Implement Naïve Bayes Algorithm with Bernoulli models and event models

Haoyu Zheng

Abstract: *Article classification is the task of classifying documents by their content: that is, by the words of which they are comprised. This problem is of great and increasing importance, mainly because of the recent explosive increase of online unlabeled articles. In this report, a learning algorithm take input as a set of labelled articles and attempts to predict a new article into their categories. Article classification is used as a concept machine learning problem where a single isolated document is an instance. Recent approaches article classification has used two different probabilistic models for classification, both of them make the naïve Bayes assumption. One of them is a multi-variate Bernoulli event model, that is, a Bayesian Network with no dependencies between words and binary word features. Another one is multinomial event model, that is, a model with an integer word counts. In this report we are interested to make clear between the confusion by describing the differences and details of two models. This report also explains how the articles are classified and predict the class label using both above model with simple examples.*

Keywords*: Bayes assumption, machine learning, multi-variate Bernoulli event model, Multinomial event model.*

1. Introduction.

In the present days, people read several articles on a daily basis. The importance of article categorization has alarming increased due to the information hidden in the articles that is useful for knowing the articles belongs to what type of class. The method of article categorization is to assign article to more predefined category or predefined class based on its content. Articles which already been labelled called training data. An article classifier is used to determine the class or label of an article according to words occur in that article. Extracting feature is other important method before classifying the different type of articles. Good feature will provide a good performance and save a lot of time. In contrast, useless feature will cost generous time and energy.

2. Naïve Bayes Classifier

2.1 Overview.

Article Classification using Naïve Bayes classifier is underlying on Bayes theorem. It is the probability model that was formulated by Thomas Bayes (1701 – 1761) [1]. Naïve Bayes classifier is easy to implement, but it has strong power on article classification.

2.2 Create Dictionary

Creating Dictionary is a method most used in article classification. We extract features from articles to create a dictionary of all the unique words occurring in all the articles. These features can be used for training Bayes algorithm. Each feature is represented by a different word. We collect all the words from all the articles, then we will set a list of stop words. Stop words are the words which are useless or unimportant for classifying an article, such as “a”, “the” and “in” and stuff like that. Eliminating unimportant features can improve the accuracy and increase the speed of running an algorithm. Words in dictionary are sorted in alphabetical order.

2.3 Extracting features of Training Sets

We set every article as a single vector Xi. The length of vector Xi equal to the length of the dictionary. If the words occur in an article also occur in dictionary. We will set the vector at this word position to be 1. If the words occur in dictionary not in the article, we will set it as 0.

Dictionary = [‘ability’, ‘able’, ‘abnormalities’, ‘above’, ‘abramson’, ‘absent’ ...’ziggy’]

X_i = [0, 1, 0, 0, 0, 1, …, 0]

The dimension of Xi is equal to the size of the dictionary. In this way, we can represent an article via a feature vector.

2.4 Building Multi-variate Bernoulli model

To model $p(x|y)$, we make a very strong assumption. We assume that the X_i’s re conditionally independent given y. Therefore, for calculating $p\ (x_1,\ldots, x_{3000}|\ y)$, we have:

$$p(\mathrm{x}_1, \ldots, \mathrm{x}_{3000}|y) = \ p(x_1|y)p(x_2|y)\cdots p(x_{3000}|y) = \prod_{j=1}^{n} p(x_j|y)$$

Since we have a multiclassification problem, so our model is parameterized by $\emptyset_{j|y=2} = p(x_j = 1|y = 2)$, $\emptyset_{j|y=1} = p(x_j = 1|y = 1)$, $\emptyset_{j|y=0} = p(x_j = 1|y = 0)$, and $\emptyset_y = p(y = 1)$. We give different type of articles with different labels. We label medical articles, music articles and sports articles as 0, 1 and 2, respectively.

Then we maximum likelihood estimates:

$$\emptyset_{j|y=0} = \frac{\sum_{i=1}^{m} 1\{x_j^{(i)} = 1\ \Lambda\ y^{(i)} = 0\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 0\}}$$

$$\emptyset_{j|y=1} = \frac{\sum_{i=1}^{m} 1\{x_j^{(i)} = 1 \Lambda\, y^{(i)} = 1\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}}$$

$$\emptyset_{j|y=2} = \frac{\sum_{i=1}^{m} 1\{x_j^{(i)} = 2 \Lambda\, y^{(i)} = 2\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 2\}}$$

$$\emptyset_{y} = \frac{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}}{m}$$

In the above equations, the "Λ" sign represent "and". The numerator of parameter $\emptyset_{j|y=0}$ means how many times word j appears in medical articles. The denominator of parameter $\emptyset_{j|y=0}$ means that the number of articles is medical articles. Thus, the parameter of $\emptyset_{j|y=0}$ is the fraction of medical articles in which word j does appear. The parameter $\emptyset_{j|y=1}$ and $\emptyset_{j|y=2}$ have the same meaning as parameter $\emptyset_{j|y=0}$.

Predicting class prior on test data, we need to calculate:

$$p(y = 0|x) = \frac{p(x|y = 0)p(y = 0)}{p(x)}$$

Since the denominator of above equation is same for all of class, so we will ignore the denominator and then only compare numerator. In case numerator become so small, we will take the log of numerator to transform multiply sign to plus sign. We have:

$$\begin{aligned}\log\big(\, p(x|y = 0)p(y = 0)\big) \\ &= \log(\prod_{j=1}^{n} p\big(x_j\big|y = 1\big)p(y = 1)) \\ &= \sum_{j=1}^{n} \log(p\big(x_j\big|y = 1\big) + \log(p(y = 1)\end{aligned}$$

We compare value of above expression between different type of class, the class will be determined by the value we obtained. Therefore, we pick whichever class has the greater value.

2.5 Laplace smoothing

Naïve Bayes algorithm as we discussed will work for most of problems. But there's an exception, if have a word appear in the testing set but not appear in the training set. You will have:

$$\emptyset_{j|y=0} = \frac{\sum_{i=1}^{m} 1\{x_j^{(i)} = 1 \Lambda\, y^{(i)} = 0\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 0\}} = 0$$

Then your program will blow out, because you will have zero on the numerator when you predict the class prior (we ignore the denominator due to every class have same value on denominator). If you take of log of numerator, there's still a problem due to log(0) which value will goes to infinite. Therefore, a useful method called Laplace smoothing has been introduced. It replaces the above estimate with

$$\emptyset_j = \frac{\sum_{i=1}^{m} 1\{z^{(i)} = j\} + 1}{m + k}$$

In above equation, we add one to the numerator, and k to the denominator. K is equal to the number of classes we have. So, in the report k is equal to 3. Backing to our Bayes classifier, with Laplace smoothing, we obtain the following estimates of the parameter:

$$\emptyset_{j|y=0} = \frac{\sum_{i=1}^{m} 1\left\{x_j^{(i)} = 1 \wedge y^{(i)} = 0\right\} + 1}{\sum_{i=1}^{m} 1\{y^{(i)} = 0\} + 3}$$

$$\emptyset_{j|y=1} = \frac{\sum_{i=1}^{m} 1\left\{x_j^{(i)} = 1 \wedge y^{(i)} = 1\right\} + 1}{\sum_{i=1}^{m} 1\{y^{(i)} = 1\} + 3}$$

$$\emptyset_{j|y=2} = \frac{\sum_{i=1}^{m} 1\left\{x_j^{(i)} = 2 \wedge y^{(i)} = 2\right\} + 1}{\sum_{i=1}^{m} 1\{y^{(i)} = 2\} + 3}$$

2.6 Building Multinomial Event Model

Multinomial event model is another import model which specifies that an article is represented by the set of word occurrences from the article. In the article the number of occurrences of every single word is estimated. While calculating the probability of an article, multiplies the probability of the words that appear. For example, if the word 'music' occurs three times and the word 'musician' occurs two times, we will set feature vector x at 'music' position to be 3 and at 'musician' position to be 2.

$$\text{dict} = \{\cdots\cdots, \text{music}, \text{musician}, \cdots\cdots\}$$

$$\text{x} = [\cdots\cdots, 3, 2, \cdots\cdots]$$

The parameters $\emptyset_y = p(y)$ of the multinomial event model is the same as multi-variate event model. But for parameter $\emptyset_{j|y=0}$, $\emptyset_{j|y=1}$and$\emptyset_{j|y=2}$ will become a little different.

For maximum likelihood estimates of the parameters in multinomial event model, we have

$$\emptyset_{k|y=2} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n_i} 1\{x_j^{(i)} = k \wedge y^{(i)} = 2\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 2\} n_i}$$

$$\emptyset_{k|y=1} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n_i} 1\{x_j^{(i)} = k \wedge y^{(i)} = 1\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 1\} n_i}$$

$$\emptyset_{k|y=0} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n_i} 1\{x_j^{(i)} = k \wedge y^{(i)} = 0\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 0\} n_i}$$

$$\emptyset_{y} = \frac{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}}{m}$$

The numerator of the parameter $\emptyset_{k|y=2}$ represents the total number of j-th word appear in the sport articles. The denominator of the parameter $\emptyset_{k|y=2}$ stands for the number of words occur in sports articles. Same meaning for parameter of $\emptyset_{k|y=1}$ and $\emptyset_{k|y=0}$. If we apply Laplace soothing when estimating $\emptyset_{k|y=2}$, $\emptyset_{k|y=1}$ and $\emptyset_{k|y=0}$, we plus one to the numerators and the size of dictionary to the denominators, and obtain:

$$\emptyset_{k|y=2} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n_i} 1\left\{x_j^{(i)} = k \wedge y^{(i)} = 2\right\} + 1}{\sum_{i=1}^{m} 1\{y^{(i)} = 2\} n_i + |D|}$$

$$\emptyset_{k|y=1} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n_i} 1\left\{x_j^{(i)} = k \wedge y^{(i)} = 1\right\} + 1}{\sum_{i=1}^{m} 1\{y^{(i)} = 1\} n_i + |D|}$$

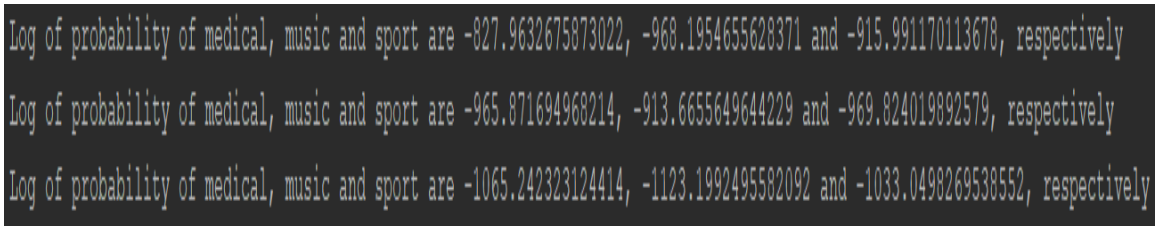
$$\emptyset_{k|y=0} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n_i} 1\left\{x_j^{(i)} = k \wedge y^{(i)} = 0\right\} + 1}{\sum_{i=1}^{m} 1\{y^{(i)} = 0\} n_i + |D|}$$

Where |D| represents the size of the dictionary we have created early.

3. Results

3.1 Multivariate Event Model's Result

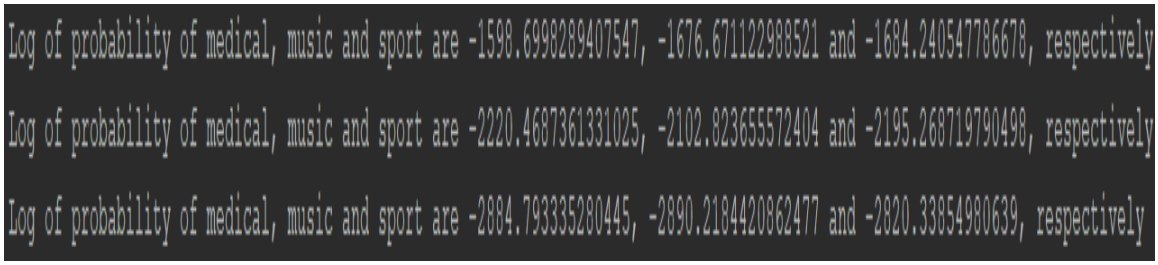
I try both model with or without Laplace smoothing. For the multi-variate event model without Laplace smoothing, we get the zero result on predicting class prior. But for this model with the Laplace smoothing, we obtain a really good performance.



We set three different type of articles as our testing data. The test sequence is medical article, music article and sport article. From above figure, we can see that for the first statement medical obtain the greatest value, Therefore, it has maximum likelihood to be the medical article. The model successful predict the class of this article. The second statement says music will be the most likelihood class of that article. It also predicts successfully. From the third statement, we can see the prediction of the third testing data is also successful.

3.2 Multinomial Event Model's Result

I test multinomial event model without Laplace smoothing, unfortunately, I obtain the same result as the multivariate event model. The program blow out due to the log(0) which is not exist. The multinomial event model with Laplace smoothing have great performance as multivariate event model.



The sequence of testing data is same. From above figure, we can see the first article is predicted as medical article. Therefore, we get the correct result. The second and the third statement say that article could be music article and sport article, separately. It also predicts well. Therefore, we come to the conclusion that the performance of our model is prefect.

4. Conclusion

Empirical comparisons provide evidence that the multinomial event model tends to perform better than multi-variate Bernoulli model if the dictionary size is relatively large [2]. Because multi-variate Bernoulli model cannot the frequency of a word. The weakness of the naïve Bayes classifier is that it highly dependent on the choice of features. Marked impacts are produced due to choice of stop word removal, stemming, and token-length. But it is still a powerful classifier. The strength of the naïve Bayes

classifier is obvious. It has a high degree of accuracy and easy to implement. From our result, we can see very clearly that the accuracy of naïve Bayes classifier with both models are pretty high. It achieves 100% accuracy. The Laplace smoothing method is very useful for estimate a parameter, from above results we can find out that without Laplace smoothing, we cannot run each of model successfully due to zero result. But with Laplace smoothing, those problems are solved completely.

Reference.

[1] G. Krishnaveni, Prof. T. Sudha, et al. Naïve Bayes Text Classification-A Comparison of Event Models. Volume 3, Issue 1, 2017.

[2]. Andrew McCallum, Kamal Nigam, et al. A comparison of event models for naive Bayes text classification. In AAAI-98 workshop on learning for text categorization, volume 752, pages 41-48. Citeseer, 1998.