A Framework for Titled Document Categorization with Modified Multinomial Naivebayes Classifier

Hang Guo and Lizhu Zhou

Computer Science & Technology Department
100084, Tsinghua University,Beijing,China
guohang@mails.tsinghua.edu.cn,
dcszlz@mail.tsinghua.edu.cn

Abstract. Titled Documents (TD) are short text documents that are segmented into two parts: Heading Part and Excerpt Part. With the development of the Internet, TDs are widely used as papers, news, messages, etc. In this paper we discuss the problem of automatic TDs categorization. Unlike traditional text documents, TDs have short headings which have less useless words comparing to their excerpts. Though headings are usually short, their words are more important than other words. Based on this observation we propose a titled document classification framework using the widely used MNB classifier. This framework puts higher weight on the heading words at the cost of some excerpt words. By this means heading words play more important roles in classification than the traditional method. According to our experiments on four datasets that cover three types of documents, the performance of the classifier is improved by our approach.

1 Introduction

TDs (Titled Documents) are short text documents composed by Heading Part and Excerpt Part. The former is the title or heading of the document. TDs are used in many applications such as papers, news, messages and so on. Their numbers are increasing very fast these years and they are getting more and more important. TDs are widely used in the Web. People are reading on-line news at home, searching for papers at college and posting messages to the Newsgroups. Now TDs have been important Internet resources for business communities. They are also valuable for data mining researchers.

The Multinomial Naivebayes [4](MNB) classifier is widely used in text categorization because it is fast and easy to implement. [6] shows that its performance is competitive with the state-of-the-art models like SVM [9] with simple modifications. MNB model follows the assumption that every word in the documents is independent with each other. [7] discusses why it performs well on such a severe assumption.

Traditional text classification models can be used in titled document classification. These classifiers are built on the bag-of-words model. They do not

R. Alhajj et al. (Eds.): ADMA 2007, LNAI 4632, pp. 335–344, 2007.

distinguish heading and excerpt words. These classifiers are developed for plain text documents, not for TDs. Usually the headings are more important. They are shorter but more important than other words. As shown in Section 2, the headings have much less useless words than the excerpts. In our previous work [11], we found that putting different weights on the words of different sections could improve classification performance of the classifiers. Based on this observation we consider the possibility to replace some excerpt words with heading words so that heading words can be "emphasized" in classification. Following this idea, we propose a titled document classification framework. In the training phase, we remove less words from the heading vocabulary than the excerpt vocabulary. In the classifying phase, the weights of heading words are increased. By this means more heading words are used in classification at the cost of some excerpt words. According to the experiments on four datasets, the performance of the classifiers is improved.

The reminder of the paper is organized as follows: in Section 2 we show that why heading words should be emphasized. Section 3 introduces our feature reduction approach. The classification framework is shown in Section 4. In Section 5 we present our experiments on four datasets. Section 6 glances at the related work. Finally the paper is concluded in Section 7.

2 Motivation

Intuitively heading words are more important than other words. The heading (title) words of a document are usually the keywords. Heading words should play more important roles than others. To test this idea, we select four real life text collections. They are: OHSUMED[1], Reuters-21578[2], 20-Newsgroups[3] and CiteSeer papers[4]. Every document in these collections has a short title(headline) and a longer abstract(excerpt). The four datasets cover three types of documents: papers, news and messages. We use Information Gain[5] – a popular dimensionality reduction method, to remove the less important words in documents. According to [5], the words left after reduction are considered to be more informative for classifiers. Before reduction there are about 3,000 words in titles and excerpts. Then we reduce the vocabulary size to 2000, 1000, 500, 200 and 100 words. Each time we compare the words left in excerpts and headings. The results are shown in Table 1.

The result shows that the number of heading words are much less than that of the excerpt words before feature reduction. However, the heading words are much more important than excerpt words in terms of the rate of informative words. Therefore heading words should play more important roles than others in training and classification.

[1] http://trec.nist.gov/data/t9_filtering/README

[2] http://www.ics.uci.edu/ kdd/databases/reuters21578/reuters21578.html

[3] http://people.csail.mit.edu/ jrennie/20Newsgroups/

[4] http://citeseer.ist.psu.edu/directory.html

Table 1. Comparison between Title Words and Other Words

Word Number		3000	2000	1000	500	300	100
OHSUMED	Excerpt	83.46	66.63	36.95	26.11	17.98	6.46
	Heading	11.43	8.37	5.87	4.58	3.74	1.99
	Excerpt/Heading	8.49	7.96	6.29	5.69	4.81	3.25
Reuters-21578	Excerpt	51.90	43.57	36.79	30.85	26.20	17.94
	Heading	6.11	5.17	4.68	4.26	3.81	2.76
	Excerpt/Heading	7.24	8.42	7.87	7.24	6.89	6.5
20-Newsgroup	Excerpt	87.79	54.11	27.87	18.52	13.26	4.91
	Heading	4.70	3.48	2.52	2.07	1.80	1.29
	Excerpt/Heading	18.68	15.55	11.05	8.95	7.36	3.81
CiteSeer	Excerpt	54.03	29.22	14.37	8.35	6.22	3.53
	Heading	7.24	4.49	2.79	2.35	2.19	1.70
	Excerpt/Heading	7.47	6.51	5.15	3.43	2.84	2.08

3 Feature Reduction

Before building the classifier, the size of the vocabulary must be reduced because the training documents usually have over 10,000 words. It is a great challenge for training a feasible classifier. Traditional feature reduction methods do not distinguish heading words and excerpt words in documents. We have shown in Table 1 that heading words are usually more informative than excerpt words. Therefore we should be more careful when a heading word is discarded.

Though heading words are less likely to be removed in feature reduction, sometimes they are incorrectly removed because they are mixed with other words. For instance, if the word “database” appear in the title of a document, this document is probably about database technology. Whereas “database” may also appear in the excerpts of articles on other categories since databases are widely used in different applications like machine learning, information retrieval, etc.. As a result, word “database” is likely to be removed according to Information Gain algorithm.

The solution to the problem is to separate heading words and excerpt words in feature reduction. Therefore we need to build two vocabularies: heading vocabulary and excerpt vocabulary. In feature reduction, we employ Information Gain algorithm on the two vocabularies independently. At last they are united as one vocabulary. The “compress rates” of the two vocabularies are different in feature reduction. Since heading words are more important, we remove less words from the heading vocabulary.

According to our feature reduction method, many less informative heading words are kept. In the classification phrase, the average length of the headings of the testing documents are increased. We make the heading words more “affective” in classification so that they can take the place of the removed excerpt words.

4 Classifying

As mentioned in Section 3, the idea of the titled document classifying method is to increase the "affects" of heading words in classification. In our previous work [11], we have found that putting higher weight on the more important words would increase the performance of the classifiers. According to the most popular weighting function– TFIDF, doubling the weight of the title words is doubling the occurrences of these words. Following this idea, we assume that

Assumption. When classifying a titled document $i = (h, e)$, it is equal to classify a plain text document $(\theta * h, e)$.
where h denotes the heading words, e denotes the excerpt words. $\theta(\theta > 1)$ is a prior.

In this way we can transform a titled document into a plain text document. Then we can use the well-developed plain text classification technology to classify titled document. Substantial efforts have been made in the literature to develop good plain text classification models, which are general enough to be applied to diverse document representations. In this study, we take one of the most popular classification model–Multinomial NaiveBayes (MNB) [4], as a case in point.

4.1 Traditional MNB Model

Multinomial Naivebayes is a widely used document classification model, whose performance is acceptable for many corpora [4]. It follows the NaiveBayes Independence Assumption that "*the probability of each word event in a document is independent of the word's context and position in the document*".

Suppose document d has words $\{w_1, w_2, \ldots, w_{|W|}\}$, the assumption is :

$$p(d|c) = \prod_{i=1}^{|W|} p(w_i|c) \tag{1}$$

here W denotes the vocabulary. According to the Bayes rule, we have

$$\begin{aligned} p(c|d) = p(c) * p(d|c)/p(d) &= p(c) * (\prod_{i=1}^{|W|} p(w_i|c))/p(d) \\ &\approx p(c) * (\prod_{i=1}^{|W|} \frac{n(w_i, c)}{n(c)})/p(d) \end{aligned}$$

where $n(w_i, c)$ is the number of occurrences of word w_i $(w_i \in W)$ in the training examples labeled with category c, and $n(c)$ is the total number of occurrences of all the words in W in the training examples associated with category c.

The goal of the classification problem is to find class j that maximizes $p(c_j|d)$. The complexity of the training procedure is $O(|D_{train}| * |W|)$ and the complexity of the classification is $O(|W| * |\mathcal{C}|)$, where $|D_{train}|$ is the total number of training examples and $|\mathcal{C}|$ is the number of categories(classes).

4.2 Our Algorithm

Following the NaiveBayes Independence Assumption in Formula 1 , we can extend the scope of MNB model for titled document classification. When classifying document $(\theta * h, e)$, here is:

$$\begin{aligned} p(c_i|\theta * h, e) &= \frac{p(\theta * h, e|c_i) * p(c_i)}{p(\theta * h, e)} \\ &= \frac{p(\theta * h|c_i) * p(e|c_i) * p(c_i)}{p(\theta * h, e)} \\ &= \frac{p(h|c_i)^\theta * p(e|c_i) * p(c_i)}{p(\theta * h, e)} \\ &= \frac{(p(c_i|h) * p(h))^\theta * p(c_i|e) * p(e)}{p(c_i)^\theta * p(\theta * h, e)} \\ &\propto \frac{p(c_i|h)^\theta * p(c_i|e)}{p(c_i)^\theta} \end{aligned} \quad (2)$$

Using Formula 2, we can easily calculate the classification result of titled document $(\theta * h, e)$ by the classification results of its heading and excerpt. When classifying document $(\theta * h, e)$, we first classify its heading and excerpt respectively. Then we have $p(c|e)$ and $p(c|h)$. At last we combine the results according to Formula 2.

Because $\theta > 1$, $p(c|h)$ is more affective than $p(c|e)$ for the classification result. However, in most cases it is hard to calculate $p(c|h)$ because headings are too short. If the approximation of $p(c|h)$ is far from the real possibility, the classification results of $p(c|\theta * h, e)$ will be incorrect.

The idea is to weight $p(c|h)$ by its classification error rate. We propose a heuristic weighting function to evaluate the classification result with different θ. The weight function of $p(c|\theta * h, e)$ is:

$$w_\theta = \begin{cases} 10^{-10*(e_h/e_e)-(\theta-1)} & : \quad e_h < 2 * e_e, \theta > 1 \\ 0 & : \quad e_h \geq 2 * e_e, \theta > 1 \\ 1 & : \quad \theta = 1 \end{cases} \quad (3)$$

here e_h is the error rate of $p(c|h)$ and e_e is the error rate of $p(c|e)$. In this paper we classify the training set to get an optimistic approximation of e_h and e_e. If e_h is too high, the weight of $p(c|\theta * h, e)$ will be dropped. According to Formula 2, the errors of $p(c|h)$ greatly affect the predication of $p(c|\theta * h, e)$ with the increase of θ. Then when θ increases, the weight of $p(c|\theta * h, e)$ will decrease.

We set an upper bound of θ, namely θ_{max} (set to 3), and combine all the classification results using different θ. The classifying algorithm is shown in Figure 1. The classification cost is $O(|\mathcal{C}| * \theta_{max})$.

```
Input: a titled document $(t, b)$, $\theta_{max}$
Output: classification result vector v

(a) Preprocessing
    Classify the training set to get $e_e$ and $e_h$.
    Calculate weight $w_1, \ldots, w_{\theta_{max}}$ by formula 3;

(b) Calculate $p(c_i|\theta * h, e)$;
    foreach $1 \leq i \leq |\mathcal{C}|$
        Calculate $P(c_i)$, $P(c_i|h)$ and $P(c_i|e)$;
        foreach $1 \leq \theta \leq \theta_{max}$
            Calculate $p(c_i|\theta * h, e)$ by formula 2;

(c) Combine Results
            $v_i = v_i + p(c_i|\theta * h, e) * w_\theta$;
        endfor
    endfor
```

Fig. 1. Pseudo Codes for classifying algorithm

5 Experiment

5.1 Datasets

To test our framework on different type of titled documents, we collect four real life datasets from a variety of applications: scientific papers search engines, the on-line news agency, Newsgroups, etc. CiteSeer and OHSUMED are widely-used paper collections. The abstracts of papers are used as Excerpt Parts. Reuters-21578 and 20-Newsgroup are often used as benchmarks in text categorization problems. We use the body section of news or messages as their Excerpt Part. Due to the size of these collections, a small portion of data are selected and used in our experiments.

CiteSeer. CiteSeer[5] provides on-line scientific materials on computer science. We use its classified papers[6] as our training and testing set. There are 17 top categories in its hierarchy. To avoid possible overlap among those categories, we only use 8 categories in our experiments. They are Database, Agents, Compression, Hardware, Networking, Programming, Security, Software Engineering and Theory.

[5] http://citeseer.ist.psu.edu/

[6] http://citeseer.ist.psu.edu/directory.html

OHSUMED. OHSUMED[7] document collection is a set of 348,566 references from MEDLINE, the on-line medical information database, consisting of titles and/or abstracts from 270 medical journals over a five-year period (1987-1991). These papers are categorized into 4904 topics. Two of the largest 10 categories are randomly selected for our experiments.

Reuters-21578. Reuters-21578 [8] is a collection of documents that appeared on Reuters newswire in 1987. The documents were assembled and indexed with categories by personnel from Reuters Ltd and and Carnegie Group, Inc.. We use ModApte Split in our experiments.

20-Newsgroup. 20-Newsgroup[9] is a collection of 20,000 messages, collected from 20 different netnews newsgroups. One thousand messages from each of the twenty newsgroups were chosen at random and partitioned by newsgroup name. We chose four newsgroups from the collection because there are overlaps between different categories. The selected newsgroups are "alt.atheism", "talk.politics.guns", "comp.os.ms-windows.misc" and "rec.sport.hockey".

5.2 Training and Testing

Environment and Library. We perform the experiments on a 1.5GHz workstation with 512M memory. We select Naivebayes[4] classifier in our experiment. It is implemented by Weka[10] and Judge[11]. They are both open source classification toolkits in Java.

Training. We use the popular Information Gain algorithm [5] to reduce the feature space. In the excerpt vocabulary we select the top 50% words out of 4,000 words. And in the heading vocabulary we select the top 30% words out of 2,000 words. Then the two reduced vocabularies are united as the training vocabulary. Based on this vocabulary we build a MNB classifier.

Classifying. We compare the results of our framework with the results of traditional plain text classification method. The results are listed in Table 2. We use the overall error rate as the criteria in our experiments. The error rates are affected by θ_{max}.

5.3 Discussion

Table 2 shows that the titled document classification framework declines the classification error rate. Generally speaking, we get more improvement in CiteSeer and OHSUMED. That is probably because the titles of papers are longer and

[7] http://trec.nist.gov/data/t9_filtering/README

[8] http://www.ics.uci.edu/ kdd/databases/reuters21578/reuters21578.html

[9] http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/news20.html

[10] http://www.cs.waikato.ac.nz/ml/weka/

[11] http://www3.dfki.uni-kl.de/judge/

Table 2. Classification Error Rates

θ_{max}	CiteSeer	OHSUMED	Reuters	20-Newsgroup
2	24.7%	5.2%	3.3%	5.5%
3	24.9%	5.4%	3.4%	5.8%
4	25.1%	5.6%	3.5%	6.1%
tradition	27.0%	6.0%	3.4%	5.8%

more formal. Researchers usually choose titles carefully. Therefore title words are usually the keywords of the whole documents. They deserve to be emphasized. However the headlines of news and messages are relatively shorter and more informal. Many news headlines use abbreviates and numbers such as "MGM/UA COMMUNICATIONS 2ND QTR FEB 28 LOSS". These headlines are meaningless for our classifier. Things are even worse in 20-Newsgroups. Newsgroup users always use titles like "I agree", "You got it" and so on. These titles are actually meaningless words that have little use in classification. Our framework keeps these title words at the cost of some excerpt words. If many title words are useless in classification, our method fails.

The performance of the classifier declines with the increase of θ_{max}, especially in 20-Newsgroups and Reuters. If θ_{max} is too large, the error rate increases. That is because $p^{\theta}(c|h)$ are used in Formula 2. It is clear that these possibilities given by MNB is not the real distributions. When θ goes larger, we have more errors. To get the optimized θ_{max} on a given corpus, it is better to run the algorithm on a small portion of documents first. For those collections that documents have long and meaningful headings (titles), we can set larger θ_{max}.

6 Related Work

Conventional automated documents categorization models have been developed for years. [5] has introduced many frequently used models, Naivebayes[4], SVM[9][10], Decision Tree, etc. SVM performs best in many datasets. However, even with the linear kernel, its training and classifying costs are higher than simple models like NB. Moreover, other simple models still can be improved in terms of accuracy. [7] shows that with simple modification, the performance of Naive Bayes Multinomial model approaches SVM.

The use of titles in document classification is first explored in [12]. Recently many experiments have shown that titles are very useful when classifying various documents. [8] uses titles and other features in Web page classification with support vector machine. [15] improves the performance of classifiers by combining the main body text, anchor text and titles when classifying Web pages. [13] proposes an intelligent document title classification method based on information theory. [14] proves that weighting titles in life science publications improves the performance of classifiers.

7 Conclusion and Future Work

Titled documents such as papers, news and messages are widely used these days. The headings words of these documents are usually more important than other words. Traditional document classification methods usually ignore the differences of heading words and other words. In this paper we propose a titled document classification framework. According to this framework, we remove less heading words in feature reduction at the cost of some excerpt words. In classification all the heading words are put more weight. By this means heading words play more important roles in classification than the traditional method. According to the experiments on four real life datasets, the error rates are dropped by our method.

We are working on new methods to determine θ_{max}. In the next step we will develop a new framework based on the state-of-art classifiers like SVM.

Acknowledgement

This work is supported by National Nature Science Foundation of China under Grant No.60520130299.

References

1. Hull, D.P., Schutze, J., Method, H.: Combination for document filtering. In: Proc. the 19th ACM SIGIR Conference on Research and Development in Information Retrieval, Switzerland, pp. 279–287 (1996)
2. Tumer, K., Ghosh, J.: Linear and order statistics combination for pattern classification. In: Sharkey, A. (Ed.) Combining Artificial Neural Networks, pp. 127–162. Springer-Verlag (1999)
3. Merz, C.J., Pazzani, M.J.: Combining neural network regression estimates with regularized linear weights. In: Advances in Neural Information Processing Systems, vol. 9, pp. 564–570. MIT Press, Cambridge (1997)
4. Mccallum, A., Nigam, K.: A Comparison of Event Models for Naive Bayes Text Classification. In: Proc. AAAI workshop on Learning for Text Categorization, Wisconsin, pp. 41–48 (1998)
5. Fabrizio, S.: Machine Learning in Automated Text Categorization. ACM Computing Surveys 34, 1–47 (2002)
6. Rennie, J.D.M., et al.: Tackling the Poor Assumptions of Naive Bayes Text Classifiers. In: Proc. International Conference on Machine Learning, Washington, DC (2003)
7. Domingos, P., Pazzani, M.: Beyond independence: conditions for the optimality of the simple Bayesian classifier. In: Proc. International Conference on Machine Learning, Italy (1996)
8. Sun, A., Lim, E., Ng, W.: Web Classification Using Support Vector Machine. In: Proc. Workshop on Web Information and Knowledge Management, Virginia (2002)
9. Joachims, T., Sebastiani, F.: Guest editors's categorization. J. Intell. Inform. Syst. 18(2/3), 103–105 (2002)

10. Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In: Nédellec, C., Rouveirol, C. (eds.) Machine Learning: ECML-98. LNCS, vol. 1398, pp. 137–142. Springer, Heidelberg (1998)
11. Guo, H., Zhou, L.: Segmented Document Classification: Problem and Solution. In: Bressan, S., Küng, J., Wagner, R. (eds.) DEXA 2006. LNCS, vol. 4080, Springer, Heidelberg (2006)
12. Hamill, K., Zamora, A.: The use of titles for automatic document classification. In J. of the American Society for Information Science (1980)
13. Song, D., Bruza, P., Huang, Z., Lau, R.: Classifying Document Titles Based on Information Inference. In: Zhong, N., Raś, Z.W., Tsumoto, S., Suzuki, E. (eds.) ISMIS 2003. LNCS (LNAI), vol. 2871, Springer, Heidelberg (2003)
14. Hakenberg, J., Rutsch, J., Leser, U.: Tuning Text Classification for Hereditary Diseases with Section Weighting. In: Proc International Symposium on Semantic Mining in Biomedicine (2005)
15. Kaist, I., Kim, G.: Query type classification for web document retrieval. In: Proc. of ACM SIGIR, ACM Press, New York (2003)