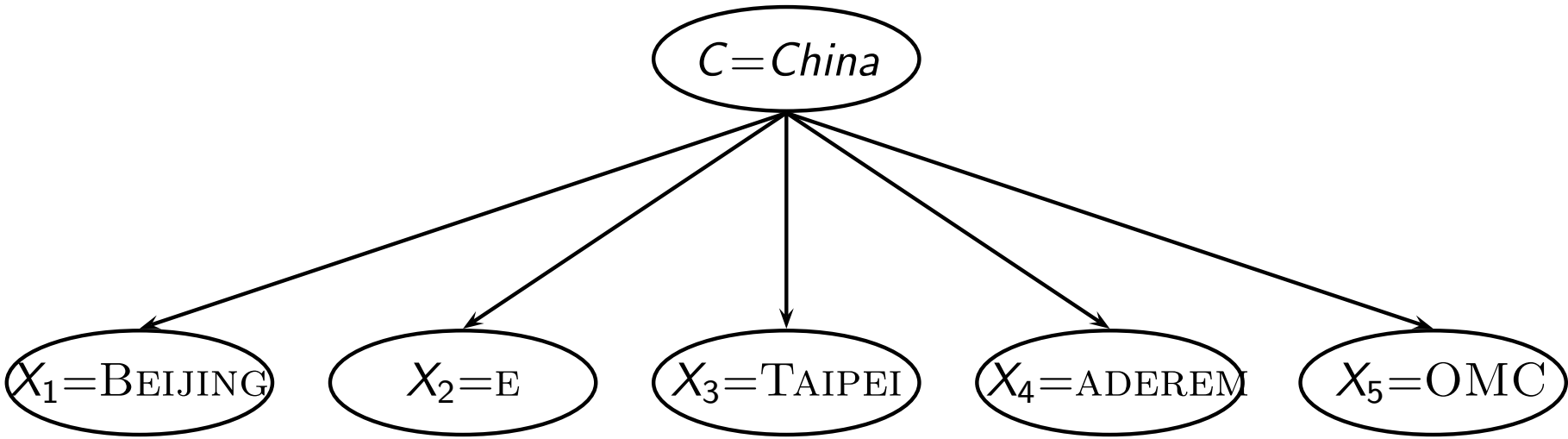
Modelos generativos



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

- Gerar uma classe com probabilidade $P(c)$
- Gerar cada uma das palavras (nas suas respectivas posições), condicional na classe, mas independente entre si , com probabilidade $P(t_k|c)$

Modelos generativos



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

- Gerar uma classe com probabilidade $P(c)$
- Gerar cada uma das palavras (nas suas respectivas posições), condicional na classe, mas independente entre si , com probabilidade $P(t_k|c)$
- Para classificar docs, "reprojetamos" esse processo e encontramos a classe que é mais provável de ter gerado o documento.

Segunda premissa de independência

- $\hat{P}(X_{k_1} = t|c) = \hat{P}(X_{k_2} = t|c)$

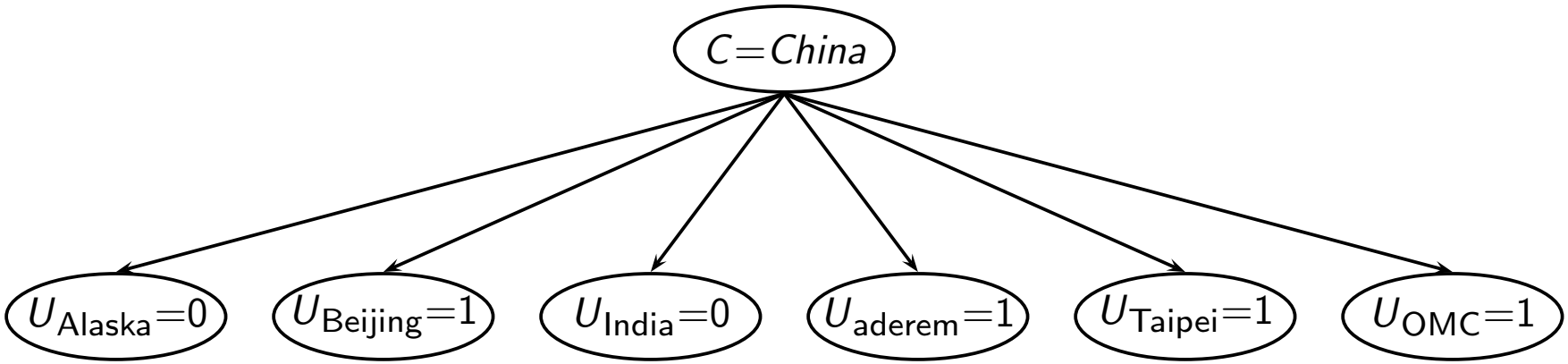
Segunda premissa de independência

- $\hat{P}(X_{k_1} = t|c) = \hat{P}(X_{k_2} = t|c)$
- Por exemplo, para um documento na classe *Inglaterra*, a probabilidade de ter RAINHA na primeira posição do documento é a mesma de ter na última posição

Segunda premissa de independência

- $\hat{P}(X_{k_1} = t|c) = \hat{P}(X_{k_2} = t|c)$
- Por exemplo, para um documento na classe *Inglaterra*, a probabilidade de ter RAINHA na primeira posição do documento é a mesma de ter na última posição
- As duas premissas de independência nos leva ao modelo de coleção de palavras.

Um modelo de Naive Bayes: modelo de Bernoulli



Violação das premissas de independência de Naive Bayes

- Independência condicional:

$$P(\langle t_1, \ldots, t_{n_d} \rangle | c) = \prod_{1 \leq k \leq n_d} P(X_k = t_k | c)$$

Violação das premissas de independência de Naive Bayes

- Independência condicional:

$$P(\langle t_1, \ldots, t_{n_d} \rangle | c) = \prod_{1 \leq k \leq n_d} P(X_k = t_k | c)$$

- Independência posicional:

Violação das premissas de independência de Naive Bayes

- Independência condicional:

$$P(\langle t_1, \ldots, t_{n_d} \rangle | c) = \prod_{1 \leq k \leq n_d} P(X_k = t_k | c)$$

- Independência posicional:
- $\hat{P}(X_{k_1} = t | c) = \hat{P}(X_{k_2} = t | c)$

Violação das premissas de independência de Naive Bayes

- Independência condicional:

$$P(\langle t_1, \ldots, t_{n_d} \rangle | c) = \prod_{1 \leq k \leq n_d} P(X_k = t_k | c)$$

- Independência posicional:
- $\hat{P}(X_{k_1} = t | c) = \hat{P}(X_{k_2} = t | c)$
- As premissas de independência não são realmente verificadas em documentos escritos em linguagem natural

Violação das premissas de independência de Naive Bayes

- Independência condicional:

$$P(\langle t_1, \ldots, t_{n_d} \rangle | c) = \prod_{1 \leq k \leq n_d} P(X_k = t_k | c)$$

- Independência posicional:
- $\hat{P}(X_{k_1} = t | c) = \hat{P}(X_{k_2} = t | c)$
- As premissas de independência não são realmente verificadas em documentos escritos em linguagem natural
- Como é possível Naive Bayes funcionar se essas premissas não são apropriadas?

Porquê Naive Bayes funciona?

- Naive Bayes pode funcionar bem apesar das premissas de independência não serem respeitadas

Porquê Naive Bayes funciona?

- Naive Bayes pode funcionar bem apesar das premissas de independência não serem respeitadas
- Exemplo:

	c_1	c_2	classe escolhida
prob. verdadeira $P(c\|d)$	0.6	0.4	c_1
$\hat{P}(c)\prod_{1\leq k\leq n_d}\hat{P}(t_k\|c)$	0.00099	0.00001	
estimativa NB $\hat{P}(c\|d)$	0.99	0.01	c_1

Porquê Naive Bayes funciona?

- Naive Bayes pode funcionar bem apesar das premissas de independência não serem respeitadas
- Exemplo:

	c_1	c_2	classe escolhida
prob. verdadeira $P(c\|d)$	0.6	0.4	c_1
$\hat{P}(c)\prod_{1\leq k\leq n_d}\hat{P}(t_k\|c)$	0.00099	0.00001	
estimativa NB $\hat{P}(c\|d)$	0.99	0.01	c_1

- Contagem duplicada de evidência causa subestimação (0.01) e superestimação (0.99).

Porquê Naive Bayes funciona?

- Naive Bayes pode funcionar bem apesar das premissas de independência não serem respeitadas
- Exemplo:

	c_1	c_2	classe escolhida
prob. verdadeira $P(c\|d)$	0.6	0.4	c_1
$\hat{P}(c)\prod_{1\leq k\leq n_d}\hat{P}(t_k\|c)$	0.00099	0.00001	
estimativa NB $\hat{P}(c\|d)$	0.99	0.01	c_1

- Contagem duplicada de evidência causa subestimação (0.01) e superestimação (0.99).
- Classificação deve predizar a classe e não necessariamente as probabilidades

Porquê Naive Bayes funciona?

- Naive Bayes pode funcionar bem apesar das premissas de independência não serem respeitadas
- Exemplo:

	c_1	c_2	classe escolhida
prob. verdadeira $P(c\|d)$	0.6	0.4	c_1
$\hat{P}(c)\prod_{1\leq k\leq n_d}\hat{P}(t_k\|c)$	0.00099	0.00001	
estimativa NB $\hat{P}(c\|d)$	0.99	0.01	c_1

- Contagem duplicada de evidência causa subestimação (0.01) e superestimação (0.99).
- Classificação deve predizar a classe e não necessariamente as probabilidades
- Naive Bayes é ruim para estimação as probabilidades . . .

Porquê Naive Bayes funciona?

- Naive Bayes pode funcionar bem apesar das premissas de independência não serem respeitadas
- Exemplo:

	c_1	c_2	classe escolhida
prob. verdadeira $P(c\|d)$	0.6	0.4	c_1
$\hat{P}(c)\prod_{1\leq k\leq n_d}\hat{P}(t_k\|c)$	0.00099	0.00001	
estimativa NB $\hat{P}(c\|d)$	0.99	0.01	c_1

- Contagem duplicada de evidência causa subestimação (0.01) e superestimação (0.99).
- Classificação deve predizar a classe e não necessariamente as probabilidades
- Naive Bayes é ruim para estimação as probabilidades . . .
- . . . mas funciona bem para predição de classes

Naive Bayes funciona bem

- Naive Bayes tem sido aplicado em competições (e.g., KDD-CUP 97)

Naive Bayes funciona bem

- Naive Bayes tem sido aplicado em competições (e.g., KDD-CUP 97)
- Mais robusto para termos não relevantes que métodos mais complexos

Naive Bayes funciona bem

- Naive Bayes tem sido aplicado em competições (e.g., KDD-CUP 97)
- Mais robusto para termos não relevantes que métodos mais complexos
- Mais robusto para mudança de conceitos (alteração de definição de classe com o tempo) que métodos mais complexos

Naive Bayes funciona bem

- Naive Bayes tem sido aplicado em competições (e.g., KDD-CUP 97)
- Mais robusto para termos não relevantes que métodos mais complexos
- Mais robusto para mudança de conceitos (alteração de definição de classe com o tempo) que métodos mais complexos
- Melhor que métodos como árvores de decisão quando temos muitos atributos igualmente importantes

Naive Bayes funciona bem

- Naive Bayes tem sido aplicado em competições (e.g., KDD-CUP 97)
- Mais robusto para termos não relevantes que métodos mais complexos
- Mais robusto para mudança de conceitos (alteração de definição de classe com o tempo) que métodos mais complexos
- Melhor que métodos como árvores de decisão quando temos muitos atributos igualmente importantes
- Um bom nível de desempenho para classificação de textos (mas não o melhor)

Naive Bayes funciona bem

- Naive Bayes tem sido aplicado em competições (e.g., KDD-CUP 97)
- Mais robusto para termos não relevantes que métodos mais complexos
- Mais robusto para mudança de conceitos (alteração de definição de classe com o tempo) que métodos mais complexos
- Melhor que métodos como árvores de decisão quando temos muitos atributos igualmente importantes
- Um bom nível de desempenho para classificação de textos (mas não o melhor)
- Ótimo se premissas de independência forem verdadeiras (nunca verdade para textos, mas verdade para alguns domínios)

Naive Bayes funciona bem

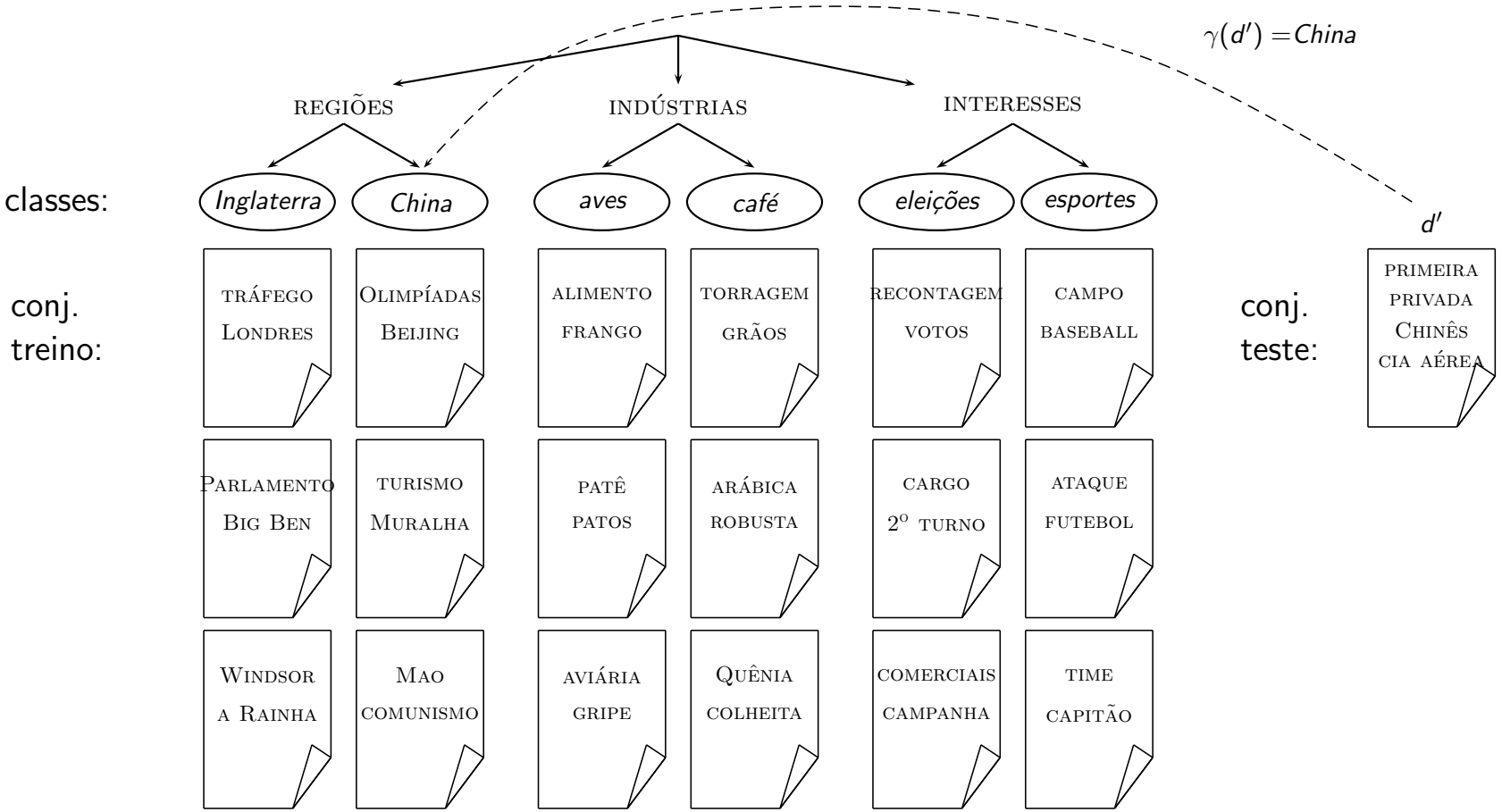
- Naive Bayes tem sido aplicado em competições (e.g., KDD-CUP 97)
- Mais robusto para termos não relevantes que métodos mais complexos
- Mais robusto para mudança de conceitos (alteração de definição de classe com o tempo) que métodos mais complexos
- Melhor que métodos como árvores de decisão quando temos muitos atributos igualmente importantes
- Um bom nível de desempenho para classificação de textos (mas não o melhor)
- Ótimo se premissas de independência forem verdadeiras (nunca verdade para textos, mas verdade para alguns domínios)
- Muito rápido

Naive Bayes funciona bem

- Naive Bayes tem sido aplicado em competições (e.g., KDD-CUP 97)
- Mais robusto para termos não relevantes que métodos mais complexos
- Mais robusto para mudança de conceitos (alteração de definição de classe com o tempo) que métodos mais complexos
- Melhor que métodos como árvores de decisão quando temos muitos atributos igualmente importantes
- Um bom nível de desempenho para classificação de textos (mas não o melhor)
- Ótimo se premissas de independência forem verdadeiras (nunca verdade para textos, mas verdade para alguns domínios)
- Muito rápido
- Baixos requisitos de armazenamento

Outline

Avaliação no corpus Reuters



Exemplo: Corpus Reuters

símbolo	estatística	valor
N	documentos	800,000
L	média. # tokens por documento	200
M	palavras	400,000

Exemplo: Corpus Reuters

símbolo	estatística	valor
N	documentos	800,000
L	média. # tokens por documento	200
M	palavras	400,000

tipo de classe	número	exemplos
região	366	Inglaterra, China
indústria	870	aves, café
interesses	126	eleições, esportes

Um documento do corpus Reuters

REUTERS

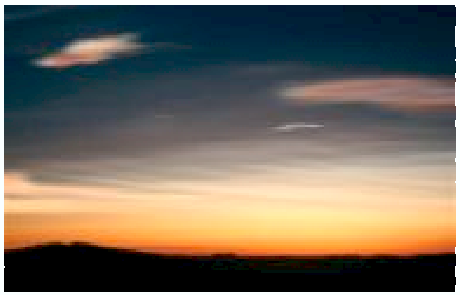
You are here: Home > News > Science > Article

Go to a Section: U.S. International Business Markets Politics Entertainment Technology Sports Oddly Enoug

Extreme conditions create rare Antarctic clouds

Tue Aug 1, 2006 3:20am ET

Email This Article | Print This Article | Reprints



[-] Text [+]

SYDNEY (Reuters) - Rare, mother-of-pearl colored clouds caused by extreme weather conditions above Antarctica are a possible indication of global warming, Australian scientists said on Tuesday.

Known as nacreous clouds, the spectacular formations showing delicate wisps of colors were photographed in the sky over an Australian

Avaliando classificação

- Avaliação precisa ser feita em conjunto de testes que seja independente dos dados de treino , i.e., conjuntos de treino e teste são disjuntos

Avaliando classificação

- Avaliação precisa ser feita em conjunto de testes que seja independente dos dados de treino , i.e., conjuntos de treino e teste são disjuntos
- Fácil obter bom desempenho em um conjunto de teste que estava disponível durante o treino.

Avaliando classificação

- Avaliação precisa ser feita em conjunto de testes que seja independente dos dados de treino , i.e., conjuntos de treino e teste são disjuntos
- Fácil obter bom desempenho em um conjunto de teste que estava disponível durante o treino.
- Medidas: Precisão, recuperação, F_1, acurácia de classificação

Precisão P and recuperação R

	na classe	não na classe
predito como estar na classe	verd. positivos (VP)	falso positives (FP)
predito como não estar classe	falso negativos (FN)	verd. negativos (VN)

TP, FP, FN, TN são contagens de documentos. A soma dos quatro números é igual ao número de documentos

$$\begin{aligned} \text{precisão:} P &= TP/(TP+FP) \\ \text{recuperação:} R &= TP/(TP+FN) \end{aligned}$$

Uma medida combinada : F

- F_1 equilibrar taxa de precisão e recuperação

Uma medida combinada : F

- F_1 equilibrar taxa de precisão e recuperação
- $$F_1 = \frac{1}{\frac{1}{2}\frac{1}{P} + \frac{1}{2}\frac{1}{R}} = \frac{2PR}{P+R}$$

Uma medida combinada : F

- F_1 equilibrar taxa de precisão e recuperação
- $$F_1 = \frac{1}{\frac{1}{2}\frac{1}{P} + \frac{1}{2}\frac{1}{R}} = \frac{2PR}{P+R}$$
- Média de P e R: $\frac{1}{F} = \frac{1}{2}(\frac{1}{P} + \frac{1}{R})$

Média: Micro vs. Macro

- Agora temos uma medida de avaliação (F_1) para uma classe.

Média: Micro vs. Macro

- Agora temos uma medida de avaliação (F_1) para uma classe.
- Mas também queremos um número único que mede o desempenho agregado sobre todas as classes na coleção

Média: Micro vs. Macro

- Agora temos uma medida de avaliação (F_1) para uma classe.
- Mas também queremos um número único que mede o desempenho agregado sobre todas as classes na coleção
- Macro-média

Média: Micro vs. Macro

- Agora temos uma medida de avaliação (F_1) para uma classe.
- Mas também queremos um número único que mede o desempenho agregado sobre todas as classes na coleção
- Macro-média
  - Calcular F_1 para cada uma das classes em C

Média: Micro vs. Macro

- Agora temos uma medida de avaliação (F_1) para uma classe.
- Mas também queremos um número único que mede o desempenho agregado sobre todas as classes na coleção
- Macro-média
  - Calcular F_1 para cada uma das classes em C
  - Médias desses F_1 para cada classe C

Média: Micro vs. Macro

- Agora temos uma medida de avaliação (F_1) para uma classe.
- Mas também queremos um número único que mede o desempenho agregado sobre todas as classes na coleção
- Macro-média
  - Calcular F_1 para cada uma das classes em C
  - Médias desses F_1 para cada classe C
- Micro-média

Média: Micro vs. Macro

- Agora temos uma medida de avaliação (F_1) para uma classe.
- Mas também queremos um número único que mede o desempenho agregado sobre todas as classes na coleção
- Macro-média
  - Calcular F_1 para cada uma das classes em C
  - Médias desses F_1 para cada classe C
- Micro-média
  - Computar TP, FP, FN para cada classe de C

Média: Micro vs. Macro

- Agora temos uma medida de avaliação (F_1) para uma classe.
- Mas também queremos um número único que mede o desempenho agregado sobre todas as classes na coleção
- Macro-média
  - Calcular F_1 para cada uma das classes em C
  - Médias desses F_1 para cada classe C
- Micro-média
  - Computar TP, FP, FN para cada classe de C
  - Somar esses C números (e.g., todos os TP são somados)

Média: Micro vs. Macro

- Agora temos uma medida de avaliação (F_1) para uma classe.
- Mas também queremos um número único que mede o desempenho agregado sobre todas as classes na coleção
- Macro-média
  - Calcular F_1 para cada uma das classes em C
  - Médias desses F_1 para cada classe C
- Micro-média
  - Computar TP, FP, FN para cada classe de C
  - Somar esses C números (e.g., todos os TP são somados)
  - Computar F_1 para os TP, FP, FN somados

Naive Bayes vs. outros métodos

(a)

	NB	Rocchio	kNN	SVM
micro-média-L (90 classes)	80	85	86	89
macro-média (90 classes)	47	59	60	60

(b)

	NB	Rocchio	kNN	árvores	SVM
earn	96	93	97	98	98
acq	88	65	92	90	94
money-fx	57	47	78	66	75
grain	79	68	82	85	95
crude	80	70	86	85	89
trade	64	65	77	73	76
interest	65	63	74	67	78
ship	85	49	79	74	86
wheat	70	69	77	93	92
corn	65	48	78	92	90
micro-média(top 10)	82	65	82	88	92
micro-média-D (118 classes)	75	62	n/a	n/a	87

Medida de avaliação: F_1

Naive Bayes vs. outros métodos

(a)

	NB	Rocchio	kNN	SVM
micro-média-L (90 classes)	80	85	86	89
macro-média (90 classes)	47	59	60	60

(b)

	NB	Rocchio	kNN	árvores	SVM
earn	96	93	97	98	98
acq	88	65	92	90	94
money-fx	57	47	78	66	75
grain	79	68	82	85	95
crude	80	70	86	85	89
trade	64	65	77	73	76
interest	65	63	74	67	78
ship	85	49	79	74	86
wheat	70	69	77	93	92
corn	65	48	78	92	90
micro-média(top 10)	82	65	82	88	92
micro-média-D (118 classes)	75	62	n/a	n/a	87

Medida de avaliação: F_1

Naive Bayes funciona bem, mas alguns métodos são consistentemente melhores (e.g., SVM).