Feature Selection on Chinese Text Classification Using Character N-Grams

Zhihua Wei[1,2], Duoqian Miao[1], Jean-Hugues Chauchat[2], and Caiming Zhong[1]

[1] Tongji University, Key laboratory "Embedded System and Service Computing" Ministry of Education, Shanghai, Cao'an Road, China 201804
[2] Université Lumière Lyon 2, Laboratoire ERIC, 5 avenue Pierre Mendès-France, 69676 Bron Cedex, France

Abstract. In this paper, we perform Chinese text classification using n-gram text representation on TanCorp which is a new large corpus special for Chinese text classification more than 14,000 texts divided into 12 classes. We use different n-gram feature (1-, 2-grams or 1-, 2-, 3-grams) to represent documents. Different feature weights (absolute text frequency, relative text frequency, absolute n-gram frequency and relative n-gram frequency) are compared. The sparseness of "document by feature" matrices is analyzed in various cases. We use the C-SVC classifier which is the SVM algorithm designed for the multi-classification task. We perform our experiments in the TANAGRA platform. We found out that the feature selection methods based on n-gram frequency (absolute or relative) always give better results and produce denser matrices.

Keywords: Chinese text classification, N-gram, Feature selection.

1 Introduction

In recent years, much attention has been given to the Chinese text classification (TC) with the rapidly increasing quantity of web sources and electronic texts in Chinese. The great difference between Chinese TC and Latin languages TC lies in the text representation. Unlike most of the western languages, the Chinese words do not have a remarkable boundary. This means that the word segmentation is necessary before any other preprocessing. The use of a dictionary is necessary. The word sense disambiguation issue and the unknown word recognition problem limit the precision of word segmentation [1]. This makes Chinese representation using words, phrases, meanings, and concepts more difficult.

In this paper, we use a method independent of languages which represents texts with character n-grams. A character n-gram is a sequence of n consecutive characters. The set of n-grams (usually, n is set to 1, 2, 3 or 4) that can be generated for a given document is basically the result of moving a window of n characters along the text. The window is moved one character at a time. Then, the number of occurrences of each n-gram is counted [2]. There are several advantages of using n-grams in TC tasks [3]. One of them is that by using n-grams, we do not need to perform word segmentation. In addition, no dictionary

G. Wang et al. (Eds.): RSKT 2008, LNAI 5009, pp. 500–507, 2008.

or language specific techniques are needed. However, n-gram extraction on a large corpus will yield a large number of possible n-grams, but only some of them will have significant frequency values in vectors representing the texts and good discriminate power. Our contribution is twofold: first we present how to choose the value of n in using n-gram to represent Chinese texts; and second the most suitable kind of feature weight is proposed.

Usually, there are two steps in the construction of an automated text classification system. The first one is that the texts are being preprocessed into a representation more suitable for the learning algorithm that is applied afterwards. The second step regards the learning algorithm that is chosen. In this work we focus on the first step. There are various ways of representing a text such as by using word fragments, words, phrases, meanings, and concepts [4]. Different text representations have different dependence on the language used in the text.

The reminder of this paper is organized as follows. Section 2 presents the text representation forms in our work. Section 3 gives our feature choosing strategies. Section 4 introduces the experiment dataset and the experiment scenarios. Section 5 analyzes the experimental results and section 6 concludes.

2 Text Representation Using N-Grams Frequencies

We adopt the VSM (Vector Space Model), where each document is considered to be a vector in feature space. Thus, given a set of N documents, d_1, $d_2 ... d_N$, the table of "document by feature" is constructed such as that shown in table 1, where each document is represented by a core "w_{ij}". Generally, w_{ij} has two kinds of value:

i) w_{ij} = frequency of feature j in document i;
ii) $w_{ij} = 0$ or 1, $w_{ij} = 1$, if feature j appears in document i, otherwise, w_{ij}=0.

In our work, we choose the first form. In table 1, F_i is n-gram. Chinese texts representation by using n-grams is concerned in some researches. [3] regards that 2-grams are best features for Chinese texts. [5] gives some experimental results by using n-gram combination. In their papers, the best result is by using 1-, 2-, 3-, 4-grams, the second best is by using 1-,2-grams, the third best is by using 2-grams, the case of using 2-, 3-, 4-grams follow and the worst one is by using 1-grams.

In Chinese language, the most part of words are made of one character (for example, some frequently used nous) or two characters. Some proper names or scientific terms have more characters [1]. It seems that the combination of 1-, 2-, 3-, 4-grams even 5-gram, 6-gram will produce a better result. However, we should not extract too many n-grams since it will produce a very large set of candidate features for a corpus which includes more than 14,000 documents. As a result, we choose the combination of 1-, 2-grams. We also do some experiments by using 1-, 2-, 3-grams in order to include some proper names or unknown words.

Table 1. "Document by feature" vector table

D	F_1	F_2	...	F_j	...	F_M	Class	Status
d_1	w_{11}	w_{12}	...			w_{1M}	A	Learning
d_2	w_{21}	w_{22}	...			w_{2M}	B	Learning
...		...					...	Learning
d_i	w_{i1}	w_{i2}	...	w_{ij}		w_{iM}	C	...
...			...				...	Testing
d_N	w_{N1}	w_{N2}	...			w_{NM}	A	Testing

3 Feature Selection

Feature selection is a term space reduction method which attempts to select the more discriminative features from preprocessed documents in order to improve classification quality and reduce computational complexity. As many n-grams are extracted from Chinese texts, we perform two steps of feature selection. The first is reducing the number of features inter-class. The second is choosing the more discriminate features among all the classes in training set.

3.1 Some Definitions

In text classification, the text is usually represented as a vector of weighted features. The difference between various in text representations comes from the definition of "feature". This work explores four kinds of feature building methods with their variations.

In the training set, each text in corpus D belongs to one class c_i. Here, $c_i \in C$, $C = \{c_1, c_2...c_i...c_n\}$, C is the class set defined before classification.

- Absolute text frequency is noted as $Text\_freq_{ij}$,which is the number of texts which include n-gram j in class c_i;
- Relative text frequency is noted as $Text\_freq\_relative_{ij}$,which is got from $Text\_freq_{ij}/N_i$, here, N_i is the quantity of texts in class c_i in training set;
- Absolute n-gram frequency is noted as $Gram\_freq_{ij}$, which is the number of n-gram j in all texts in class c_i in training set;
- Relative n-gram frequency is noted as $Gram\_freq\_relative_{ij}$, which is got from $Gram\_freq_{ij}/N'_i$, here, N'_i is the total of occurrence of all n-grams in all texts in class c_i in training set.

3.2 Inter-class Feature Number Reduction

We extract all the 1-, 2-grams or 1-, 2-, 3-grams in all the texts of the corpus and divide the corpus into training set and testing set. In our work, 70% texts in each class are selected by random to constitute the learning set and the rest 30% are used for the testing set. The following inter-class feature number reduction algorithm is performed only on training set.

Algorithm 1
Begin
For $c_i \in C$, $C = \{c_1, c_2 ... c_i ... c_n\}$,
$Term'_i = \emptyset$, $Term = \emptyset$;
For $n - gram_j \in Term_i$,
If $Text\_freq\_relative_{ij} > \alpha$, then $n - gram_j \in Term'_i$.
$Term = \{Term'_1, Term'_2 ... Term'_i ... Term'_n\}$.
End.

Here, $Term_i$ include all the n-grams extracted from the texts in class c_i , $Term'_i$ include all the n-grams selected in class c_i and $Term$ is n-gram set in all classes selected by algorithm 1. We choose $\alpha = 0.02$ as threshold in order to keep as many as possible features in each class. After this selection, there are 7000 features in each class in average which are enough for text classification task. In the case of $Text\_freq\_relative_{ij} < 0.03$, there are only 4,000 features left in each class in average. It is not enough for the further steps.

3.3 Cross-Class Feature Selection

We construct "feature by class" matrix (noted as $Matrix_{cf}$) by algorithm 2 to select discriminative features.

Algorithm 2
Begin
For $c_i \in C$, $C = \{c_1, c_2 ... c_i ... c_n\}$,
For $n - gram_j \in Term$,
If $n - gram_j \notin Term_i$, $\{O_{ij}\} = 0$
Else $\{O_{ij} = Text\_freq_{ij}$ or $Text\_freq\_relative_{ij}$ or
$Gram\_freq_{ij}$ or $Gram\_freq\_relative_{ij}\}$,
End.

In $Matrix_{cf}$, each feature "j" is assigned a numeric score based on its occurrence within the different document classes c_i. The choice of the scoring method in this work is the CHI-Square test. There are many other tests available as summarized in [6], but the CHI-Square is often cited as one of the best tests for the feature selection. It gives a similar result as Information Gain because it is numerically equivalent as shown by [7]. The score of n-gram "j" is:

$$\sum_i \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \tag{1}$$

Where "i" is the class, "j" is the n-gram and O_{ij} is the observed value. E_{ij} represent the expectation value in the hypothesis of independence of classes and features:

$$E_{ij} = \frac{O_{i+} * O_{+j}}{O_{++}} \tag{2}$$

Here, we define four kinds of values on O_{ij} (as described in algorithm 2) in different experiment scenarios. According to the result of CHI-Square, we

separately perform the classification using the 200, 500, 800, 1000, 2000... 5000 features.

4 Experiment

We adopt TanCorp-12 corpus, a collection of 14,150 texts in Chinese language, has been collected and processed by Songbo Tan [8]. It contains 12 categories (art, car, career, computer, economy, education, entertainment, estate, medical, region, science and sport). The biggest class contains 2865 texts (4.17M) and the smallest class contains 150 texts (0.49M).

In order to test the results given from different kinds of methods in feature selection, we set different experiment scenarios, as described in table 2. In the following section, we use a short name (e.g. Ex_1) and a long name (e.g. $1,2-gram\&ngram-re$) to describe each experiment scenario. The first part of the long name can be "1,2-gram" or "1,2,3-gram" and it notes the items extracted from texts as features. The second can be "text-re", "ngram-re", "ngram-ab" or "text-ab" notes the feature selection method cross-class.

Table 2. Experiment scenarios list

Experiment scenario	N-gram combination	Feature selection cross-class
Ex_1: 1,2-gram&ngram-re	1+2-gram	Relative n-gram frequency
Ex_2: 1,2-gram&text-re	1+2-gram	Relative text frequency
Ex_3: 1,2-gram&ngram-ab	1+2-gram	Absolute n-gram frequency
Ex_4: 1,2-gram&text-ab	1+2-gram	Absolute text frequency
Ex_5: 1,2,3-gram&ngram-ab	1+2+3-gram	Absolute n-gram frequency
Ex_6: 1,2,3-gram&text-re	1+2+3-gram	Relative text frequency

We use the C-SVC classifier which was introduced in LIBSVM [9]. It is the SVM algorithm designed for the multi-classification task. We use a linear kernel. Learning parameters are set to $gamma = 0$ and $penaltycost = 1$. We perform our experiments in the platform TANAGRA which is a free data mining software for academic and research purposes developed by Ricco Rakotomalala [10].

We use the F1 measure introduced by [11]. This measure combines recall and precision in the following way for bi-class case.

$$Recall = \frac{number\ of\ correct\ positive\ prediction}{number\ of\ positive\ examples} \tag{3}$$

$$Precision = \frac{number\ of\ correct\ positive\ prediction}{number\ of\ positive\ predictions} \tag{4}$$

$$F1 = \frac{2 * Recall * Precision}{Recall + Precision} \tag{5}$$

For more than 2 classes, the F1 scores are summarized over the different categories using the Micro-averages and Macro-averages of F1 scores.

1) Micro - F1 = average in documents and classes
2) Macro - F1 = average of within - category F1 values

5 Results and Discussions

5.1 Comparison of Macro-F1 and Micro-F1 in All Experiment Cases

Figure 1 show that Ex_3 and Ex_1 have the best performance, Ex_5 has the second best, Ex_2 and Ex_4 follow and the Ex_6 has the worst results. The first three best results are in the Ex_3, Ex_1 and Ex_5 which are all using n-gram frequency (relative or absolute) for feature selection. In the situation of absolute frequency (Ex_3) and relative frequency (Ex_1), the results are similar. The results indicate that using n-gram frequency for feature selection is better than using text frequency. Also the relative frequency does not give better results than the absolute frequency. We use 1-, 2-grams in Ex_1, Ex_2, Ex_3 and Ex_4 and use 1-, 2-, 3-grams in $Ex5$ and Ex_6. Fig.1 shows that the results produced by using 1-, 2-grams are little better than those produced by using 1-, 2-, 3-grams.

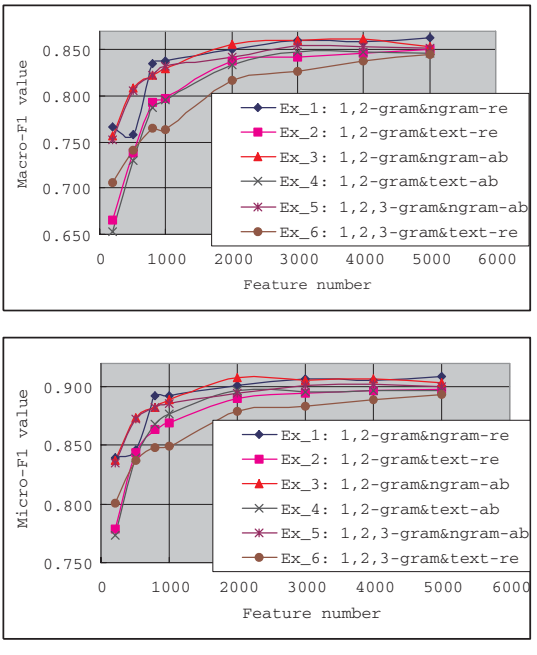

Fig. 1. Macro-F1 and Micro-F1 comparison on all experiment cases

Our experiments also indicates that the number of 2-grams and 3-grams increase with the increasing of feature number. In the case of more than 3000 features, the percentage of 1-grams, 2-grams and 3-grams do not change greatly.

Either in the case of using 1-, 2-, 3-grams or using 1-, 2-grams, 2-grams are always the most important features. There are 1108 3-grams in 5000 features and 206 3-grams are words. Most of them are new words, scientific terms, proper names, abbreviations and phrases which are very difficulties in Chinese word segmentation. In this regard, the method based on n-gram can solve the problem of unknown words recognition to some degree.

5.2 Sparseness Comparison

[12] shows that the computational time is more linked with the number of non-zero values in the cross-table (document by feature) than with its number of columns (features). Fig. 2 shows the non-zero value distribution in the "document by feature" matrix for six experiment cases. Ex_2 (1,2-gram&text-re) has about two times less non-zero cells than Ex_1 (1,2-gram&ngram-re), which indicates that it will produce less dense matrices after cross-class feature selection, so in this way the computation will be faster. Similarly, Ex_4 (1,2-gram&text-ab) has about two times less non-zero cells than Ex_3 (1,2-gram&ngram-ab). Ex_6 (1,2,3-gram&text-re) has two times less non-zero cells than Ex_5 (1,2,3-gram&ngram-ab). The matrices are denser when we use an absolute frequency than a relative frequency. For example, Ex_3 has more non-zero cells than Ex_1 and Ex_4 has more non-zero cells than Ex_2. The number of non-zero cells in the cases of using 1-, 2-, 3-grams is less than that of using 1-, 2-grams.

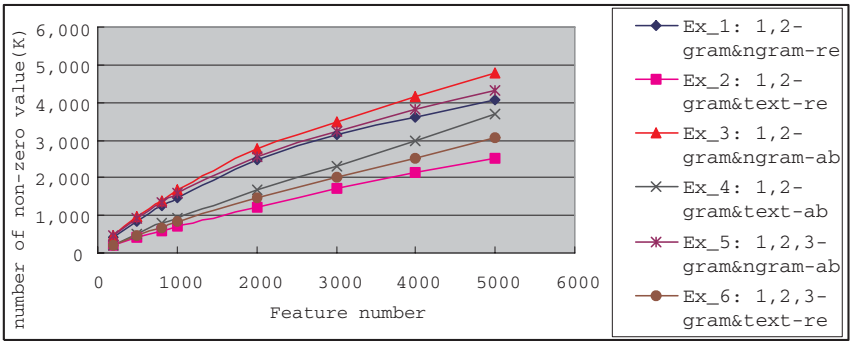

Fig. 2. Comparison of non-zero value in "text by feature" matrix on six cases

6 Conclusion

In this paper, we perform Chinese text categorization on a large corpus using n-gram text representation and different cross-class feature selection methods. Our experiments show that a combination of 1-, 2-grams is little better than that of 1-, 2-, 3-grams for Chinese text classification.

The feature selection methods based on n-gram frequency (absolute or relative) always give better results than those based on text frequency (absolute or

relative). Relative frequency is not better than the absolute frequency. Methods based on n-gram frequency also produce denser "document by feature" matrices. Our further work are exploring more excellent methods for feature selection using 1-, 2-grams in Chinese text classification, for example, the methods based on rough set.

Acknowledgments. This paper is sponsored by the National Natural Science Foundation of China (No. 60475019 and No. 60775036) and the Research Fund for the Doctoral Program of Higher Education of China (No. 20060247039). Our colleagues, Ruizhi WANG and Anna Stavrianou gave many good advices for this paper. We really appreciate their helps.

References

1. Miao, D.Q., Wei, Z.H.: Chinese Language Understanding Algorithms and Applications. Tsinghua University Press (2007)
2. Radwan, J., Chauchat, J.-H.: Pourquoi les n-grammes permettent de classer des textes? Recherche de mots-clefs pertinents l'aide des n-grammes caractèristiques. In: JADT 2002: 6es Journées internationales d'Analyse statistique des Données Textuelles, pp. 381–390 (2002)
3. Alain, L., Halleb, M., Delprat, B.: Recherche d'information et cartographie dans des corpus textuels à partir des fréquences de n-grammes. In: Mellet, S. (ed.) 4èmes Journées Internationales d'Analyse statistique des Données Textuelles, Université de Nice - Sophia Antipolis, pp. 391–400 (1998)
4. Joachims, T.: Learning to Classify Text Using Support Vector Machines. University Dortmund (February 2001)
5. Zhou, S.G., et al.: A Chinese Document Categorization System Without Dictionary Support and Segmentation Processing. Journal of Computer Research and Development 38(7), 839–844 (2001)
6. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys 34(1), 1–47 (2002)
7. Benzécri, J.-P., L'Analyse, D.: T1 = la Taxinomie. DUNOD, Paris (1973)
8. Tan, S.B., et al.: A novel refinement approach for text categorization. In: CIKM 2005, pp. 469–476 (2005)
9. Fan, R.-E., Chen, P.-H., Lin, C.-J.: Working set selection using second order information for training SVM. Journal of Machine Learning Research, 1889–1918 (2005)
10. Ricco, R.: TANAGRA: un logiciel gratuit pour l'enseignement et la recherché. In: EGC 2005, RNTI-E-32, pp. 697–702 (2005)
11. Van Rijsbergen, C.J.: Information Retrieval. Butterworths, London (1979)
12. Artur, Š, et al.: Detailed experiment with letter n-gram method on Croatian-English parallel corpus. In: EPIA 2007, Portuguese Conference on Artificial Intelligence (2007)