Text Classification using Naive Bayes

Guido Sanguinetti

Informatics 2B— Learning and Data Lecture 7
28 February 2012

Overview

Today's lecture

- Naive Bayes text classification
- Two models to estimate $P(\text{Document} \mid \text{Class})$
  - Bernoulli Model
  - Multinomial Model
- Comparing the two models

Text Classification using Bayes Theorem

- Document D, with class c_k
- Classify D as the class with the highest posterior probability:

$$P(c_k \mid D) = \frac{P(D \mid c_k)P(c_k)}{P(D)} \propto P(D \mid c_k)P(c_k)$$

Text Classification using Bayes Theorem

- Document D, with class c_k
- Classify D as the class with the highest posterior probability:

$$P(c_k \mid D) = \frac{P(D \mid c_k)P(c_k)}{P(D)} \propto P(D \mid c_k)P(c_k)$$

- How do we represent D? How do we estimate $P(D \mid c_k)$?

Text Classification using Bayes Theorem

- Document D, with class c_k
- Classify D as the class with the highest posterior probability:

$$P(c_k \mid D) = \frac{P(D \mid c_k)P(c_k)}{P(D)} \propto P(D \mid c_k)P(c_k)$$

- How do we represent D? How do we estimate $P(D \mid c_k)$?
- **Bernoulli document model:** a document is represented by a binary feature vector, whose elements indicate absence or presence of corresponding word in the document

Text Classification using Bayes Theorem

- Document D, with class c_k
- Classify D as the class with the highest posterior probability:

$$P(c_k \mid D) = \frac{P(D \mid c_k)P(c_k)}{P(D)} \propto P(D \mid c_k)P(c_k)$$

- How do we represent D? How do we estimate $P(D \mid c_k)$?
- **Bernoulli document model:** a document is represented by a binary feature vector, whose elements indicate absence or presence of corresponding word in the document
- **Multinomial document model:** a document is represented by an integer feature vector, whose elements indicate frequency of corresponding word in the document

Naive Bayes: Bernoulli Document Model

- We have a *vocabulary* V containing a set of $|V|$ words
- Dimension t of a document vector corresponds to word w_t in the vocabulary
- $P(w_t \mid c_k)$ is the probability of word w_t occurring in document of class c_k; $(1 - P(w_t \mid c_k))$ is probability of w_t not occurring.
- Generative model:
  - for each word w
  - flip a (biased) coin, with probability of heads $P(w \mid c_k)$
  - if heads, w is included the document

  We thus generate a document containing the selected words

  But no count information for each word

Naive Bayes: Bernoulli Document Model

- $\mathbf{B}_i$ is the feature vector for the ith document D^i
- B_{it}, is either 0 or 1 representing the absence or presence of word w_t in the ith document

$$P(B_{it} \mid c_k) = B_{it} P(w_t \mid c_k) + (1 - B_{it})(1 - P(w_t \mid c_k))$$

- Naive Bayes:

$$\begin{aligned} P(\mathbf{B}_i \mid c_k) &= \prod_{t=1}^{|V|} P(B_{it} \mid c_k) \\ &= \prod_{t=1}^{|V|} [B_{it} P(w_t \mid c_k) + (1 - B_{it})(1 - P(w_t \mid c_k))] \end{aligned}$$

Parameters of Bernoulli Model

- Parameters of the model are:

Parameters of Bernoulli Model

- Parameters of the model are:
  - likelihoods of each word given the document class $P(w_t \mid c_k)$

Parameters of Bernoulli Model

- Parameters of the model are:
  - likelihoods of each word given the document class $P(w_t \mid c_k)$
  - prior probabilities $P(c_k)$

Parameters of Bernoulli Model

- Parameters of the model are:
  - likelihoods of each word given the document class $P(w_t \mid c_k)$
  - prior probabilities $P(c_k)$
- Let $n_k(w_t)$ be the number of documents of class c_k in which w_t is observed, and let N_k be the total number of documents in c_k

Parameters of Bernoulli Model

- Parameters of the model are:
  - likelihoods of each word given the document class $P(w_t \mid c_k)$
  - prior probabilities $P(c_k)$
- Let $n_k(w_t)$ be the number of documents of class c_k in which w_t is observed, and let N_k be the total number of documents in c_k
- Estimate the word likelihoods as:

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

Parameters of Bernoulli Model

- Parameters of the model are:
  - likelihoods of each word given the document class $P(w_t \mid c_k)$
  - prior probabilities $P(c_k)$
- Let $n_k(w_t)$ be the number of documents of class c_k in which w_t is observed, and let N_k be the total number of documents in c_k
- Estimate the word likelihoods as:

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

- Estimate priors as

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

Training a Bernoulli Model

1. Define the vocabulary V
2. Count in the training set:
   - N (number of documents)
   - N_k (number of documents of class c_k)
   - $n_k(w_t)$ (number of documents of class c_k containing w_t)
3. Estimate likelihoods $P(w_t \mid c_k)$
4. Estimate priors $P(c_k)$

Classifying with the Bernoulli Model

To classify an unlabelled document D^j, we estimate the posterior probability for each class:

$$\begin{aligned} P(c_k \mid D^j) &= P(c_k \mid \mathbf{B}_j) \\ &\propto P(\mathbf{B}_j \mid c_k) P(c_k) \\ &\propto P(c_k) \prod_{t=1}^{|V|} [B_{jt} P(w_t \mid c_k) + (1 - B_{jt})(1 - P(w_t \mid c_k))] \end{aligned}$$

Example

Consider a set of documents each of which is related either to *Sports* (S) or to *Informatics* (I).

We define a vocabulary V of eight words:

$$
\begin{aligned}
w_1 &= \text{goal} \\
w_2 &= \text{tutor} \\
w_3 &= \text{variance} \\
w_4 &= \text{speed} \\
w_5 &= \text{drink} \\
w_6 &= \text{defence} \\
w_7 &= \text{performance,} \\
w_8 &= \text{field}
\end{aligned}
$$

Example(cont.)

Training data (each corresponds to a document, each column corresponds to a word):

$$\mathbf{B}^{\text{Sport}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$\mathbf{B}^{\text{Inf}} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Example(cont.)

Classify the following:

1. $B_1 = [1\ 0\ 0\ 1\ 1\ 1\ 0\ 1]$
2. $B_2 = [0\ 1\ 1\ 0\ 1\ 0\ 1\ 0]$

Multinomial model

- Document feature vectors capture *word frequency* information (not just presence or absence)
- As in the Bernoulli model
  - Vocabulary V containing a set of $|V|$ words
  - Dimension t of a document vector corresponds to word w_t in the vocabulary
  - $P(w_t \mid c_k)$ is the probability of word w_t occurring in document of class c_k
- Multinomial generative model
  - consider a (biased) $|V|$-sided dice
  - each side i corresponds to word w_i with probability $P(w_t \mid c_k)$
  - at each position in the document roll the dice and insert the corresponding word

Generates a document as a *bag* of words — includes what words are in the document, and how many times they occur

Multinomial model

- $\mathbf{M}_i$ is the multinomial model feature vector for the ith document D^i
- M_{it}, is the number of times word w_t occurs in document D^i; $n_i = \sum_t M_{it}$ is the total number of words id D^i
- Estimate $P(w_t \mid c_k)$ using word frequency information from the multinomial model feature vectors

Multinomial model: Naive Bayes approximation

- Naive Bayes approximation: Generation of documents is modelled by repeatedly drawing words from a multinomial distribution

$$P(\mathbf{M}_i \mid c_k) = \frac{n_i!}{\prod_{t=1}^{|V|} M_{it}!} \prod_{t=1}^{|V|} P(w_t \mid c_k)^{M_{it}}$$

Multinomial model: Naive Bayes approximation

- Naive Bayes approximation: Generation of documents is modelled by repeatedly drawing words from a multinomial distribution

$$P(\mathbf{M}_i \mid c_k) = \frac{n_i!}{\prod_{t=1}^{|V|} M_{it}!} \prod_{t=1}^{|V|} P(w_t \mid c_k)^{M_{it}}$$

- If comparing likelihoods of the same document for different classes (e.g. $P(\mathbf{M}_i \mid c_k)$ vs. $P(\mathbf{M}_i \mid c_j)$), then

$$P(\mathbf{M}_i \mid c_k) \propto \prod_{t=1}^{|V|} P(w_t \mid c_k)^{M_{it}}$$

Multinomial model: Naive Bayes approximation

- Naive Bayes approximation: Generation of documents is modelled by repeatedly drawing words from a multinomial distribution

$$P(\mathbf{M}_i \mid c_k) = \frac{n_i!}{\prod_{t=1}^{|V|} M_{it}!} \prod_{t=1}^{|V|} P(w_t \mid c_k)^{M_{it}}$$

- If comparing likelihoods of the same document for different classes (e.g. $P(\mathbf{M}_i \mid c_k)$ vs. $P(\mathbf{M}_i \mid c_j)$), then

$$P(\mathbf{M}_i \mid c_k) \propto \prod_{t=1}^{|V|} P(w_t \mid c_k)^{M_{it}}$$

- Since $x^0 = 1$ the above product is only affected by words in the D^i. If D^i is a sequence of ℓ words, $u_1, u_2, \ldots, u_\ell$:

$$P(\mathbf{M}_i \mid c_k) \propto \prod_{h=1}^{\ell} P(u_h \mid c_k)$$

Parameters of a multinomial model

- As for the Bernoulli model, the model parameters are:

Parameters of a multinomial model

- As for the Bernoulli model, the model parameters are:
  - likelihoods of each word given the document class $P(w_t \mid c_k)$

Parameters of a multinomial model

- As for the Bernoulli model, the model parameters are:
  - likelihoods of each word given the document class $P(w_t \mid c_k)$
  - prior probabilities $P(c_k)$

Parameters of a multinomial model

- As for the Bernoulli model, the model parameters are:
  - likelihoods of each word given the document class $P(w_t \mid c_k)$
  - prior probabilities $P(c_k)$
- Let $z_{ik} = 1$ when D^i has class c_k; $z_{ik} = 0$ otherwise

Parameters of a multinomial model

- As for the Bernoulli model, the model parameters are:
  - likelihoods of each word given the document class $P(w_t \mid c_k)$
  - prior probabilities $P(c_k)$
- Let $z_{ik} = 1$ when D^i has class c_k; $z_{ik} = 0$ otherwise
- If N is the total number of documents then:

$$\hat{P}(w_t \mid c_k) = \frac{\sum_{i=1}^{N} M_{it} z_{ik}}{\sum_{s=1}^{|V|} \sum_{i=1}^{N} M_{is} z_{ik}}$$

Estimate $P(w_t \mid c_k)$ as relative frequency of w_t in documents of class c_k with respect to the total number of words in documents of that class

Parameters of a multinomial model

- As for the Bernoulli model, the model parameters are:
  - likelihoods of each word given the document class $P(w_t \mid c_k)$
  - prior probabilities $P(c_k)$
- Let $z_{ik} = 1$ when D^i has class c_k; $z_{ik} = 0$ otherwise
- If N is the total number of documents then:

$$\hat{P}(w_t \mid c_k) = \frac{\sum_{i=1}^{N} M_{it} z_{ik}}{\sum_{s=1}^{|V|} \sum_{i=1}^{N} M_{is} z_{ik}}$$

  Estimate $P(w_t \mid c_k)$ as relative frequency of w_t in documents of class c_k with respect to the total number of words in documents of that class

- Estimate priors as before

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

Training a multinomial model

1. Define the vocabulary V; the number of words in the vocabulary defines the dimension of the feature vectors

Training a multinomial model

1. Define the vocabulary V; the number of words in the vocabulary defines the dimension of the feature vectors
2. Count the following in the training set:

Training a multinomial model

1. Define the vocabulary V; the number of words in the vocabulary defines the dimension of the feature vectors
2. Count the following in the training set:
   - N the total number of documents

Training a multinomial model

1. Define the vocabulary V; the number of words in the vocabulary defines the dimension of the feature vectors
2. Count the following in the training set:
   - N the total number of documents
   - N_k the number of documents labelled with class c_k, for all classes

Training a multinomial model

1. Define the vocabulary V; the number of words in the vocabulary defines the dimension of the feature vectors
2. Count the following in the training set:
   - N the total number of documents
   - N_k the number of documents labelled with class c_k, for all classes
   - M_{it} the frequency of word w_t in document D^i for all words in V and all documents

Training a multinomial model

1. Define the vocabulary V; the number of words in the vocabulary defines the dimension of the feature vectors
2. Count the following in the training set:
   - N the total number of documents
   - N_k the number of documents labelled with class c_k, for all classes
   - M_{it} the frequency of word w_t in document D^i for all words in V and all documents
3. Estimate the likelihoods $P(w_t \mid c_k)$

Training a multinomial model

1. Define the vocabulary V; the number of words in the vocabulary defines the dimension of the feature vectors
2. Count the following in the training set:
   - N the total number of documents
   - N_k the number of documents labelled with class c_k, for all classes
   - M_{it} the frequency of word w_t in document D^i for all words in V and all documents
3. Estimate the likelihoods $P(w_t \mid c_k)$
4. Estimate the priors $P(c_k)$

Classifying with a multinomial model

To classify an unlabelled document D^j, we estimate the posterior probability for each class:

$$
\begin{aligned}
P(c_k \mid D^j) &= P(c_k \mid \mathbf{M}_j) \\
&\propto P(\mathbf{M}_j \mid c_k) P(c_k) \\
&\propto P(c_k) \prod_{t=1}^{|V|} P(w_t \mid c_k)^{M_{it}} \\
&\propto P(c_k) \prod_{h=1}^{len(D^i)} P(u_h \mid c_k)
\end{aligned}
$$

Question

Consider the word "the". What will be the approximate value of the probability $P(\text{"the"} \mid c_k)$ in
(a) the Bernoulli model;
(b) the multinomial model?

The zero probability problem

- Consider Naive Bayes multinomial likelihood estimate:

$$\hat{P}(w_t \mid c_k) = \frac{\sum_{i=1}^{N} M_{it} z_{ik}}{\sum_{s=1} |V| \sum_{i=1}^{N} M_{is} z_{ik}}$$

If w_t never appears in class c_k then $M_{it} z_{ik} = 0$ for all documents i. This means $\hat{P}(w_t \mid c_k) = 0$

The zero probability problem

- Consider Naive Bayes multinomial likelihood estimate:

$$\hat{P}(w_t \mid c_k) = \frac{\sum_{i=1}^{N} M_{it} z_{ik}}{\sum_{s=1} |V| \sum_{i=1}^{N} M_{is} z_{ik}}$$

If w_t never appears in class c_k then $M_{it} z_{ik} = 0$ for all documents i. This means $\hat{P}(w_t \mid c_k) = 0$

- Naive Bayes involves a product of probabilities

$$P(\mathbf{M}_i \mid c_k) \propto \prod_{h=1}^{\ell} P(u_h \mid c_k)$$

If we have a document for which our estimate of $P(u_h \mid c_k) = 0$, then $P(\mathbf{M}_i \mid c_k) = 0$: i.e. it is impossible for the document to belong to the class c_k!

The zero probability problem

- Consider Naive Bayes multinomial likelihood estimate:

$$\hat{P}(w_t \mid c_k) = \frac{\sum_{i=1}^{N} M_{it} z_{ik}}{\sum_{s=1} |V| \sum_{i=1}^{N} M_{is} z_{ik}}$$

  If w_t never appears in class c_k then $M_{it} z_{ik} = 0$ for all documents i. This means $\hat{P}(w_t \mid c_k) = 0$

- Naive Bayes involves a product of probabilities

$$P(\mathbf{M}_i \mid c_k) \propto \prod_{h=1}^{\ell} P(u_h \mid c_k)$$

  If we have a document for which our estimate of $P(u_h \mid c_k) = 0$, then $P(\mathbf{M}_i \mid c_k) = 0$: i.e. it is impossible for the document to belong to the class c_k!

- **If a word does not occur in the training data for a class that does not mean it cannot occur in any document of that class**

Add-one smoothing

- The maximum likelihood estimated is an underestimate for words that do not appear in the training data

Add-one smoothing

- The maximum likelihood estimated is an underestimate for words that do not appear in the training data
- We would like $P(w \mid c_k) > 0$ for all words and all classes

Add-one smoothing

- The maximum likelihood estimated is an underestimate for words that do not appear in the training data
- We would like $P(w \mid c_k) > 0$ for all words and all classes
- *Add one smoothing* (Laplace's law of succession): add a count of 1 to the count of each word type:

$$P_{\mathrm{Lap}}(w_t \mid c_k) = \frac{1 + \sum_{i=1}^{N} M_{it} z_{ik}}{|V| + \sum_{s=1}^{|V|} \sum_{i=1}^{N} M_{is} z_{ik}}$$

Note that the denominator is modified to take account of the additional count of one for each word type

Add-one smoothing

- The maximum likelihood estimated is an underestimate for words that do not appear in the training data
- We would like $P(w \mid c_k) > 0$ for all words and all classes
- *Add one smoothing* (Laplace's law of succession): add a count of 1 to the count of each word type:

$$P_{\mathrm{Lap}}(w_t \mid c_k) = \frac{1 + \sum_{i=1}^{N} M_{it} z_{ik}}{|V| + \sum_{s=1}^{|V|} \sum_{i=1}^{N} M_{is} z_{ik}}$$

  Note that the denominator is modified to take account of the additional count of one for each word type
- Add-one smoothing is often used in practice.

Add-one smoothing

- The maximum likelihood estimated is an underestimate for words that do not appear in the training data
- We would like $P(w \mid c_k) > 0$ for all words and all classes
- *Add one smoothing* (Laplace's law of succession): add a count of 1 to the count of each word type:

$$P_{\mathrm{Lap}}(w_t \mid c_k) = \frac{1 + \sum_{i=1}^{N} M_{it} z_{ik}}{|V| + \sum_{s=1}^{|V|} \sum_{i=1}^{N} M_{is} z_{ik}}$$

  Note that the denominator is modified to take account of the additional count of one for each word type
- Add-one smoothing is often used in practice.
- (There are many other possible ways to smooth such estimates)

Comparing multinomial with Bernoulli

	Bernoulli	**Multinomial**
Generative model	draw a document from a multidimensional Bernoulli distribution	draw a words from a multinomial distribution
Document representation	Binary vector	Vector of frequencies
Multiple occurences	Ignored	Taken into account
Document length	Best for short docs	longer docs OK
Feature vector dimension	Shorter	Longer OK
Behaviour with "the"	$P(\text{"the"} \mid c_k) \sim 1.0$	$P(\text{"the"} \mid c_k) \sim 0.05$
Non-occurring words	affect likelihood	do not affect likelihood

Summary

- Bernoulli Model for text classification
- Multinomial model: an alternative Naive Bayes model that takes word frequencies into account
- The zero probability problem
- Next lecture: using Bayes' Theorem with continuous valued data