Knowledge-Based Systems 100 (2016) 137–144

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Two feature weighting approaches for naive Bayes text classifiers

Lungan Zhang [a], Liangxiao Jiang [a,b,*], Chaoqun Li [c,**], Ganggang Kong [a]

[a] Department of Computer Science, China University of Geosciences, Wuhan 430074, China
[b] Hubei Key Laboratory of Intelligent Geo-Information Processing, China University of Geosciences, Wuhan 430074, China
[c] Department of Mathematics, China University of Geosciences, Wuhan 430074, China

ARTICLE INFO

Article history:
Received 23 November 2015
Revised 22 February 2016
Accepted 23 February 2016
Available online 2 March 2016

Keywords:
Naive Bayes text classifiers
Feature weighting
Gain ratio
Decision tree

ABSTRACT

This paper works on feature weighting approaches for naive Bayes text classifiers. Almost all existing feature weighting approaches for naive Bayes text classifiers have some defects: limited improvement to classification performance of naive Bayes text classifiers or sacrificing the simplicity and execution time of the final models. In fact, feature weighting is not new for machine learning community, and many researchers have made fruitful efforts in the field of feature weighting. This paper reviews some simple and efficient feature weighting approaches designed for standard naive Bayes classifiers, and adapts them for naive Bayes text classifiers. As a result, this paper proposes two adaptive feature weighting approaches for naive Bayes text classifiers. Experimental results based on benchmark and real-world data show that, compared to their competitors, our feature weighting approaches show higher classification accuracy, yet at the same time maintain the simplicity and lower execution time of the final models.

1. Introduction

In recent years, the exponential growth of text documents on the Internet, digital libraries and other fields [26,32] has attracted the attention of many scholars. The task of automatic text classification is to assign text documents to pre-specified classes, which has been an important task in information retrieval [18,31]. Text classification presents unique challenges due to a large number of features, a large number of documents and strong dependencies among features [8,9].

To tackle text classification tasks, documents are characterized by the words that appear in them. Thus, one simplest way to apply machine learning to text classification is to treat each word as a Boolean variable. This is the first statistical language model called multi-variate Bernoulli naive Bayes (BNB) model [20]. BNB assumes that a document is represented by a vector of binary feature variables. The vector indicates which words occur or not in the document, and ignores the information of the number of times a word occurs in the document. To overcome this shortcoming confronting BNB, the multinomial naive Bayes (MNB) model [19] is proposed by capturing the information of the number of times a word occurs in a document. However, one systemic problem confronting MNB is that when one class has more training documents than others, MNB selects poor weights for the decision boundary. This is due to an under-studied bias effect that shrinks weights for classes with few training documents. To balance the amount of training documents used per estimate and deal with skewed training data, a complement class version of MNB called complement naive Bayes (CNB) is proposed [21]. The one-versus-all-but-one model (commonly misnamed one-versus-all, simply denoted by OVA) is a combination of MNB and CNB [21]. It is proved that OVA performs much better than MNB. Rennie et al. [21] attributed the improvement with OVA to the use of complement weights.

Although these naive Bayes text classifiers have already demonstrated remarkable classification accuracy, like naive Bayes classifiers, their conditional independence assumption is rarely true in reality. So, it is natural to improve naive Bayes text classifiers by relaxing the conditional independence assumption required by them. There are some approaches to do it such as structure extension [14], local learning [11,23], instance weighting [5,13], feature selection [2,10,27,30], and feature weighting [4,12,16,22], and so on.

This paper focuses on feature weighting approaches for naive Bayes text classifiers. To our knowledge, there exist some feature weighting approaches especially designed for naive Bayes text classifiers [16,22]. However, almost all of these existing approaches have some defects. The χ^2 statistic-based feature weighting approach [16] runs fast but the improvement to classification performance of naive Bayes text classifiers is limited. The CFS-based feature weighting algorithm [22] shows good classification accuracy but suffers from relative high execution time. So this paper tries to

* Corresponding author at: Department of Computer Science, China University of Geosciences, Wuhan 430074, China. Tel.: +86 2767883716.
** Corresponding author.
E-mail addresses: ljiang@cug.edu.cn (L. Jiang), chqli@cug.edu.cn (C. Li).

http://dx.doi.org/10.1016/j.knosys.2016.02.017
0950-7051/

propose some feature weighting approaches which have good classification performance and simultaneously maintain the simplicity and low execution time of the final models.

Feature weighting is not new for machine learning community. Many feature weighting algorithms have been especially designed for standard naive Bayes classifiers. We hope to borrow from previous research achievements about feature weighting of standard naive Bayes classifiers to improve naive Bayes text classifiers. For this purpose, this paper reviews some feature weighting algorithms especially designed for standard naive Bayes classifiers and finds some simple and efficient algorithms. But directly applying them to naive Bayes text classifiers cannot get good results, and some actual problems have to be solved. We adapt these algorithms for improving naive Bayes text classifiers. As a result, this paper proposes two adaptive feature weighting approaches for naive Bayes text classifiers. Compared to their competitors, our feature weighting approaches show higher classification accuracy, yet at the same time maintain the simplicity and lower execution time of the final models.

The remainder of this paper is organized as follows. Section 2 reviews the related work with regard to this paper. Section 3 proposes two adaptive feature weighting approaches for naive Bayes text classifiers. Section 4 describes in detail the experimental setup and results. The last section draws conclusions and outlines main directions for our future work.

2. Related work

Given a test document d, represented by a word vector $< w_1, w_2, \ldots, w_m >$, MNB, CNB, and OVA classify d using Eqs. (1)–(3), respectively.

$$c(d) = \arg\max_{c\in C}[logP(c) + \sum_{i=1}^{m} f_i logP(w_i|c)] \tag{1}$$

$$c(d) = \arg\max_{c\in C}[-logP(\bar{c}) - \sum_{i=1}^{m} f_i logP(w_i|\bar{c})] \tag{2}$$

$$c(d) = \arg\max_{c\in C}[(logP(c) - logP(\bar{c})) + \sum_{i=1}^{m} f_i(logP(w_i|c) - logP(w_i|\bar{c}))] \tag{3}$$

where C is the set of all class labels, $\bar{c}$ is the complement classes of the class c (all classes except the class c), m is the vocabulary size in the text collection (the number of different words in all of the documents), $w_i (i = 1, 2, \ldots, m)$ is the ith word occurs in the document d, f_i is the frequency count of the word w_i in the document d. The prior probabilities $P(c)$ and $P(\bar{c})$ are generally estimated by Eqs. (4) and (5), respectively, and the conditional probabilities $P(w_i|c)$ and $P(w_i|\bar{c})$ are generally estimated by Eqs. (6) and (7), respectively.

$$P(c) = \frac{\sum_{j=1}^{n} \delta(c_j, c) + 1}{n + l} \tag{4}$$

$$P(\bar{c}) = \frac{\sum_{j=1}^{n} \delta(c_j, \bar{c}) + 1}{n + l} \tag{5}$$

$$P(w_i|c) = \frac{\sum_{j=1}^{n} f_{ji}\delta(c_j, c) + 1}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ji}\delta(c_j, c) + m} \tag{6}$$

$$P(w_i|\bar{c}) = \frac{\sum_{j=1}^{n} f_{ji}\delta(c_j, \bar{c}) + 1}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ji}\delta(c_j, \bar{c}) + m} \tag{7}$$

where n is the number of training documents, l is the number of classes, c_j is the class label of the jth training document, f_{ji} is the frequency count of the word w_i in the jth training document, $\delta(c_j, c)$ and $\delta(c_j, \bar{c})$ are two binary functions, which can be defined as:

$$\delta(c_j, c) = \begin{cases} 1, & if\ c_j = c \\ 0, & otherwise \end{cases} \tag{8}$$

$$\delta(c_j, \bar{c}) = \begin{cases} 1, & if\ c_j \in \bar{c}, namely\ c_j \neq c \\ 0, & otherwise \end{cases} \tag{9}$$

Of numerous approaches to improve above naive Bayes text classifiers by relaxing their conditional independence assumption, feature weighting has received some attention from researchers. The resulting improved models classify d using Eqs. (10)–(12), respectively.

$$c(d) = \arg\max_{c\in C}[logP(c) + \sum_{i=1}^{m} W_i f_i logP(w_i|c)] \tag{10}$$

$$c(d) = \arg\max_{c\in C}[-logP(\bar{c}) - \sum_{i=1}^{m} W_i f_i logP(w_i|\bar{c})] \tag{11}$$

$$c(d) = \arg\max_{c\in C}[(logP(c) - logP(\bar{c})) + \sum_{i=1}^{m} W_i f_i(logP(w_i|c) - logP(w_i|\bar{c}))] \tag{12}$$

where W_i is the weight of the word w_i.

Obviously, how to learn each feature's weight $W_i (i = 1, 2, \ldots, m)$ is crucial in improving naive Bayes text classifiers by feature weighting. In order to learn the weights of features (words), [16] proposed a χ^2 statistic-based feature weighting approach, simply denoted by $R_{w,c}$. The weighted naive Bayes classifier using $R_{w,c}$ improves the text classification performance of basic naive Bayes classifier by measuring positive term-class dependency accurately at the training phase. Note that this feature weighting approach is originally proposed to improve standard naive Bayes for text classification, and thus the improvement to above naive Bayes text classifiers, including MNB, CNB, and OVA, is proved to be very limited [22]. To scale up the classification performance of above naive Bayes text classifiers, Wang et al. [22] proposed a CFS-based feature weighting approach, which firstly conducts a correlation-based feature selection (CFS) [6] process to select a best feature subset from the whole feature space and then assigns larger weights to the features in the selected feature subset and smaller weights to others. Their experimental results on a large suite of benchmark datasets show that the CFS-based feature weighting approach can dramatically improve the classification accuracy of above naive Bayes text classifiers. However, this feature weighting approach needs to employ a best first heuristic search to find the best feature subset, which incurs an approximately quadratic time complexity and affects its application in high-dimensional text data classification tasks.

Although there are not many feature weighting approaches especially designed for naive Bayes text classifiers, previous works have presented many feature weighting algorithms for standard naive Bayes classifier. Zhang and Sheng [29] proposed a gain ratio-based feature weighting approach for standard naive Bayes, in which a feature with higher gain ratio is assigned higher weight. Hall [7] proposed a decision tree-based feature weighting approach for standard naive Bayes. The decision tree-based feature weighting approach weights predictive features according to the degree to which they depend on other features' values and assigns lower weights to those features that have many dependencies. To estimate the degree to which a feature depends on others, an unpruned decision tree is built from a training data and the minimum depth d at which the feature is tested in the built tree is