2012 10th IAPR International Workshop on Document Analysis Systems

Impact of Word Segmentation Errors on Automatic Chinese Text Classification

Xi Luo, Wataru Ohyama, Tetsushi Wakabayashi, Fumitaka Kimura
Graduate School of Engineering
Mie University
Tsu-shi, Mie, Japan
luoxi@hi.info.mie-u.ac.jp

***Abstract*—In this paper, several sets of experiments were carried out to study the impact of word segmentation errors on automatic Chinese text classification. Comparison experiment of four word-based approaches was first carried out and the results show that the performance was significantly reduced when using automatic word segmentation instead of manual word segmentation which means errors caused by automatic word segmentation have an obvious impact on classification performance. We further conducted the experiment using character-based approach (*N*-gram). Although *N*-gram approach produces a large number of ambiguous words, the results show that it performed better than automatic word segmentation.**

Keywords-Chinese text classification/categorization; word segmentation; ICTCLAS; N-gram; support vector machine

I. INTRODUCTION

Automatic text classification (ATC) is the task to automatically assign one or more appropriate categories for a document according to its content or topic [1]. Traditionally, text classification is carried out by human experts as it requires a certain level of vocabulary recognition and knowledge processing. With the rapid explosion of texts in digital form and growth of online information, text classification has become an important research area owing to the need to automatically handle and organize text collections.

Since there is no natural delimiter between Chinese words, this means that the Chinese segmentation is necessary before any other preprocessing. Numerous different segmentation approaches have been proposed for Chinese text classification. These approaches can be basically divided into word-based approach and character-based approach. For word-based approach, Chinese word segmentation is an important fundamental task and the quality of which has a direct impact on the performance of classification.

In this paper, several sets of experiments were carried out to study the impact of automatic word segmentation errors on Chinese text classification. We used ICTCLAS [4] to perform automatic word segmentation. Manual word segmentation was obtained from TanCorp-12 [2]. It was considered as an ideal word segmentation result set and was used to evaluate the accuracy of automatic word segmentation. The experimental results show that the performance was significantly reduced by 4.62% when using automatic word segmentation instead of manual word segmentation. It means errors caused by automatic word segmentation have an obvious impact on classification performance.

Furthermore, we performed Chinese text classification using character-based approach (*N*-gram) instead of word segmentation. By using *N*-grams, we do not have to perform word segmentation and no dictionary or language specific techniques are needed. Although *N*-gram approach produces a large number of ambiguous words, the results show that it performed better than automatic word segmentation. When ambiguous words were deleted, the performance was only slightly improved less than 0.4%, which was not as much as we expected.

The rest of this paper is organized as follows: In Section II we describe two basic approaches of Chinese segmentation. Section III describes the procedures of Chinese text classification and methodologies used. Experiments and results are presented in Section IV and Section V respectively. We summarize our research and point out some future direction in Section VI.

II. CHINESE SEGMENTATION

Word is the minimum meaningful unit of languages. Unlike English and other western languages, there is no natural delimiter between Chinese words and even no uniform smallest semantic units. Therefore, Chinese segmentation is necessary before any other preprocessing.

Numerous different segmentation approaches have been proposed for Chinese text classification. The basic approaches of Chinese segmentation can be roughly divided into two groups, character-based approach and word-based approach.

A. Word-based Approach

As we have discussed before, there are no delimiters to mark word boundaries and no explicit definitions of words in Chinese. For word-based approach, Chinese word segmentation is an important fundamental task and the quality of which has a direct impact on the performance of classification. The inherent errors caused by automatic word segmentation always remain as a problem.

1) Automatic Word Segmentation

For automatic word segmentation, lexical analysis is a prerequisite to Chinese text classification. In our work, we used ICTCLAS (Institute of Computing Technology, Chinese Lexical Analysis System) [4] to perform word segmentation. ICTCLAS is widely used in the field of

978-0-7695-4661-2/12 $26.00 © 2012 IEEE
DOI 10.1109/DAS.2012.43

Chinese word segmentation and ranked first with 97.58% in word segmentation precision in a recent official evaluation, which was held by the National 973 Fundamental Research Program of China.

2) Manual Word Segmentation

The manual word segmentation was obtained from TanCorp-12 [2] which is a new large corpus special for Chinese text classification. It was considered as an ideal word segmentation result set and was used to evaluate the accuracy of automatic word segmentation. Arabic number, English strings and punctuation marks were not treated as segmentation units and excluded from documents.

B. Character-based (N-gram) Approach

Character-based approach can be defined as purely mechanical processes that extract certain number of characters from documents.

In this paper, we use a method independent of languages which represents documents with character *N*-grams [3]. A character *N*-gram is a sequence of *N* consecutive characters. Sequences of one character (*N*=1) are called uni-gram (1-gram). Sequences of two characters (*N*=2) are called bi-gram (2-gram). Sequences of three characters (*N*=3) are called tri-gram (3-gram). Table I shows examples of *N*-gram features.

TABLE I. EXAMPLES OF *N*-GRAM FEATURES

Original Text	明天我们去北京
1-gram	明; 天; 我; 们; 去; 北; 京
2-gram	明天; 天我; 我们; 们去; 去北; 北京
3-gram	明天我; 天我们; 我们去; 们去北 ; 去北京

The use of *N*-gram feature instead of word segmentation in text classification tasks offers several advantages. One of them is that by using *N*-grams, we do not have to perform word segmentation. In addition, no dictionary or language specific techniques are needed and *N*-grams are also language independent.

III. PROCEDURE OF CHINESE TEXT CLASSIFICATION

Fig.1 shows the general procedure of Chinese text classification. In the following subsections, we introduce the methodologies used for the procedure in more detail.

A. Feature Vector Generation

In order for a machine learning system to recognize a document there should be a way of representing it. This is usually done by the use of feature vectors. First a lexicon including all different features in training data was generated. Then the feature vector represents the frequency of a specific feature in the document. The form of the feature vector *X* can be denoted as:

$$X = [\, x_1 \;\; x_2 \;\; \cdots \;\; x_n \,]^T \tag{1}$$

where *n* is the dimensionality of the feature vector (lexicon size), x_i is the frequency value of i^{th} feature and *T* refers to the transpose of a vector.

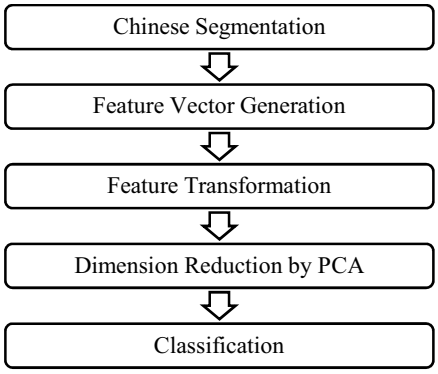

Figure 1. Procedure of Chinese text classification

Assume that the following two documents of uni-gram features in Fig.2 represent a text collection:

1. 我; 是; 一; 个; 中; 国; 人
2. 中; 国; 是; 发; 展; 中; 国; 家

Figure 2. Example of a text collection composed of two documents.

The lexicon (word list) including all different features was generated as:

{ 我 是 一 个 中 国 人 发 展 家 }

Figure 3. Lexicon (word list).

Fig.4 shows the feature vector obtained for each document from the lexicon.

1. $[\,1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 0 \;\; 0 \;\; 0\,]^T$
2. $[\,0 \;\; 1 \;\; 0 \;\; 0 \;\; 2 \;\; 2 \;\; 0 \;\; 1 \;\; 1 \;\; 1\,]^T$

Figure 4. Feature vector obtained from the lexicon (absolute frequency).

B. Feature Transformation Techniques

1) Normalization to Relative Frequency: The feature vector generated by above process is composed of the absolute frequency. In practice, textual data vary in content and length. The limitation of the absolute frequency is dependency on text length which usually leads into lower performance. This is because text length may differ within the same class of documents consequently more complexity of learning. In order to normalize the lengths of documents, absolute frequency is transformed to relative frequency:

$$y_i = \frac{x_i}{\sum_{j=1}^{n} x_j} \tag{2}$$

where x_i is the absolute frequency of feature *i* and *n* is the lexicon size. Fig.5 shows the results after transformed to relative frequency.

1. $\left[\, \frac{1}{7} \;\; \frac{1}{7} \;\; \frac{1}{7} \;\; \frac{1}{7} \;\; \frac{1}{7} \;\; \frac{1}{7} \;\; \frac{1}{7} \;\; 0 \;\; 0 \;\; 0 \,\right]^T$

2. $\left[0 \quad \frac{1}{8} \quad 0 \quad 0 \quad \frac{1}{4} \quad \frac{1}{4} \quad 0 \quad \frac{1}{8} \quad \frac{1}{8} \quad \frac{1}{8}\right]^T$

Figure 5. Relative frequency.

2) Power Transformation: The distribution of absolute/ relative frequency are generally skewed. Therefore power transformation [5] is applied to improve the symmetry of the distribution:

$$z_i = x_i^v \ (0 < v < 1) \tag{3}$$

This transformation generates Gaussian-like sample distribution. When power transformation is applied to the relative frequency with v = 0.5, the length of transformed vector is normalized to 1 which leads to higher classification performance [6]. Therefore in the experiments, v is set to 0.5.

Fig.6 shows the result when power transformation was applied to relative frequency which called relative frequency with power transformation.

1. $\left[\frac{1}{\sqrt{7}} \quad \frac{1}{\sqrt{7}} \quad \frac{1}{\sqrt{7}} \quad \frac{1}{\sqrt{7}} \quad \frac{1}{\sqrt{7}} \quad \frac{1}{\sqrt{7}} \quad \frac{1}{\sqrt{7}} \quad 0 \quad 0 \quad 0\right]^T$

2. $\left[0 \quad \frac{1}{\sqrt{8}} \quad 0 \quad 0 \quad \frac{1}{2} \quad \frac{1}{2} \quad 0 \quad \frac{1}{\sqrt{8}} \quad \frac{1}{\sqrt{8}} \quad \frac{1}{\sqrt{8}}\right]^T$

Figure 6. Relative frequency with power transformation.

C. Dimension Reduction

In ATC, high dimensionality of the feature space may be problematic in terms of computational time and storage resources. In order to solve this problem, the dimensionality is required to be reduced without deterioration of the performance.

1) Dimension Reduction by Feature Selection: N-gram and word segmentation extraction on a large corpus will yield a large number of possible features. In fact, only some of them will have significant frequency values in vectors representing the documents and good discriminating power. Yang and Pedersen [7] have shown that it is possible to reduce the dimensionality by a factor of 10 with no loss in effectiveness. Hence in the experiments, features with frequency value of 10 or less in all training data were removed to reduce the high dimensionality.

2) Dimension Reduction by PCA: Then Principal Component Analysis (PCA) was applied to further reduce the high dimensionality. PCA involves a mathematical procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. In the experiments, PCA was used to reduce the dimensionalily to 1200.

D. Classification

Support Vector Machines (SVM) is a relatively new class of machine learning techniques first introduced by Vapnik [8]. Based on the structural risk minimization principle from the computational learning theory, SVM seeks a decision surface to separate the training data points into two classes and makes decisions based on the support vectors that are selected as the only effective elements in the training set. In the experiments, we used *SVM*^{light} package [9]. We adopted three different types of SVM kernel functions: Linear Kernel (Linear), Polynomial Kernel (Poly) and Radial Basis Function (RBF).

IV. EXPERIMENTS

A. Data for Experiments

Experimental data were obtained from a Chinese corpus called TanCorpV1.0 [2] which is a new large corpus special for Chinese text classification. It was collected and processed by Songbo Tan and is categorized in two hierarchies. The first hierarchy contains 12 big categories (art, car, career, computer, economy, education, entertainment, estate, medical, region, science and sport) and the second hierarchy consists of 60 subclasses. It is totally composed of 14,150 texts. This corpus can serve as three categorization datasets: one hierarchical dataset (TanCorpHier) and two flat dataset (TanCorp-12 and TanCorp-60). In our experiments, we use TanCorp-12 for manual word segmentation and TanCorpHier for automatic word segmentation and character-based (*N*-gram) approach.

In the experiments, 150 texts were selected randomly from the corpus for each big category, and totally 1800 texts were used. The ratio of training data to test data is set as 2:1.

B. Experiment for Word-based Approach

Table II shows four different experimental methods used to evaluate word-based approach. The lexicon generated from manual word segmentation is called manual lexicon and all the words in manual lexicon are considered as ideal and correct Chinese words. The lexicon generated from automatic word segmentation is called automatic lexicon.

TABLE II. EXPERIMENTAL METHODS FOR WORD-BASED APPROACH

	Training Data	**Test Data**	**Word List**
Method 1 (M-M-m)	manual	manual	manual
Method 2 (M-A-m)	manual	automatic	manual
Method 3 (A-A-a)	automatic	automatic	automatic
Method 4 (A-A-m)	automatic	automatic	manual

Method 1 (M-M-m): Manual word segmentation was used for both training data and test data. The lexicon generated is the manual lexicon.

Method 2 (M-A-m): Manual word segmentation was used for training data and automatic word segmentation was used for test data. The lexicon generated is the manual lexicon.

Method 3 (A-A-a): Automatic word segmentation was used for both training data and test data. The lexicon generated is the automatic lexicon.

Method 4 (A-A-m): Automatic word segmentation was used for both training data and test data. Before generating feature vectors, only words that appeared on the manual lexicon were retained.

C. Experiment for Character-based (N-gram) Approach

The following methods were used as character-based (*N*-gram) approach.

1-gram: Use uni-gram feature to represent documents.

2-gram: Use bi-gram feature to represent documents.

1+2-gram: Use both uni-gram feature and bi-gram feature to represent documents.

1+2+3-gram: Use uni-gram feature, bi-gram feature and tri-gram feature to represent documents.

For each *N*-gram method, two sets of experiments were carried out. (1) The first set of experiment was performed when all the features generated from each *N*-gram method were used as the lexicon. (2) The second set of experiment was performed when using the manual lexicon. Since *N*-gram extraction on a large corpus produces large number of ambiguous words and manual word segmentation is assumed as an ideal result without any ambiguous words, only features that appeared on the manual lexicon were retained.

D. Evaluation

We adopt the most commonly used *F*-measure (*F*) metric introduced by Van Rijsbergen [10], which is the weighted harmonic mean of precision (*P*) and recall (*R*).

For ease of comparison, we summarize the *F*-measure over the different categories using the Micro-averaged *F*-measure which is viewed as a per-document average since it gives equal weight to every document. It is defined as:

$$F(\text{micro-averaged}) = \frac{2RP}{R+P} \tag{9}$$

In micro-averaging, precision and recall are obtained by summing over all individual decisions:

$$P = \frac{TP}{TP+FP} = \frac{\sum_{i=1}^{M} TP_i}{\sum_{i=1}^{M}(TP_i+FP_i)} \tag{10}$$

$$R = \frac{TP}{TP+FN} = \frac{\sum_{i=1}^{M} TP_i}{\sum_{i=1}^{M}(TP_i+FN_i)} \tag{11}$$

where *M* is the number of categories. *TP*, *TN*, *FN* and *FP* are the number of true positives, true negatives, false negatives and false positives, respectively.

V. RESULTS

A. Results of Word-based Approach

Fig.7 shows the best Micro-averaged *F*-measure comparison of four word-based approaches on three types of SVM kernels. The best Micro-averaged *F*-measure (92.20%) was achieved when using manual word segmentation for both training and test data. For other three methods when automatic word segmentation was used, the performances were significantly decreased below 87.6%. It indicates that automatic word segmentation errors are significantly reduced the performance of the classification by 4.62%. It can be also observed that when using manual lexicon instead of automatic lexicon, there is no performance improvement but rather a small decrease.

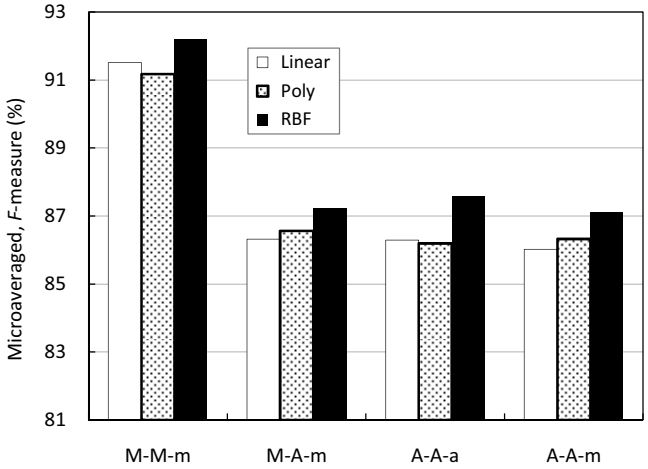

Figure 7. The best Micro-averaged *F*-measure for word-based approaches.

B. Resutls of Character-based (N-gram) Approach

Fig.8 shows the best performance comparison in Micro-averaged *F*-measure when all *N*-gram features were used. It indicates that 1+2-gram produce the highest effectiveness.

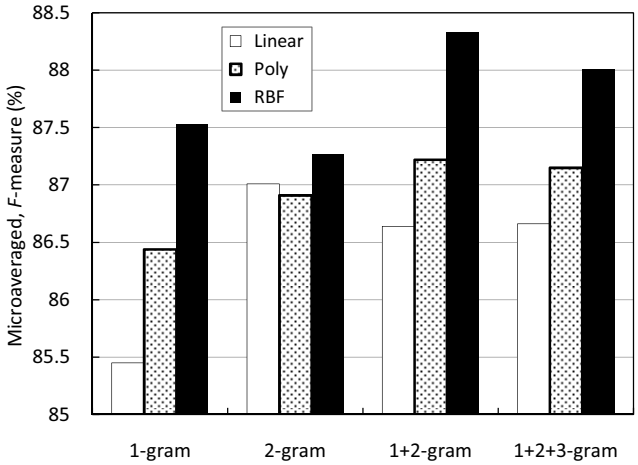

Figure 8. The best Micro-averaged *F*-measure for character-based approaches.

Fig.9 shows the best performance comparison of RBF kernel between all *N*-gram features were used and the manual lexicon was used. For each kind of *N*-gram feature, the results are similar. When ambiguous words were deleted, the performance was only slightly improved less than 0.4%, which was not as much as we expected. The possible reason

is that longer words with more than three characters may have much more significant effects of discriminate one category from another and ambiguous words do not have an obvious negative impact on classification results.

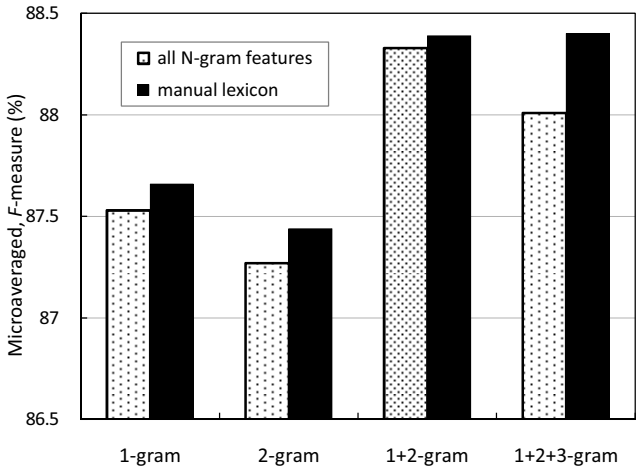

Figure 9. Comparison between all *N*-gram features were used and the manual lexicon was used of RBF kernel.

C. Comparison between Word-based Approach and Character-based Approach

The comparison between manual approach and automatic approach is shown in Table III. Method 3 (A-A-a) is the best method for automatic word-based approach and 1+2-gram is the best method for character-based approach. From Table III, it can be observed that manual word segmentation performed better than all the automatic approaches and has yielded relatively satisfactory results. However, automatic word segmentation did not perform better than 1+2-gram. Thus, it is reasonable to conclude that errors caused by automatic word segmentation significantly influence the classification performance.

TABLE III. COMPARISON BETWEEN MANUAL APPROACH AND AUTOMATIC APPROACH

		Linear	**Poly**	**RBF**
Manual Approach	**Method 1 (M-M-m)**	91.51%	91.18%	92.20%
Automatic Approach	**Method 3 (A-A-a)**	86.29%	86.20%	87.58%
	1+2-gram	86.64%	87.22%	88.33%

VI. CONCLUSION

In this paper, several sets of experiments were carried out to study the impact of word segmentation errors on Chinese text classification. Comparison experiment of word-based approach shows that the performance was significantly reduced by 4.62% when using automatic word segmentation instead of manual word segmentation. It means errors caused by automatic word segmentation have an obvious impact on classification performance. We further conducted the experiment using character-based approach (*N*-gram). Although *N*-gram approach produces a large number of ambiguous words, the results show that it still performed better than automatic word segmentation and 1+2-gram produced the highest effectiveness.

Future work includes:

1. Extensive experimental evaluation using more texts on more categories.

2. Text classification on error prone Chinese OCR texts.

REFERENCES

[1] F. Sebastiani, Machine learning in automated text categorization, ACMComputing Surveys, Vol. 34, No. 1, (March 2002), 1-47.

[2] Songbo Tan and Yuefen Wang, Chinese text categorization corpus-TanCorpV1.0. http://www.searchforum.org.cn/tansongbo/corpus.htm

[3] D. Jurafsky & J.H. Martin, An Introduction to natural language processing, computational linguistics, and speech recognition, Speech and Language Processing, Prentice Hall, 2000.

[4] Huaping Zhang, Hongkui Yu, Deyi Xiong, Qun Liu,HHMM-based Chinese Lexical Analyzer ICTCLAS.In Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, July 11-12, 2003, Sapporo, Japan.

[5] K. Fukunaga, Introduction to statistical pattern recognition, Academic Press, Inc, (1990), 76-77.

[6] L. S. P. Busagala, W. Ohyama, T. Wakabayashi, and F. Kimura, Machine learning with transformed features in automatic text classification, In Proceedings of ECML/PKDD-05 Workshop on Sub-symbolic Paradigms for Learning in Structured Domains (Relational Machine Learning), pages 11-20, 2005

[7] Y. Yang and J. O. Pedersen, A Comparative study on feature selection in text categorization, In Proceedings of ICML-97, 14th International Conference on Machine Learning (Nashville, TN, 1997), 412-420.

[8] Corinna Cortes and V. Vapnik, Support-Vector Networks, Machine Learning, 20, 1995.

[9] T. Joachims, Learning to classify text using support vector machines: Methods, Theory and Algorithms, Kluwer Academic Publishers Boston Dordrecht London, 2001.

[10] C. van Rijsbergen, Information Retrieval, Butterworths, London, 1979.