Xudong LIU

7lxd@tongji.edu.cn

School of Software Engineering

Tongji University

Nov 22, 2015

Naive Bayes Classifier

ABSTRACT

This document is going to talk about what I have done with Naive Bayes classifier in text classification field. I will use the Reuters data (source:https://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection）, which has been preprocessed to a cleaning data set with less than 300 words using mutual information. First of all , I have to convert the textual data to vector space by performing TF/IDF transformation. Then I build Naive Bayes classifier (*source code: https://github.com/toshiakit/Naive-Bayes*) using 5-fold cross-validation. At last the accuracy, precision, recall and F-measure are estimated to show the quantity of Naive Bayes classifier on the this binary text classification with Reuters data.

Keywords: Data Mining, Naive Bayes Classifier, TF/IDF, Cross-Validation, Accuracy, Sensitivity, Specificity, Precision.

INTRODUCTION

Bayes classification is based on a set of data, which we call them training set, to build a model for class attribute as a function of the values of other attributes. What we hope to do is that to assign a class of previously unseen records as accurately as possible. A test set is used to determine the accuracy of the model. However, in this experiment, we will use 5-fold cross validation to testing the accuracy of the model we just build.

Naive Bayes is another algorithm for modelling classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. All naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable.

To classifier text or to document categorization is a problem in library science, information science and computer science. The goal is to assign a document to one or more classes or categories.

We will use naive bayes classification to build a model to solve the text classification problem. The data we use is Reuters data, one of the benchmark datasets for text categorization and natural language processing. It is a collection of articles, that consists of articles in the top 10 categories. The task is to assign a topic to a new article. This dataset has been preprocessed to reduce the vocabulary size to 300 words using mutual information. The original data is from https://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection.

DESIGN & METHODS

The basic idea for implementing naive bayes modelling is as follows:

- Load data from train set.
- Extract Terms from file.
- Calculate TF/IDF for each term and convert the textual data to vector space.
- Create Naive Bayes classifier.
- Execute 5-Fold cross validation and predict the label.
- Estimate accuracy, precision, recall and F-measure.

In this experiment, we will repeat the above steps for all the 10 categories.

2.1 Load data

- Load the original data from the reuters data.
- Input: the data path.
- Output: every words count & label.
- Process: Open <acq.train> and use for-loop to collect the term in the training set if the term doesn't exist in the cell array.

2.2 TF-IDF

- Calculator every words TF-IDF
- Input: words count.
- Output: Y(the TF-IDF value of every word)
- How to calculate the TF-IDF:

$$\mathrm{tf}(t,d) = 0.5 + \frac{0.5 \times f_{t,d}}{\max\{f_{t,d} : t \in d\}}$$

$$\mathrm{idf}(t,D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

figure 2.1

With the two formulas, we can calculate every word's TF-IDF.

2.3 Naive Bayes Classification Modelling

I find the implementation of naive bayes from the github(source:*https://github.com/toshiakit/NaiveBayes*). Compared with matlab's fitcnb, it improved the accuracy and speed. Therefore, I used this code to build my naive bayes model.

2.4 Evaluation

Used the 5-fold cross-validation to evaluate the quality of the naive bayes model we just build.

EXPERIEMENT

3.1 Structure of the Code

• CalculatorBayes.m	The main method to control the whole program running
• LoadData.m	Load original data from the disk (Reuters data)
• mySpamFilter.m	Naive bayes model
• myNaiveBayes.m	Naive bayes Model

3.2 Result

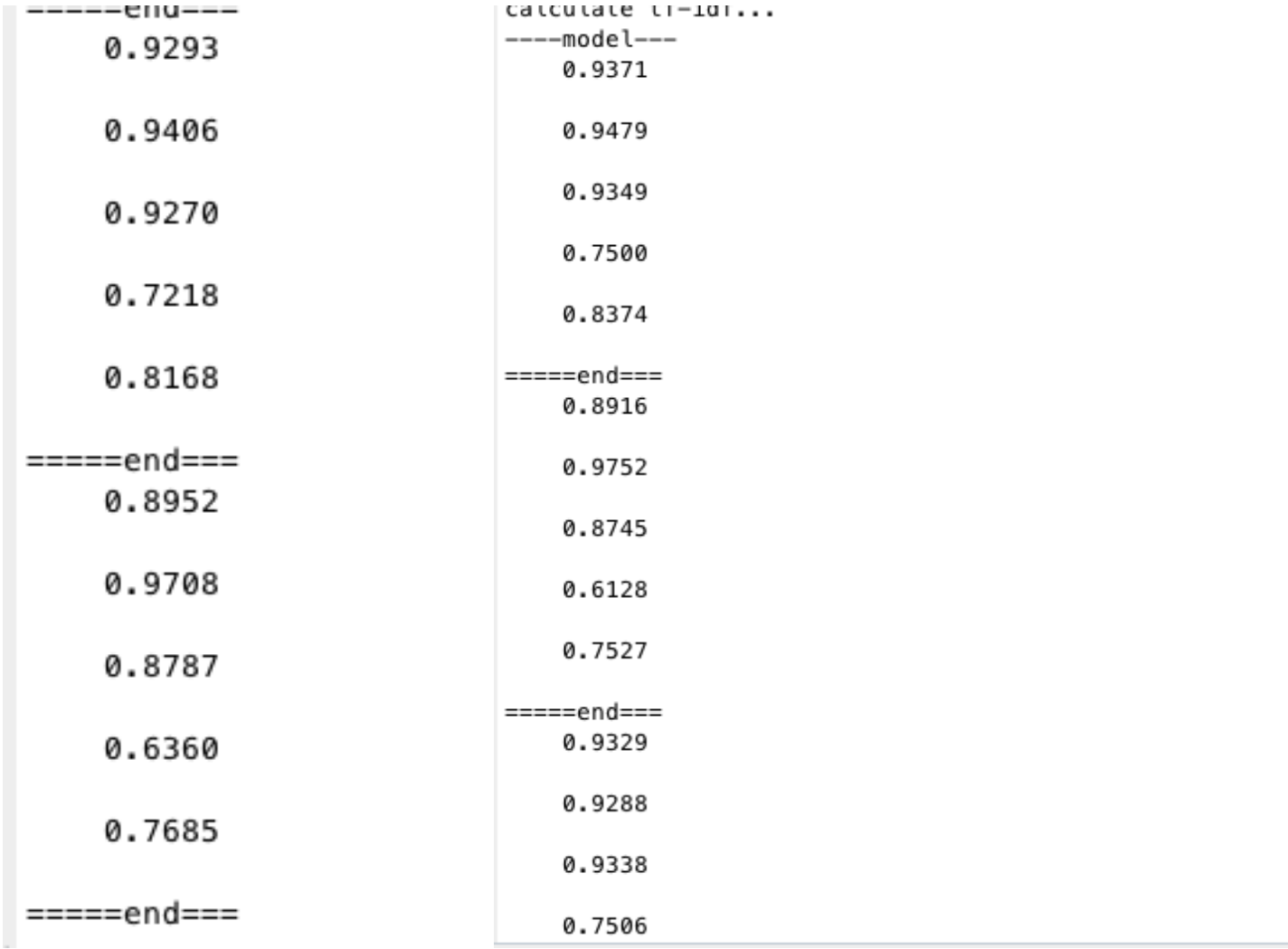

figure 3.1

Accuracy:

Testing set accuracy : **89%~94%, average level: 91.33%**

Sensitivity:

Testing set sensitivity: **92%~98%, average level: 94.26%**

Specificity:

Testing set specificity : **87%~94%, average level: 90.37%**

Precision:

Testing set precision: **63%~94%, average level:71.67%**

F-measure:

Testing set precision: **73%~87%, average level:82.78%**

CONCLUSION

- TP, FP, TN, FN provide the relevant information
- No single measure tells the whole story
- A classifier with 90% accuracy can be useless if 90% of the population does not have cancer and the 10% that do are misclassified by the classifier
- Use of multiple measures recommended
- Beware of terminological confusion in the literature!
- **The Whole Program will run about 7mins.(Faster and accuracy)**

BIBLIOGRAPHY:

[1] Naive Bayes by toshiakit (source: https://github.com/toshiakit/NaiveBayes)
[2] Reuters data (source: https://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection)
[3] Definition of TF-IDF (source: https://en.wikipedia.org/wiki/Tf%E2%80%93idf)
[4] Definition of cross-validation(source: https://en.wikipedia.org/wiki/Cross-validation_(statistics))