CS47300: Web Information Search and Management

Text Categorization
Prof. Chris Clifton
30 September 2020
Material adapted from course created by Dr. Luo Si, now leading Alibaba research group

Problem: Weighting of Terms

- K-NN treats all terms equally
  - Frequent but unimportant terms may dominate
- Which terms are more important?
  - TF.IDF?
  - …
- Solution – machine learning
  - We have training data

34

Naïve Bayes Classification

- Naïve Bayes (NB) Classification
  - Generative Model: Model both the input data (i.e., document contents) and output data (i.e., class labels)
  - Make strong assumption of the probabilistic modeling approach
- Methodology
  - Similar with the idea of language modeling approaches for information retrieval
  - Train a language model for all the documents in one category

Naïve Bayes Classification

- Methodology
  - Train a language model for all the documents in one category

$$\text{Category 1}: \left(\vec{d}_{1,1}, \vec{d}_{1,2}, \ldots, \vec{d}_{1,n_1}\right) \rightarrow \text{Language model } \theta_1$$

$$\text{Category 2}: \left(\vec{d}_{2,1}, \vec{d}_{2,2}, \ldots, \vec{d}_{2,n_2}\right) \rightarrow \text{Language model } \theta_2$$

......

$$\text{Category C}: \left(\vec{d}_{C,1}, \vec{d}_{C,2}, \ldots, \vec{d}_{C,n_K}\right) \rightarrow \text{Language model } \theta_C$$

  - What is the language model? (Multinomial distribution)
  - How to estimate the language model for all the documents in one category?

Naïve Bayes Classification

- Representation
  - Each document is a "bag of words" with weights (e.g., TF.IDF)
  - Each category is a super "bag of words", which is composed of all words in all the documents associated with the category
  - For all the words in a specific category c, it is modeled by a multinomial distribution as

$$p(\vec{d}_{c1},..,\vec{d}_{cn_c} \mid \theta_c)$$

  - Each category (c) has a prior distribution P(c), which is the probably of choosing category c BEFORE observing the content of a document

39

Naïve Bayes Classification

Maximum Likelihood Estimation:

- Find model parameters for a category that maximizes generation likelihood:

$$\theta_c^* = \arg\max_{\theta_c} p(\vec{d}_{c1},..,\vec{d}_{cn_c} \mid \theta_c)$$

There are K words in vocabulary, $w_1 \ldots w_K$

Data: documents $\vec{d}_{c1},\ldots,\vec{d}_{cn_c}$

For $\vec{d}_{ci}$ with counts $c_i(w_1)$, …, $c_i(w_k)$, and length $|\vec{d}_c|$

Model: multinomial M with parameters $\{p(w_k)\}$

Likelihood: $\Pr(\vec{d}_{c1},\ldots,\vec{d}_{cn_c} \mid \theta)$

$$\theta_c^* = \arg\max_{\theta_c} p(\vec{d}_{c1},..,\vec{d}_{cn_c} \mid \theta_c)$$

40

Maximum Likelihood Estimation (MLE)

$$p(\vec{d}_{c1},..,\vec{d}_{cn_c} \mid \theta) = \prod_{i=1}^{n_c} \binom{|\vec{d}_{ci}|}{c_{ci}(w_1)...c_{ci}(w_K)} \prod_{k=1}^{K} p_k^{c_{ci}(w_k)} \propto \prod_{i=1}^{n_c} \prod_k p_k^{c_{ci}(w_k)}$$

$$l(\vec{d}_{c1},..,\vec{d}_{cn_c} \mid \theta) = \log\ p(\vec{d}_{c1},..,\vec{d}_{cn_c} \mid \theta) = \sum_{i=1}^{n_c} \sum_k c_{ci}(w_k) \log\ p_k$$

$$l'(\vec{d}_{c1},..,\vec{d}_{cn_c} \mid \theta) = \sum_{i=1}^{n_c} \sum_k c_{ci}(w_k) \log\ \theta_k + \lambda(\sum_k p_k - 1)$$

$$\frac{\partial l'}{\partial p_k} = \frac{\sum_{i=1}^{n_c} c_{ci}(w_k)}{p_k} + \lambda = 0 \quad \Rightarrow \quad p_k = -\frac{\sum_{i=1}^{n_c} c_{ci}(w_k)}{\lambda}$$

Use Lagrange multiplier approach
Set partial derivatives to zero
Get maximum likelihood estimate

$$Since\ \sum_k p_k = 1,\ \lambda = -\sum_k \sum_{i=1}^{n_c} c_{ci}(w_k) = -\sum_{i=1}^{n_c} |\vec{d}_{ci}| \qquad So,\ p_k = p(w_k) = \frac{\sum_{i=1}^{n_c} c_{ci}(w_k)}{\sum_{i=1}^{n_c} |\vec{d}_{ci}|}$$

Naïve Bayes Classification

- **MLE Estimator: Normalization by simple counting**
  - Train a language model for all the documents in one category

$$p(w \mid \theta_c^*) = \frac{\sum_{i=1}^{n_c} c_{ci}(w)}{\sum_{i=1}^{n_c} |\vec{d}_{ci}|}$$

$$p(c) = \frac{n_c}{\sum_{c'} n_{c'}}$$

- **Category Prior:**
  - Number of documents in the category divided by the total number of documents

Naïve Bayes Classification

PURDUE UNIVERSITY
Department of Computer Science

- **Smoothed Estimator:**
  - Laplace Smoothing

$$p(w \mid \theta_c^*) = \frac{1 + \sum_{i=1}^{n_c} c_{ci}(w)}{K + \sum_{i=1}^{n_c} \left| \vec{d}_{ci} \right|}$$

Number of Words in Vocabulary

  - Hierarchical Smoothing

$$p(w \mid \theta_c^*) = \lambda_1 P(w \mid \theta_c^*) + \lambda_2 P(w \mid \theta_{c^{up1}}^*) .... + \lambda_m P(w \mid \theta_{c^{root}}^*)$$

  - Dirichlet Smoothing

Naïve Bayes Classification

PURDUE UNIVERSITY
Department of Computer Science

- **Prediction:**

$$\begin{aligned}
c^* &= \arg\max_c p(c \mid \vec{d}_i) \\
&= \arg\max_c \left\{ \frac{p(c)\, p(\vec{d}_i \mid c)}{p(\vec{d}_i)} \right\} \\
&= \arg\max_c \left\{ p(c)\, p(\vec{d}_i \mid c) \right\} \quad (Bayes\ Rule) \\
&= \arg\max_c \left\{ p(c) \prod_k p(w_k \mid c)^{c_i(w_k)} \right\} \quad (Multinomail\ Dist) \\
&= \arg\max_c \left\{ \log(p(c)) + \sum_k c_i(w_k) \log p(w_k \mid c) \right\}
\end{aligned}$$

Plug in the estimator

Naïve Bayes Classification

- **Example of Binary Classification**

  Two classes

$$c^* = \arg\max_{l\in[-,+]} p(c_l \mid \vec{d}_i) \rightarrow \frac{p(c_+ \mid \vec{d}_i)}{p(c_- \mid \vec{d}_i)}$$

$$p(c_+ \mid \vec{d}_i) \propto \prod_k [p(w_k \mid c_+)]^{c_i(w_k)} \frac{n_+}{n_+ + n_-}$$

$$p(c_- \mid \vec{d}_i) \propto \prod_k [p(w_k \mid c_-)]^{c_i(w_k)} \frac{n_-}{n_+ + n_-}$$

Naïve Bayes Classification

- **Example of Binary Classification**

$$c^* = \arg\max_{l\in[-,+]} p(c_l \mid \vec{d}_i) \rightarrow \frac{p(c_+ \mid \vec{d}_i)}{p(c_- \mid \vec{d}_i)}$$

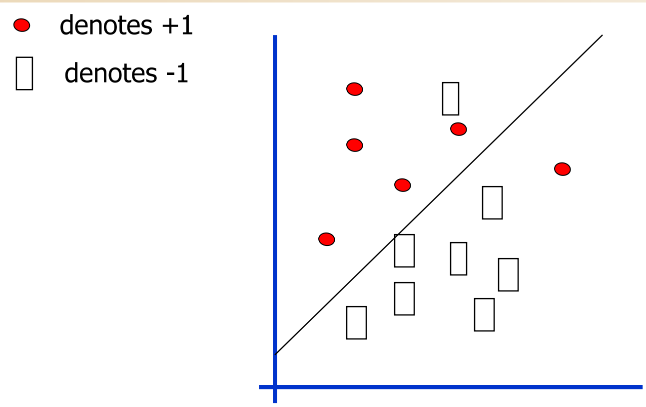
$$\log \frac{p(c_+ \mid \vec{d}_i)}{p(c_- \mid \vec{d}_i)} = \log \left\{ \frac{\prod_k [p(w_k \mid c_+)]^{c_i(w_k)} \frac{n_+}{n_+ + n_-}}{\prod_k [p(w_k \mid c_-)]^{c_i(w_k)} \frac{n_-}{n_+ + n_-}} \right\}$$

$$= \boxed{\log\left(\frac{n_+}{n_-}\right)} + \sum_k c_i(w_k) \boxed{\log\left(\frac{p(w_k \mid c_+)}{p(w_k \mid c_-)}\right)}$$

$$\log \frac{p(c_+ \mid \vec{d})}{p(c_- \mid \vec{d})} \propto \boxed{b_0} + \sum_k c_i(w_k) \times \boxed{weight(w_k)}$$

PURDUE UNIVERSITY
Department of Computer Science

Naïve Bayes = Linear Classifier



$$\log \frac{p(c_+ \mid \vec{d}_i)}{p(c_- \mid \vec{d}_i)} \propto b_0 + \sum_k c_i(w_k) \times weight(w_k)$$

PURDUE UNIVERSITY
Department of Computer Science

Naïve Bayes Classification

- Summary
  - Utilize multinomial distribution for modeling categories and documents
  - Use posterior distribution (posterior of category given document) to predict optimal category
- Pros
  - Solid probabilistic foundation
  - Fast online response, linear classifier for binary classification
- Cons
  - Empirical performance not very strong
  - Probabilistic model for each category is estimated to maximize the data likelihood for documents in the category (generative), not for purpose of distinguishing documents in different categories (discriminative)

52