Context-Based Chinese Word Segmentation using SVM Machine-Learning Algorithm without Dictionary Support

Chia-ming Lee
Department of Engineering Science and Ocean Engineering,
National Taiwan University,
Taipei, Taiwan (R.O.C.)
trueming@gmail.com

Chien-Kang Huang
Department of Engineering Science and Ocean Engineering,
National Taiwan University,
Taipei, Taiwan (R.O.C.)
ckhuang@ntu.edu.tw

Abstract

This paper presents a new machine-learning Chinese word segmentation (CWS) approach, which defines CWS as a break-point classification problem; the break point is the boundary of two subsequent words. Further, this paper exploits a support vector machine (SVM) classifier, which learns the segmentation rules of the Chinese language from a context model of break points in a corpus. Additionally, we have designed an effective feature set for building the context model, and a systematic approach for creating the positive and negative samples used for training the classifier. Unlike the traditional approach, which requires the assistance of large-scale known information sources such as dictionaries or linguistic tagging, the proposed approach selects the most frequent words in the corpus as the learning sources. In this way, CWS is able to execute in any novel corpus without proper assistance sources. According to our experimental results, the proposed approach can achieve a competitive result compared with the Chinese knowledge and information processing (CKIP) system from Academia Sinica.

1 Introduction

Chinese sentences contain sequences of characters that are not delimited by white spaces or any other symbol used for word identification, so Chinese word segmentation (CWS) is one of the fundamental issues in Chinese natural language processing studies.

One of the major aspects in existing CWS researches is the resolution of word segment ambiguities. The conventional approach of ambiguity detection is to use two maximum matching methods (MMs), which scan corpora forward (Forward Maximum Matching, FMM) and backward (Backward Maximum Matching, BMM) based on dictionaries (Kit, Pan, & Chen, 2002). Meanwhile, disambiguation methods can be classified into two different categories: rule-based methods and statistical-based methods. (Ma & Chen, 2003b). Problem disambiguity is often accompanied by the problem resolution of an unknown word or out-of-vocabulary (OOV) extraction (K.-J. Chen & Ma, 2002). Besides the MMs with dictionaries, which are also known as word-based approaches, there are character-based approaches. The word-based approach treats words as the basic unit of a language, and the character-based approach labels each character as the beginning, middle, or end of a word. Character-based approaches are often implemented with a machine-learning classification algorithm for handling disambiguation (Wang, Zong, & Su, 2012). In addition to dictionaries, other linguistic resources such as part-of-speech (POS) or semantic information can be integrated for further improvement (M.-y. Zhang, Lu, & Zou, 2004).

In addition to the disambiguation strategy, many researchers provide the best word sequence identification methods for their CWS. The Hidden Markov model (HMM) (Lin, 2006; M.-y. Zhang et al., 2004), maximum entropy (ME), mutual information (MI) and boundary dependency (Peng & Schuurmans, 2001) are often used. Theoretically, to get the best CWS result is to obtain the optimized word sequence.

As described above, existing CWS research takes either words or characters as the core unit of their methodologies. Instead of identifying word ambiguity, finding word sequence or joining characters into words, we redefine the CWS problem as the identification of "break points" among the "joint points" in Chinese character sequences. In this paper, we define a "joint

International Joint Conference on Natural Language Processing, pages 614–622,
Nagoya, Japan, 14-18 October 2013.

point" as a point between adjacent characters, and a "break point" as the boundary of two subsequent words; further, the characters between two break points will consist of words.

The identification of break points among joint points is a binary classification problem. In this study, we use a support vector machine (SVM) machine-learning algorithm with contextual statistical measures to construct the feature vector model of the joint points. Based on our assumption that the Chinese word segmentation rule can be learned from non-linguistic contextual information, all features selected for the joint point model are purely statistical measures without any linguistic tagging information. Moreover, a systematic approach for creating effective positive and negative samples is provided for training the SVM classifier.

Furthermore, in order to meet the need of a CWS approach for a novel corpus, which has no appropriate dictionaries or linguistic tagging, in this study, we select a small set of assistant known source from the experimental corpus as the learning samples, which can be reduced to only 3 words: the most frequent bi-gram, tri-gram, and four-gram words. The experimental results show that by using the joint point model within long contextual information, a small set of learning samples can lead to competitive CWS results compared with the Chinese knowledge and information processing (CKIP) system, which is supported by a large-scale term database that contains approximately 5 million Chinese terms, from Academia Sinica.

2 Related Works

Conventionally, ambiguity and OOV are two major problems in the field of CWS research (K.-J. Chen & Ma, 2002). From the methodological perspective, there are rule-based, statistical-based, and machine-learning approaches (Kit et al., 2002; Peng & Schuurmans, 2001; Wang et al., 2012). Moreover, on the basis of the basic language unit used, existing research can be categorized into either word-based or character-based methods (Y. Zhang & Clark, 2007; Zhao, Huang, Li, & Lu, 2010). Most CWS research has resolved problems using labeled corpora while a few have managed CWS using pure text corpora (Dai, Loh, & Khoo, 1999; Jin Kiat Low, 2005). In labeled corpora, the tagging of dictionary matches, parts-of-speech, semantics, and character positions inside a word, are all popular methods for incorporating known information (Kit et al., 2002).

2.1 Ambiguity and the unknown word

There are two types of ambiguities in CWS: overlapping and combinational ambiguities. They can be defined as follows: given a dictionary D and a string "abc," if the set of sub-strings $\{ab, bc\} \subset D$, "abc" involves an overlapping ambiguity; given a dictionary D and a string "ab," if the set of sub-strings $\{a, b, ab\} \subset D$, "ab" involves a combinational ambiguity.

Conventional dictionary-based FMM and BMM are straightforward strategies for detecting ambiguities (Kit et al., 2002) and certainly provide an applicable foundation for disambiguation methods. However, dictionaries can never contain all words. Every corpus will have, on average, 3% to 5% OOV words (K.-J. Chen & Ma, 2002); hence, the identification of unknown words has become an important branch of CWS studies (K.-J. Chen & Ma, 2002; Ma & Chen, 2003a). Besides MMs, there are other corpus-based learning approaches to detect ambiguities for CWS (K.-J. Chen & Bai, 1998).

2.2 Word-based and character-based approaches

Another way to catalogue CWS is dependent on the basic information unit used; there are both word-based and character-based CWS methods. Word-based approaches treat the word as the basic unit, and POS and other word-based linguistic resources are often integrated into such approaches in order to improve the CWS results. From this point of view dictionary-based approaches can be treated as word-based approaches. Character-based approaches disregard the linguistic information and directly calculate the character-to-character statistical features. One popular way is to label each character as the beginning, middle, or the end of a word, and generate sequence words in sentences on the basis of the position labels of the characters (Goh, 2005; Peng & Schuurmans, 2001; Zhao et al., 2010). There are few character-based CWS approaches that use pure text corpora without additional label information (Dai et al., 1999; Jin Kiat Low, 2005).

2.3 Rule-based, statistical-based, and machine-learning methods

From the methods perspective, the earlier CWS used heuristic rules to resolve ambiguities (Ma &

Chen, 2003b) accompanied by the development of unknown word extraction or identification technologies (Ma & Chen, 2003a). Besides rule-based approaches, statistic-based approaches involved the concept of language models trained on large-scale corpora, and many such algorithms have been used and improved over time, such as Maximum Entropy (ME), Mutual Information (MI), and boundary dependency (Jin Kiat Low, 2005; Peng & Schuurmans, 2001). Some statistic-based approaches do not focus on resolving ambiguities, but provide strategies for word sequence identification in sentences. In general, statistical-based approaches tend to provide a generative or discriminative (Wang et al., 2012) probability formula for Chinese words. In contrast, machine-learning approaches pay more attention to the selection of effective features for Chinese word representations. The HMM (Lin, 2006; M.-y. Zhang et al., 2004) and SVM (Li, Huang, Gao, & Fan, 2005) are popular in CWS studies. Currently, a combination of a character-based approach and statistical or machine-learning algorithms is a common strategy for CWS (Goh, 2005; Wang et al., 2012; Zhao et al., 2010).

2.4 Contextual information

Dai and Loh have proposed "The Contextual Information Formula" of Chinese bi-gram words (Dai et al., 1999). It is an MI improving formula trained on a large-scale corpus. In this formula, the frequency of a sample bi-gram, the frequencies of its context characters and document frequencies of its context bi-grams are used. They suggest that Chinese words can be defined by a non-linguistic formula that depends on context character measures. Low and Ng conducted a series of studies using context features for their CWS research (Jin Kiat Low, 2005). Further, the concept of contextual information has often been used in information extraction research as well as in existing Chinese term extraction research for entity identification (Gao, 2005; Lee, 2012). In addition, Japanese has no word delimiter like Chinese. Sassano and Neubig et al. have defined Japanese word segmentation (JWS) as a classification task of word boundaries, and also used contextual feature sets in their studies (Neubig, Nakata, & Mori, 2011; Sassano, 2002). Inspired by these ideas of using contextual information, our research aims to extract a contextual information feature vector of "joint points" and uses an SVM algorithm to train a break point classifier.

2.5 Complete lexical patterns

Chien has proposed the estimation of complete lexical patterns (Figure 1) (Chien, 1999) in a series of Chinese term extraction papers. There are three important measures used in these lexical patterns, including association, and left and right dependency. These three measures will be integrated into our contextual information feature vector.

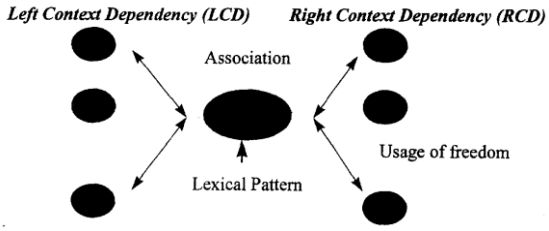

Figure 1. The estimation of complete lexical patterns

(1) Association(AEc) = $f(x) / (f(y)+f(z)–f(x))$
x is the lexical pattern; $x = x_1, x_2, \ldots, x_n$; $y = x_1, \ldots, x_{n-1}$, $z = x_2, \ldots, x_n$; f(x) is the frequency of x; $f(y)$ is the frequency of y; $f(z)$ is the frequency of z
(2) Left Context Dependency(LCD) = f(max_xl) / f(x)
f(max_x_l) is the maximum frequency of distinct characters to the left of x
(3) Right Context Dependency(RCD) = f(max_x_r) / $f(x)$
f(max_x_l) is the maximum frequency of distinct characters to the right of x

3 Method

In the proposed approach, the CWS was treated as the problem of identifying word break points among joint points in a corpus, and we resolved it by using a SVM machine-learning classifier. The term "joint point" in this paper refers to a point between two adjacent Chinese characters. Our approach is to classify all joint points into either a break point class or non-break point class. The function of break points is similar to that of white spaces in sentences in English.

The SVM is a multi-vector classification algorithm (Boser, Guyon, & Vapnik, 1992). It is also a two-phase algorithm that employs a model-training phase and a model-using (predicting) phase. The major task of the model-training phase is collecting learning samples in different classes and extracting sample feature vectors for training the SVM model. In the model-using phase, the SVM will predict which class an un-

known sample belongs to. Unknown samples need to be formed using the same feature vector as the learning samples. In this paper, we set two classes, the break point class and the non-break point class, and the final predicted break-point outputs are the results of our CWS.

3.1 Positive and negative contextual sample generation

In this paper, we propose an efficient method of contextual learning sample generation to build a two class SVM classifier with positive learning samples for the break point class and negative learning samples for the non-break point class. Because break points are the boundaries of words, we first collect the known words in the corpus, and take their boundary points as the positive samples. In contrast, the negative samples are the joint points inside these words. This means that every matching of a word will get two positive learning samples. It will also get one negative (learning sample) for a bi-gram match, tow negatives for a tri-gram match and three negatives for a four-gram match.

Take the sentence, "…。我行菩薩道時， …" (Figure 2), from the experimental corpus as an example, there are nine joint points, p1~p9, in this case. In this sentence, "。" is the period and "， " is the comma in Chinese, and "我行菩薩道時 " means 'As I practice the way of Bohdhisattva.' In this case, if 菩薩道 'the way of Bohdhisattva', is a collected known word, then p4 and p7 will be the positive samples and p5 and p6 will be the negative samples; the other joint points will be the unknown samples.

Figure 2. Sample selection example

Joint points, including positive and negative learning samples, and unknown samples, are not characters and therefore do not take up space in a corpus. For making specific samples of the joint points, we always take the same number of characters before and after the joint point to generate contextual samples. Take "p4" in Figure 2 as an example, depending on how long the context information we want to integrate, it can be sampled as a short-distance bi-gram 行菩, which takes one character on both sides of p4, a four-gram 我行菩薩, or a longer-distance n-gram contextual positive sample. In our experiments, a six-gram contextual sample, catching three characters on both sides of a learning sample, support a better SVM CWS classifier.

The known words, or learning words, for contextual learning sample generation can be collected from dictionaries. However, for the purpose of reducing the preparation loading of CWS for a novel corpus without appropriate dictionaries, in this study, we also set the highest-frequency bi-gram, tri-gram and four-gram words in the corpus for the learning samples generation. Hence, the known words can be collected systematically in this way, and the experiment results suggest that the small size of the known words leads to a competitive result compared with the big numbers of known words, which collected from dictionaries.

The reason for using the most frequent bi-gram, tri-gram and four-gram words, but not uni-gram words is that single-character words do not have a negative case, which would cause an imbalance of positive and negative learning samples. Further, bi-gram words are found to be the majority in Chinese texts, and long words tend to be combinations of short words (梁曉虹, 2005). Further, based on our observation, the highest-frequency bi-gram, tri-gram and four-gram in a Chinese corpus are almost always words and nouns, as well.

3.2 Feature vector extraction

The contextual learning sample needs to be modeled as a feature vector for the machine-learning algorithm. There are 10 types of features chosen for the feature vector extraction of the contextual learning sample, including frequency, the number of distinct characters to the left and right, the number of breaking symbols (non-Chinese characters and paragraph marks) to the left and right, association, and the usage freedom to the left and right of characters in the contextual sample. Among these features, association and the usage freedom (also called left and right context dependency) refer to "The Estimation of Complete Lexical Patterns" as proposed by Chien (Chien, 1999). Table 1 shows the complete feature set used in our experiment.

For feature vector extraction, we applied the feature set to every bi-gram plus all uni-gram frequencies within the contextual learning sample. In this way, the long-context feature vector was modeled by measures of short strings, and it could keep particular context information and avoid the probability sparsity to features. Take

the four-gram learning sample, 我行菩薩, generated from p4 in Figure 2 as an example, there are a total of 34 features in its feature vector, including the four uni-gram frequencies of 我, 行, 菩, and 薩, and 30 features from three bi-gram feature sets of 我行, 行菩, and 菩薩. Hence, depending on different extended length of context, there will be 56 features for a six-gram sample and 78 features for an eight-gram sample. Table 2 shows the numbers of features in different lengths of contextual samples.

No.	Features	
1	Frequency	
2	Association (AEc) measure	
3	Number of distinct characters	to the left side
4	Maximum frequency of distinct characters	
5	Number of breaking symbols	
6	Left-context dependency (LCD) measure	
7	Number of distinct characters	to the right side
8	Maximum frequency of distinct characters	
9	Number of breaking symbols	
10	Right-context dependency (RCD) measure	

Table 1. Feature set

Contextual sample length	4	6	8
-uni-grams	4	6	8
-bi-grams	3	5	7
-frequency of each uni-gram	4	6	8
-feature set of bi-grams	30	50	70
Total number of features	34	56	78

Table 2. Number of features in a four-gram feature vector

3.3 SVM algorithm and Libsvm package

The Support Vector Machine (SVM) algorithm constructs a hyper-plane in a high-dimensional space for classification and other tasks (Cristianini & Shawe-Taylor, 2000). A good separation is achieved by the hyper-plane farthest from the nearest training data point of any class.

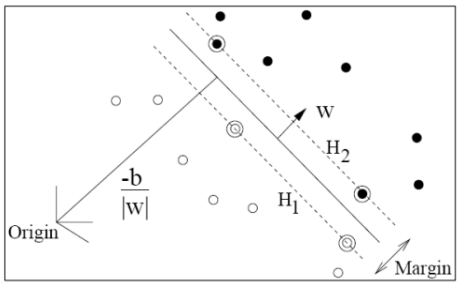

Figure 3. Support Vector Machine (SVM)

In Figure 3, W is the good separation (the classification hyper-plane) of the two classes—white spots and black spots—and H1 and H2 are the support hyper-planes.

$\mathbf{W}^T\mathbf{X}+b=0$

H1: $\mathbf{W}^T\mathbf{X}+b=1$

H2: $\mathbf{W}^T\mathbf{X}+b=-1$

To maximize the distance between H1 and H2 (2 / ||w||):

$$L(w,b,\alpha)=\frac{1}{2}\|w\|^2-\sum_{i=1}^{N}\alpha_i\left[y_i\left(w^T x_i+b\right)-1\right]$$

In our study, Libsvm tools (Chang & Lin, 2011) were used for executing the SVM algorithm. The SVM algorithm includes two phases: model training and model using, called break-point-predicting phases. In the model-training phase, the input data for the Libsvm package is the feature vector data set of all the learning samples, both positives and negatives, and the output data is a classification model file. Meanwhile, all joint points are considered to be unknown samples. The unknown samples should be converted to the feature vector data set in the exactly same way as the learning samples are. In the break-point-predicting phase, the input data is the feature vector data set of the unknown samples and the classification model file output from the earlier phase, and the output data contents of the joint points, which are predicted to be break points. The predicted output data are the CWS results.

4 Experiment

4.1 Corpus

The collection of Saddharma Puṇḍarīka (Lotus of the True Dharma), which is part of a Chinese text archive from the Middle Ages provided by the Chinese Buddhist Electronic Text Association (CBETA), was selected as the experimental corpus. It consists of 16 sutras labeled T0262 to T0277 of the Taisho Revised Tripitaka. This corpus contains 514,722 Chinese characters without punctuation, and there are a total of 514,721 joint points available for the experiment.

Generally speaking, CWS in ancient Chinese corpora is usually difficult than in modern Chinese collections, as the modern dictionaries are not very suitable for ancient Chinese collections, plus ancient Chinese collections lack punctuations and stop-words. Since the proposed method was designed to solve the CWS without the use of a dictionary, this collection is a good corpus to

demonstrate the powerfulness of the proposed method.

4.2 Performance evaluation method

In this study, we selected paragraphs, evaluation texts, from the experimental corpus, and compared the results of the evaluation texts from a subject matter expert's answers and the SVM CWS predicted answers as a means of evaluating the system's performance.

Sàtánfēntuólìjīng, a sutra (T0265) from the collection of Saddharma Puṇḍarīka was chosen as the evaluation text. In Sàtánfēntuólìjīng, there are 1,588 joint points; the ratio size of the evaluation text is 0.3% of the entire corpus. The evaluation text was not included in the training data, and experts provided 616 break points, true answers, and 972 non-break points, false answers for it. Precision, recall, and f-measure were used for evaluating the effectiveness of the CWS results. The evaluation definitions were as follows:

$$Precision = \frac{True\ Positive}{True\ Positive\ +\ False\ Positive}$$

$$Recall = \frac{True\ Positive}{True\ Positive\ +\ False\ Negative}$$

$$F\text{-}measure = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

4.3 Experiments

In order to reveal the effectiveness of the training data size, we prepared three training sets; each had different numbers of training data. The first set consisted of learning samples collected from the highest-frequency bi-gram, tri-gram and four-gram, for a total of three Chinese words in the corpus. The second set consisted of learning samples from the top-10 high-frequency words of each bi-gram, tri-gram, and four-gram in the corpus, and the third set consisted of learning samples from 7,309 words, which were collected from the book index of the corpus (大藏經學術用語研究會, 198-?). Table 3 shows the number of learning samples for each training size.

At the end of the experiment, we compared the overall performance with that of CKIP, which is a Chinese word segmentation system supported by 4,892,324 Chinese-word database (Sinica, 2013).

In Table 3, the three highest-frequency words in the first training set are 菩薩 'bodhisattva' (the bi-gram), which had a frequency of 3133; 摩訶薩 'mahasattva' (the tri-gram), which had a frequency of 382; and 文殊師利 'manjushri', a name of the bodhisattvas (the four-gram), which had a frequency of 514. This is a total of 4,029 matched strings, which contributed 7,658 positive and 5,439 negative samples. Since the matched strings are adjacent in some places, the total number of positives is not exactly twice the summation of the three frequencies. However this does not apply to the negatives samples because there is no commonality of location.

	Highest-frequency words	Top-10 high-frequency words	Dictionary-based group
bi-gram	1	10	2678
tri-gram	1	10	2227
four-gram	1	10	2404
Total words	3	30	7,309
Total positives	7658	35,199	150,441
Total negatives	5439	23,677	105,035

Table 3. Learning sample comparison of three training sets

Besides setting a different size of the training data, we set different context distances, character extensions in context, of samples. The more the context characters are extended, the more is the contextual information involved in the feature vector model. Hence, a two-character extension in context means catching 2 characters on both sides of the joint points to make a 4-gram context learning sample. Table 4 shows the CWS results for the highest-frequency training set, Table 5 shows the results of the top-10 high-frequency training set, and Table 6 shows the results of the dictionary-based training set. Every group was segmented in three different context distances.

Sample length	Four-gram	Six-gram	Eight-gram
Context extension	2 characters	3 characters	4 characters
Precision	51.1%	51.2%	49.5%
Recall	94.5%	94.2%	96.3%
F-measure	66.3%	66.3%	65.3%

Table 4. CWS results of the highest frequency words

In the tables, the dictionary-based results (Table 6) exhibit stable performances; the results growing with context distances. Although the performance of the set of the highest-frequency words is not as good as that of the dictionary-based ones, it is still competitive, and most importantly, it used no assistant sources outside of the corpus.

We believe that this shows the potential of the non-dictionary CWS method proposed in this paper.

Sample length	Four-gram	Six-gram	Eight-gram
Context extension	2 characters	3 characters	4 characters
Precision	56.6%	56.7%	57.0%
Recall	83.4%	82.1%	81.7%
F-measure	67.4%	67.1%	67.2%

Table 5. CWS results of the top 10 high frequency words

Sample length	Four-gram	Six-gram	Eight-gram
Context extension	2 characters	3 characters	4 characters
Precision	57.9%	58.6%	59.1%
Recall	79.5%	80.4%	81.2%
F-measure	67.0%	67.8%	68.4%

Table 6. CWS results of the known words from the index book

4.4 Feature selection analysis

This section analyzes the importance of features used in the SVM classifier. A total of 56 features, in the highest frequency word dataset with 6-gram learning samples, were calculated and sorted by the f-score algorithm proposed by Chen and Lin's SVM feature-selected research (Y.-W. Chen & Lin, 2006).

Table 7, the top 10 features of the training dataset, shows that the contextual dependency measures around joint points have a significant influence on the SVM classifier.

4.5 Iterative CWS strategy

Because the learning samples can be collected systematically and generated from very few words in the proposed CWS method, we provide an iterative training process to improve the CWS results. In the iterative CWS strategy, we select training samples for the next SVM CWS iterative round from the previous SVM CWS results.

Libsvm provides a probability measure for every joint point in the predicting phase, and in the Libsvm default setting, joint points will be classified in to the break point class when their predicting probability is greater than 50%, which is also the SVM classifier predicting threshold in our experiments.

Based on the probability measure, in the iterative experiment, points whose probability was greater than 90% were taken as positives and points whose probability was less than 10% were taken as the negatives for the next round. In this way, the size of positives and negatives is imbalanced, so we set a stricter threshold on the side having bigger numbers to make both sides have the same number of learning samples.

Table 8 shows a three-round iterative CWS result using the highest-frequency words training set with the context extension of three characters, the Six-gram learning samples, which led to better performance in the earlier experiment. Based on the performance evaluation over all rounds, the precision in the second round increased by approximately 10%, but other CWS results did not improve as expected.

No.	Features of six-character context sample "ABCDEF"	f-score
1	RCD of "CD"	0.4420
2	LCD of "CD"	0.3304
3	LCD of "DE"	0.3281
4	RCD of "BC"	0.3144
5	Number of distinct characters to the left of "CD"	0.2284
6	Number of distinct characters to the right of "CD"	0.2199
7	AEc of "CD"	0.2108
8	Number of distinct characters to the left of "DE"	0.1598
9	LCD of "BC"	0.1513
10	Number of breaking symbols to the left of "CD"	0.1480

Table 7. Top 10 features of training dataset

	Iteration 1	Iteration 2	Iteration 3
Positives	7658	39520	163849
Negatives	5439	39520	163849
Total learning samples	13097	79040	327698
Precision	51.2%	62.3%	61.6%
Recall	94.2%	66.2%	65.3%
F-measure	66.3%	64.2%	63.4%

Table 8. CWS results of the iterative experiment

4.6 Comparison

Table 9 compares four different CWS results: the highest-frequency words, top-10 high-frequency words, the dictionary group, and the results from CKIP. The best results of each method are shown in this table. CKIP is the segmentation tool by Sinica, which enhances the segmentation using a large-scale term database having approximately 5-million, cross-field Chinese words (Group; Ma & Chen, 2003b). The comparison table shows

that the control group has higher recall, the dictionary-based group has higher precision and the CKIP exhibits a more balanced result.

	Highest frequency words	Top10 high-frequency words	Diction-ary	CKIP
Precision	51.2%	57.0%	59.1%	57.4%
Recall	94.2%	81.7%	81.2%	88.3%
F-measure	66.3%	67.2%	68.4%	69.6%

Table 9. Performance comparison

5 Conclusion and Discussion

In this paper, we proposed a novel corpus machine-learning CWS approach that identified break points from joint points. The proposed approach is different from existing researches, which tended to create a generating model or formula of Chinese words. In this study, we provided a long-distance context model of joint points and defined the model by non-linguistic contextual features. The experimental results suggested that break points among Chinese texts could be identified on the basis of their non-linguistic contextual features in our chosen corpus.

According to the experimental results, the proposed approach can achieve precision 51.2% and recall 94.2% with only 3 learning words systematically selected from the experiment corpus. It is a very competitive result comparing with the CKIP system, which achieves precision 57.4% and recall 88.3%, and it is supported by an approximately 5-million Chinese-word database. Therefore, this study met the need of carrying out CWS in a novel corpus without appropriate dictionaries.

Further, the proposed approach can systematically select balanced positive and negative learning samples staring from a very small number of learning words. Hence, we chunked long-distance context samples into short-distance strings, uni-grams and bi-grams, for feature vector extraction. Thus, we could collect long-distance context information without dealing with the probability sparsity problem.

Since the CWS rules can be trained from context without linguistic information, the proposed CWS method might also work for Chinese texts from different ages. However, there are some issues and problems that require further investigation.

First, the selection of learning words can affect the final performances. Different learning words may cause the different results, and this affection needs to be further studied. For instance, if we took learning words by their parts-of-speech instead of frequency, the proposed approach might change its behavior.

Further, the detection of combination words and the overlapping problem needs to be addressed. The Libsvm classifier can assign a predicting probability measure to every joint point. Instead of setting a threshold to filter out break points via these probabilities, these probability measures can be used for identifying the combination words and detecting overlapping problems, as well.

Finally, the effect of iterative process needs to be further studied. Currently, the iterative results can led to better precision in the second round. However, it performs worse in recall and f-measure. Besides, other iteration parameters need to be decided, such as the number of iteration, the optimal predicting threshold, and the saturation condition for stopping the iterative process properly.

Acknowledgments

The authors would like to thank the National Science Council of the Republic of China, Taiwan, for financially supporting this research under Contract No. 102-2420-H-002-050-MY2 and No. 102-2410-H-002-083-.

References

Boser, Bernhard E., Guyon, Isabelle M., & Vapnik, Vladimir N. (1992). *A training algorithm for optimal margin classifiers*. Paper presented at the Proceedings of the fifth annual workshop on Computational learning theory, Pittsburgh, Pennsylvania, United States.

Chang, Chih-Chung, & Lin, Chih-Jen. (2011). LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol., 2*(3), 1-27. doi: 10.1145/1961189.1961199

Chen, Keh-Jiann, & Bai, Ming-Hong. (1998). *Unknown Word Detection for Chinese by a* Corpus-*based Learning Method*. Paper presented at the Computational Linguistics and Chinese Language Processing.

Chen, Keh-Jiann, & Ma, Wei-Yun. (2002). *Unknown word extraction for Chinese documents*. Paper presented at the Proceedings of the 19th international conference on Computational linguistics - Volume 1, Taipei, Taiwan.

Chen, Yi-Wei, & Lin, Chih-Jen. (2006). Combining SVMs with Various Feature Selection Strategies. *Feature Extraction - Studies in Fuzziness and Soft Computing, 207*, 315-324.

Chien, L. (1999). PAT-tree-based adaptive keyphrase extraction for intelligent Chinese information retrieval. *Information Processing and Management, 35*(4), 501.

Cristianini, Nello, & Shawe-Taylor, John. (2000). *An* introduction *to support Vector Machines: and other kernel-based learning methods*: Cambridge University Press.

Dai, Yubin, Loh, Teck Ee, & Khoo, Christopher S. G. (1999). *A new statistical formula for Chinese text* segmentation *incorporating contextual information*. Paper presented at the Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, Berkeley, California, United States.

Gao, J. (2005). Chinese word segmentation and named entity recognition: A pragmatic approach. *Computational Linguistics, 31*(4), 531.

Goh, C. L. (2005). Chinese word segmentation by classification of characters. *International Journal of Computational Linguistics and Chinese Language Processing, 10*(3), 381.

Group, Chinese Knowledge Information Processing. CKIP Chinese Word Segmentation System, from http://ckipsvr.iis.sinica.edu.tw/

Jin Kiat Low, Hwee Tou Ng, Wenyuan Guo. (2005). *A maximum entropy approach to Chinese* word *segmentation.* Paper presented at the Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing, Korea.

Kit, Chunyu, Pan, Haihua, & Chen, Hongbiao. (2002). Learning *case-based knowledge for disambiguating Chinese word segmentation: a preliminary study*. Paper presented at the Proceedings of the first SIGHAN workshop on Chinese language processing - Volume 18.

Lee, Chia-ming; Huang, Chien-Kang; Shi, Fayuan; Chen, Kuang-Hua. (2012). *Iterative Chinese Bi-*gram *Term Extraction Using Machine-learning Classification Approach.* Paper presented at the The 1st Workshop on Optimization Techniques for Human Language Technology -- Coling 2012, Mumbai, India.

Li, Hongqiao, Huang, Chang-Ning, Gao, Jianfeng, & Fan, Xiaozhong. (2005). The Use of SVM for Chinese New Word Identification

Natural Language Processing – IJCNLP 2004. In Keh-Yih Su, Jun'ichi Tsujii, Jong-Hyeok Lee & Oi Kwong (Eds.), (Vol. 3248, pp. 723-732): Springer Berlin / Heidelberg.

Lin, Qian-Xiang. (2006). *Chinese Word Segmentation using Specializd HMM*. (Master), National Central University, Jhongli.

Ma, Wei-Yun, & Chen, Keh-Jiann. (2003a). *A bottom-up merging algorithm for Chinese* unknown *word extraction*. Paper presented at the Proceedings of the second SIGHAN workshop on Chinese language processing - Volume 17, Sapporo, Japan.

Ma, Wei-Yun, & Chen, Keh-Jiann. (2003b). Introduction *to CKIP Chinese word segmentation system for the first international Chinese Word Segmentation Bakeoff*. Paper presented at the Proceedings of the second SIGHAN workshop on Chinese language processing - Volume 17, Sapporo, Japan.

Neubig, Graham, Nakata, Yosuke, & Mori, Shinsuke. (2011). *Pointwise prediction for robust, adaptable Japanese morphological analysis*. Paper presented at the Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2, Portland, Oregon.

Peng, Fuchun, & Schuurmans, Dale. (2001). Self-Supervised Chinese Word Segmentation. In Frank Hoffmann, DavidJ Hand, Niall Adams, Douglas Fisher & Gabriela Guimaraes (Eds.), *Advances in Intelligent Data Analysis* (Vol. 2189, pp. 238-247): Springer Berlin Heidelberg.

Sassano, Manabu. (2002). *An empirical study of active learning with support vector machines for Japanese word segmentation*. Paper presented at the Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, Pennsylvania.

Sinica, Academia. (2013). Academia Sinica Balanced Corpus of Modern Chinese, from http://db1x.sinica.edu.tw/cgi-bin/kiwi/mkiwi/mkiwi.sh

Wang, Kun, Zong, Chengqing, & Su, Keh-Yih. (2012). Integrating Generative and Discriminative Character-Based Models for Chinese Word Segmentation. *ACM Transactions on Asian Language Information Processing 11*(2), 1-41. doi: 10.1145/2184436.2184440

Zhang, Mao-yuan, Lu, Zheng-ding, & Zou, Chun-yan. (2004). A Chinese word segmentation based on language situation in processing ambiguous words. *Inf. Sci., 162*(3-4), 275-285. doi: http://dx.doi.org/10.1016/j.ins.2003.09.010

Zhang, Yue, & Clark, Stephen. (2007). *Chinese Segmentation with a Word-Based Perceptron* Algorithm. Paper presented at the Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics.

Zhao, Hai, Huang, Chang-Ning, Li, Mu, & Lu, Bao-Liang. (2010). A Unified Character-Based Tagging Framework for Chinese Word Segmentation. *ACM Transactions on Asian Language Information Processing, 9*(2), 1-32. doi: 10.1145/1781134.1781135

大藏經學術用語研究會. (198-?). *大藏經索引*. 台北: 新文豐.

梁曉虹, 徐時儀, 陳五雲. (2005). *佛經音義與漢語詞彙研究*. 北京: 北京商務印書館.