MAC 5855 - Desenvolvimento Web

Filtragem de Documentos

Israel Danilo Lacerra
Thiago Henrique Coraini

Introdução

- Filtragem de documentos é o processo de classificar, automaticamente, diversos documentos em diferentes categorias.
- Aplicação mais comum: *anti-spam*
- Com a Web cada vez mais interativa, filtrar o conteúdo submetido pelos usuários torna-se um problema importante. Ex:
  - Wikis
  - Fóruns
  - Comentários em geral

Filtragem

- O algoritmo deve aprender, baseado em dicas do usuário, a que categoria um determinado documento pertence.
- Se o usuário "diz" ao algoritmo que um email é sobre esportes, o algoritmo "aprende" que emails parecidos com esse serão sobre esportes.
- Elementos centrais: *Features* e Documentos.

Filtragem: *Features*

- Uma ***Feature*** é algo que pode ter a ocorrência testada num documento. Em geral, *features* representam palavras.

- A idéia é que uma determinada *feature* pode ser mais comum em emails de uma determinada categoria.

  Por exemplo: *casino* ou *Viagra* para emails de *spam* .

Filtragem: *Features*

Importante: *features* **não** precisam ser palavras. Podem ser qualquer elemento que possa ser testado como presente ou não num documento. Ex:

- Conjuntos de palavras
- Porcentagem específica de palavras em letras maiúsculas
- Informações específicas, como o remetente de um email por exemplo.

Treinamento

- Numa primeira fase, o classificador precisa ser "treinado", para aprender quais *features* caracterizam documentos de cada categoria.

- Para isso, ele armazena, para cada *feature* , quantas vezes ela foi classificada em cada categoria.

```
{'casino': {'good': 1, 'spam': 7}, 'java':
          {'good': 5, 'spam': 1}}
```

- O treinamento acontece passando diversos documentos ao algoritmo e suas categorias.

Probabilidades

O algoritmo trabalha com cálculos de probabilidade. A primeira que ele calcula é:

$$\Pr(\textit{feature} \mid \text{categoria}) = \frac{\textit{nº de ocorrências da feature na categoria}}{\textit{nº total de documentos na categoria}}$$

No começo do treinamento, o algoritmo pode estar muito "sensível" quanto a *features* que apareceram pouco. Solução: todas as *features* começam com uma *probabilidade assumida* .

Classificador *Bayseano*

- Já mostramos como obter Pr (*feature* | categoria). Como, a partir daí, conseguir Pr (documento | categoria)?
- Assumindo a probabilidade de cada *feature* do documento como independente, basta multiplicá-las.

- Mas, será que essa suposição é válida?

Classificador *Bayseano*

- Já mostramos como obter Pr (*feature* | categoria). Como, a partir daí, conseguir Pr (documento | categoria)?
- Assumindo a probabilidade de cada *feature* do documento como independente, basta multiplicá-las.

- Mas, será que essa suposição é válida? **NÃO!**

  Porém, as probablidades obtidas, mesmo não sendo realísticas, podem ser **comparadas**.

Classificador *Bayseano*

Temos agora $\Pr(\text{documento} \mid \text{categoria})$. Mas, o que realmente desejamos é $\Pr(\text{categoria} \mid \text{documento})$.

Como conseguir isso? **Teorema de *Bayes*** .

$$\Pr(A|B) = \Pr(B|A) \times \Pr(A)/\Pr(B)$$

Ou seja:

$$\Pr(\text{categoria}|\text{documento}) = \Pr(\text{documento}|\text{categoria}) \times \frac{\Pr(\text{categoria})}{\Pr(\text{documento})}$$

Classificador *Bayseano*

- Queremos comparar as probabilidades para apenas um documento.
- Pr(documento) não influi na nossa comparação.

Então:

$$Pr(categoria|documento) = Pr(documento|categoria) \times \frac{Pr(categoria)}{Pr(documento)}$$

Pode ser simplificado:

$$Pr\,(categoria \mid documento) = Pr\,(documento \mid categoria) \times Pr\,(categoria)$$

Classificador *Bayseano*

- A classificação de um documento fica então muito simples: calcula-se a probabilidade dele estar em cada categoria, e decide-se pela de maior probabilidade.

- Porém, deve-se tomar **cuidado**! Mais vale um *spam* na caixa de entrada do que um email importante perdido da caixa de *spam* .

- Para cada categoria define-se então um *limiar* . Para entrar naquela categoria, o documento deve ter uma probabilidade X vezes maior do que de estar em qualquer outra categoria.

Método de Fisher

- Método alternativo, criado por R. A. Fisher, mostrou-se bastante preciso, em particular na filtragem de *spam.*
- É utilizado pelo plugin *SpamBayes* do Outlook
- Ao contrário do algoritmo de Bayes, que calcula Pr (*feature* | categoria), o método de Fisher calcula Pr (categoria | *feature*).
- As probabilidades são combinadas utilizando a distribuição chi-quadrado.

Método de Fisher

Precisamos calcular Pr(categoria | feature).
Como?

Intuitivamente temos:

$$\frac{\text{(número de documentos com essa feature e categoria)}}{\text{(número de documentos com essa feature)}}$$

Método de Fisher

Precisamos calcular Pr(categoria | feature).
Como?

Intuitivamente temos:

$$\frac{\text{(número de documentos com essa feature e categoria)}}{\text{(número de documentos com essa feature)}}$$

Ruim se recebemos muito mais documentos de uma categoria e pouquíssimos de outra!

Método de Fisher

Para evitar injustiças, vamos normalizar:

$$\Pr(\text{categoria}|\text{feature}) = \frac{\Pr(\text{feature} \mid \text{categoria})}{\Pr(\text{feature}|\text{categoria}) \text{ para todas categorias}}$$

Por exemplo, no caso de spam teríamos duas categorias (bom, spam), e então a conta ficaria assim:

$$\Pr(\text{"spam"}|\text{feature}) = \frac{\frac{\text{spams com essa feature}}{\text{spams total}}}{\frac{\text{spams com essa feature}}{\text{spams total}} + \frac{\text{bom's com essa feature}}{\text{bom's total}}}$$

Método de Fisher

Agora devemos combinar as probabilidades obtidas a partir de todas as features do documento:

- Multiplicamos todas elas
- Aplicamos o log natural no resultado
- Multiplicamos tudo por -2.

Fisher provou em 1950 que esse resultado deve estar em uma distribuição **chi-quadrado**.

Método de Fisher

Intuitivamente podemos pensar em como esses valores se comportam. Queremos testar se um documento(que na verdade não é spam!) é um spam:

- As probabilidades das *features* para a categoria spam terão valores baixos.
- Multiplicados, esses valores serão mais baixos ainda.
- O log dessa multiplicação será um valor negativo de módulo grande.
- Multiplicado por -2, esse valor será um positivo de módulo maior ainda.

Método de Fisher

Conclusão: Quanto menor a probabilidade de um documento estar em uma dada categoria, maior será o resultado dessa conta.

Método de Fisher

- Usando a inversa da chi-quadrado poderemos aplicar testes de hipóteses para decidir se um documento pertence ou não a uma categoria.
- A distribuição da chi-quadrado e sua inversa dependem também de um parâmetro k, o grau de liberdade.
- Para o método de Fisher o grau de liberdade será duas vezes o número de probabilidades que foram multiplicadas(k = 2x features).
- Portanto, a chi-quadrado dependerá do valor obtido ao combinarmos as probabilidades (aquela conta maluca, lembram?) e do número de probabilidades que combinamos.

Método de Fisher

E o que a inversa da chi-quadrado vai nos devolver?

- Uma probabilidade. Essa probabilidade será mais baixa a medida que o valor combinado das probabilidades for maior que o número de probabilidade que combinamos.

- E quanto mais baixa ela for, menor a chance dela pertencer a tal categoria (aquela...que usamos nas probabilidades da conta maluca).

Método de Fisher

E agora como classificamos o documento??

- Devemos obter os valores da inversa da chi-quadrado para todas as categorias.
- Definimos limites mínimos para que uma hipótese seja aceita. Exemplo: uma mensagem só será aceita como spam se a inversa da chi-quadrado retornar > 0.8.

- E se for aceito em várias categorias? Pegamos a maior probabilidade.
- E se não for aceito em nenhuma? Então não conseguimos classificar o documento!

Fim

Obrigado!