# Chinese Word Segmentation Based on Contextual Entropy


## Abstract

Chinese is written without word delimiters. Word segmentation is generally considered the key step in processing Chinese texts. This paper presents a new statistical approach to segment Chinese sequences into words. This approach is based on contextual entropy on both sides of a bigram. It is used to capture the dependency with the left and right contexts in which a bigram occurs. Our approach tries to find the word boundaries instead of words for segmentation. Experimental results show that it is effective for Chinese word segmentation.


## 1 Introduction

Unlike English there is no explicit word boundary in Chinese text. Chinese words can comprise one, two, three or more characters without delimiters. But almost all techniques to Chinese language processing, including machine translation, information retrieval and natural language understanding are based on words. Word segmentation is a key step in Chinese language processing.

Several approaches have been developed for Chinese word segmentation. In general two main approaches are widely used: the statistical approach (Gua and Gan, 1994, Sproat and Shih, 1990, Dai, et al, 1999, Teaban, et al., 2000) and rule-based approach (Yeh and Lee, 1991, Swen and Yu, 1999). Some statistical approaches are based on the mutual information (Sproat and Shih, 1990), which only captures the dependency among characters of a word. Some need large pre-tagged corpus for training (Teaban, et al., 2000), which is too expensive to construct at present. Rule-based approaches require a pre-defined word list (dictornary, or lexicon). The coverage of the dictionary is critcal for these aprroaches. Many researches use a combination of approaches (Nie, Jin and Hanna 1994).

It has been long known that contextual information can be used for segmentation (Harris 1955). Dai (1999) used weighted document frequency as contextual information for Chinese word segmentation. Zhang, Gao and Zhou (2000) used the context dependency for word extraction. Tung and Lee (1994) used contextual entropy to identify unknown Chinese words. Chang, et al (1995) and Ponte, et al (1996) used contextual entropy for automatic lexical acquisition. Hutchens and Alder (1998) and Kempe(1999) used the contextual entropy to detect the separator in English and German corpus.

In this paper we will present a simple purely statistical approach using contextual entropy for word segmentation. Details about our approach are covered in section 1 and 2.

## 2 Contextual Entropy

We use a Markov model to estimate the probabilities of symbols of a corpus. The probability of a symbol w with respect to this model M and to a context c can be estimated by:

$$P(w \mid M, c) = \frac{f(w, M, c)}{f(M, c)}$$

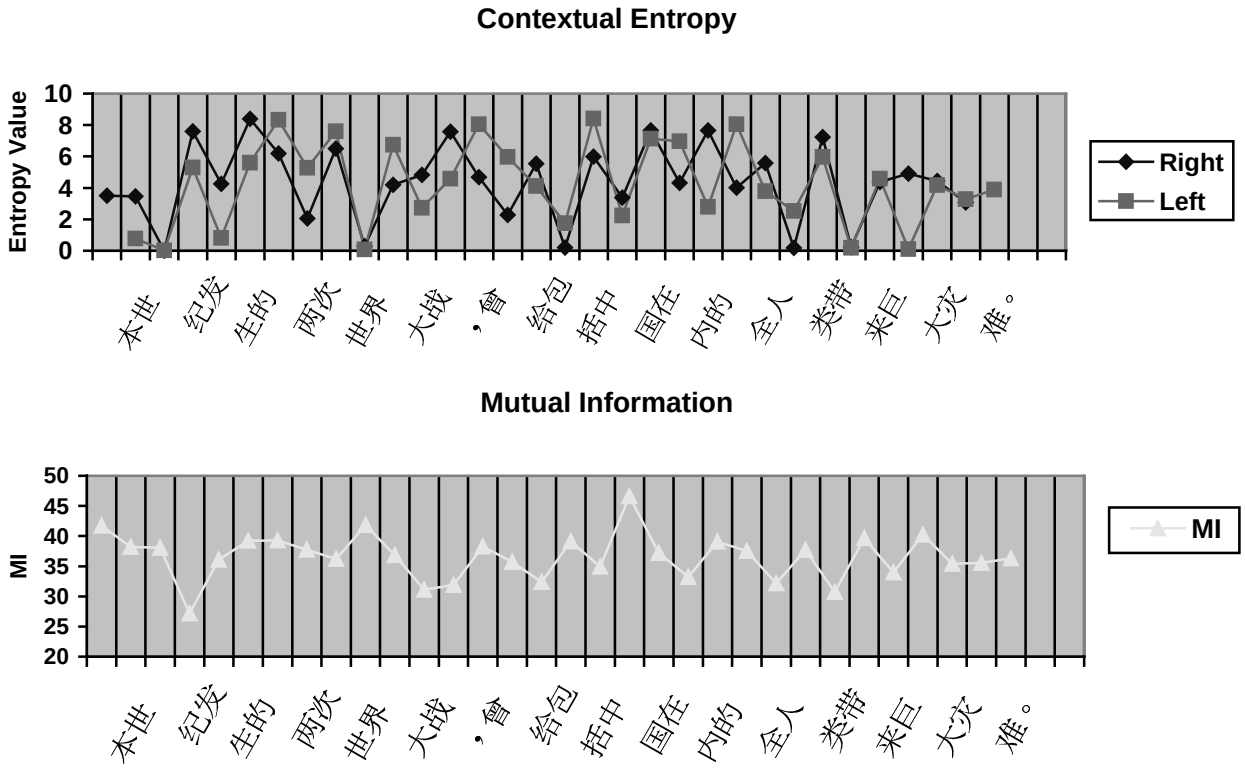
The information of a symbol w with respect to the model M and to a context c is defined by:

$$I(w \mid M, c) = -\log_2 p(w \mid M, c)$$

The entropy of a context c with respect to this model M is defined by:

$$H(M,c) = \sum_{w \in \Sigma} p(w \mid M,c) I(w \mid M,c)$$

This entropy measures the uncertainty about the next symbol after having seen the context c. We call it contextual entropy. It will be low if one particular symbol is expected to occur with a high probability. Otherwise it will high if the model has no "idea" what kind of symbol will follow the context.




Monitoring entropy in the figure above shows regions of high entropy correspond with word boundary. Given the left context, a word boundary will follow the context. Given the right context, a boundary is followed by the context. In other words, the beginning and the end of a boundary are often marked by high entropy as any symbol can follow a boundary and occur before a boundary.

## 3 Algorithm

To find Chinese words, We look for character sequences that are stable in the corpus. The components of a word are strongly correlated but appear in various contexts in the corpus. Contextual entropy among components of a word is low. High entropy appears at word boundaries.

We calculate both left and right contextual entropy values for each bigram occurring in the corpus.

$$H(x_1,x_2) = -\sum_{x_3 \in \Sigma} p(x_3 \mid x_1,x_2) \log_2 p(x_3 \mid x_1,x_2)$$

$$H(x_2,x_3) = -\sum_{x_1 \in \Sigma} p(x_1 \mid x_2,x_3) \log_2 p(x_1 \mid x_2,x_3)$$

We only store the positive contextual entropy value. An entropy of zero indicates there is no boundary before or after the context given the right or left context. We assume the value for the bigrams which do not appear in the corpus is zero as we can still predict the boundary according to the left and right context in which it appears. This can save a lot of space to store bigrams with zero value.

From the figure above we know that there is a word boundary at a rest for both entropy values. On the contrary there is no boundary at a trough. For a punctuation or a Chinese word marker, there is a rest preceding it given the right context and a rest following it given the left context. In other words, after having seen a punctuation or a word marker we do not know what occurs before and after it. This is very useful for detecting punctuations and word markers. Most other work did not treat the punctuation as a character (Peng and Schuurmans, 2001, Dai, et al, 1999) or could not detect word markers well based on statistical methods (Ge, et al, 1999). They treated punctuations as separators for sentences.

The process of segmenting the text is how to find the word boundaries. Across a word boundary there is a significant change in the contextual entropy. We apply the following algorithms to determine whether there is a word boundary between C and D for a string ABCDEF.

1. $LH_{BC} - LH_{AB} > h1$
2. $LH_{BC} - LH_{CD} > h2$
3. $RH_{DE} - RH_{EF} > h3$
4. $RH_{DE} - RH_{CD} > h4$

For each of word markers or punctuations, there is a boundary before or after it. We apply the following algorithms to detect a punctuation or marker C for a string ABCDE.

5. $LH_{BC} - LH_{AB} > h5$
6. $LH_{BC} - LH_{CD} > h6$
7. $RH_{CD} - RH_{DE} > h7$
8. $RH_{CD} - RH_{BC} > h8$

LH for left contextual entropy, RH for right contextual entropy. h1, h2, h3, h4, h5, h6, h7, h8 are the threshold value.

We try to test whether mutual information can help to improve the model, by capturing the dependency within a bigram. There is a low mutual information across a boundary. We use (9) (10) to test whether there is a minimum value at a boundary between C and D for a string ABCDE.

9. $MI_{BC} - MI_{CD} > m1$
10. $MI_{DE} - MI_{CD} > m2$

MI for mutual information, m1, m2 are the threshold value.

11. $LH_{BC} > h9$
12. $RH_{DE} > h10$
13. $LH_{BC} + RH_{DE} > h11$

For a boundary between BC and DE, the contextual entropy given left context BC or right context DE are very high. We try to test whether there is a threshold for boundaries and non-boundaries.

## 4 Experiment Results

We trained the bi-directional 2nd order Markov model on 220MB corpora mainly news from People Daily (91-95). We obtained about 1M pairs of bigrams with positive entropy. We stored the mutual information for the bigram at the same time.

In order to evaluate our algorithm, we used 100 articles of Penn Treebank Tagged Chinese Corpus including 325 articles from People Daily (94-98) to train the thresholds. Then we tested on the rest of the articles. We used recall and precision to measure our performance both on discovering word boundaries and words. A word is considered correctly segmented only if there is a word boundary in front of and at the end of the word and these is no boundary among the word. The following Table 1,2,3,4 show the testing result for our algorithms.

| | Boundaries | | Words | |
|---|---|---|---|---|
| | Precision | Recall | Precision | Recall |
| 1(h1=0) | 90.7% | 75.5% | 64.4% | 53.6% |
| 2(h2=0) | 89.3% | 72.6% | 54.6% | 44.4% |
| 3(h3=0) | 93.2% | 78.0% | 67.8% | 56.8% |
| 4(h4=0) | 88.0% | 71.5% | 52.9% | 43.0% |
| AND(1,2,3,4),h1,h2,h3,h4=0 | 98.8% | 35.2% | 46.6% | 16.6% |
| OR(1,2,3,4),h1,h2,h3,h4=0 | 82.9% | 96.4% | 66.6% | 77.4% |
| OR(1,2,3,4),h1,h2,h3,h4=1 | 87.4% | 94.0% | 72.3% | 77.7% |
| OR(1,2,3,4),h1,h2,h3,h4=2 | 91.0% | 91.8% | 76.3% | 76.9% |
| OR(1,2,3,4),h1,h2,h3,h4=3 | 93.5% | 83.0% | 74.0% | 65.7% |
| AND(1,2),h1,h2=0 | 95.4% | 73.3% | 67.2% | 51.7% |
| AND(3,4),h3,h4=0 | 97.0% | 75.3% | 70.7% | 54.8% |
| ***OR(AND(1,2),AND(3,4)),h=0*** | ***94.5%*** | ***88.7%*** | ***80.0%*** | ***75.0%*** |
| AND(1,2),h1,h2=1 | 96.6% | 65.1% | 65.4% | 44.0% |
| AND(3,4),h3,h4=1 | 98.6% | 65.3% | 68.0% | 45.0% |
| OR(AND(1,2),AND(3,4)),h=1 | 96.5% | 78.7% | 75.5% | 61.6% |

**Table 1 Testing results for according to Equation (1)(2)(3)(4)**

| | Boundaries | | Words | |
|---|---|---|---|---|
| | Precision | Recall | Precision | Recall |
| 9(m1=0) | 81.1% | 68.0% | 45.8% | 38.4% |
| 10(m2=0) | 84.5% | 68.7% | 51.0% | 41.5% |
| OR(9,10),m1,m2=0 | 78.8% | 85.9% | 54.1% | 59.0% |
| AND(9,10),m1,m2=0 | 90.4% | 50.8% | 41.9% | 23.6% |
| AND(9,10),m1,m2=1 | 92.7% | 42.2% | 38.6% | 17.6% |
| AND(9,10),m1,m2=2 | 94.4% | 34.5% | 33.9% | 12.4% |

**Table 2 Testing results for Equation (9)(10)**

| | Boundaries | | Words | |
|---|---|---|---|---|
| | Precision | Recall | Precision | Recall |
| AND(6,8)(h6,h8=0) | 88.1% | 48.1% | 50.4% | 27.5% |
| AND(5,7)(h5,h7=0) | 84.8% | 38.3% | 43.7% | 19.7% |
| AND(5,6,7,8)h5,h6,h7,h8=0 | 97.4% | 29.7% | 52.8% | 16.1% |

**Table 3 Testing results for Equation (5)(6)(7)(8)**

| | Boundaries | | Words | |
|---|---|---|---|---|
| | Precision | Recall | Precision | Recall |
| 11(h9=3) | 84.5% | 92.0% | 66.5% | 72.5% |
| 11(h9=4) | 90.3% | 86.1% | 72.6% | 69.2% |
| 12(h10=3) | 82.8% | 92.1% | 61.8% | 68.7% |
| 12(h10=4) | 90.1% | 86.0% | 70.2% | 67.1% |
| OR(11,12)(h9,h10=3) | 76.8% | 98.3% | 54.9% | 70.2% |
| OR(11,12)(h9,h10=4) | 85.1% | 95.7% | 68.9% | 77.4% |
| AND(11,12)(h9,h10=3) | 93.2% | 85.8% | 77.2% | 71.1% |
| AND(11,12)(h9,h10=4) | 97.4% | 76.4% | 76.9% | 60.3% |
| ***13(h11=7)*** | ***91.7%*** | ***91.3%*** | ***77.5%*** | ***77.7%*** |
| 13(h11=8) | 93.6% | 88.3% | 79.1% | 74.7% |

**Table 4 Testing results according to Equation (11)(12)(13)**

From Table 1 we know there is a significant change in contextual entropy across a word boundary. Either side of contextual entropy change is useful to detect the word boundary. If we use F-measure:

$$F = \frac{2 * p * r}{p + r}$$

as a testing metric, using a threshold value around 2 with an "OR" relationship among Eq.(1)(2)(3)(4) we achieve the best result for the testing corpus.

Table 2 shows there is not much change in mutual information cross a word boundary. Using mutual information alone is not enough to detect the word boundary for a 2nd order markov model. Table 3 shows (5)(6)(7)(8) properties are useful to detect a single character word marker in Chinese or punctuation. We obtained the highest precision under the four conditions. Table 4 shows using equation (13) sum of both left and right contextual entropy is better than either left Eq. (11) or right contextual entropy Eq. (12).

From the results above, the following conditions and thresholds we get the best results on the training corpus (100 articles):

1. OR(AND(1,2),AND(3,4)),h1,h2,h3,h4=0
2. 13(h11=9)
3. AND(5,6,7,8)h5,h6,h7,h8=0
4. AND(9,10),m1,m2=3

We obtained 93.2% precision with 93.1% recall on discovering word boundaries and 81.2% precision with 81.1% recall on discovering words. And we got 93.3% precision with 92.4% recall on discovering word boundaries and 81.3% precision with 80.4% recall on discovering words

Peng and Dale (2001) used successive EM phases to learn a probabilistic model of character sequences and pruned the model with a mutual information selection criterion. They achieved 75.1% precision with 74.0% recall on discovering words by repeatedly applying lexicon pruning to an improved EM training. Their results are tested on the same corpus as ours. Compared with their approaches, our approaches are simpler, faster and achieved better results.

We had the same errors as Peng and Dale (2001) mentioned. Most errors caused by our approaches are numbers. As in training corpus, numbers are written in full-width alphabetic number but in testing corpus numbers are written in Chinese character. We did not obtain enough statistics to predict the boundary. The other kind of errors is compound nouns. We segmented "开发区" as "开发/区". But note that there is no standard definition for Chinese words. It should be noted that there is poor agreement on word segmentation amongst human annotators and at least three relative widespread

conventions (China, Taiwan, Penn Treebank). Our results should be lower than those judged by hand (which can bias judgements) and tested on non-standard corpora. Although our approach only used a 2nd order Markov model, we still can find words longer than 2 characters as we only used our model to identify the word boundaries rather than words.

## 5 Conclusion

This paper describes a new approach for Chinese word segmentation based contextual entropy from an unsegmented corpus. Contextual entropy is used to capture the dependency with the both contexts in which a word occurs. We used a relative short order Markov model to train our model and tried to identify the word boundary rather than the word. Our approach is simple and fast, and although it is unsupervised it gives very competitive results.

## References


Andre Kempe, 1999, Experiments in unsupervised Entropy-Based Corpus Segmentation, Ninth Conference of the European Chapter of the Assocciation for Computational Linguistics' 99 Workshop, 12th June 1999, Bergen, Norway

Jason L. Hutchens and Michael Alder, 1998, Finding structure via compression. In D. Powers, ed., NeMLap3/CoNLL98, Sydney, Australia, 1998.

Zellig Harris, 1955, From phoneme to morpheme. Language, 31(2), 1955

Yu Bin Dai, C. Kgoo and T. Loh, 1999, A new statistical Formula for Chinese Text Segmentation Incorporating Contextual Information, SIGIR'99 Berkley, CA USA

Jian Zhang, Jianfeng Gao, Ming Zhou, 2000, "Extraction of Chinese compound words – an experimental study on a very large corpus". The second Chinese Language Processing Workshop attached to ACL2000, Hong Kong, October 8, 2000.

K. T. Lua and K. W Gan, 1994 An application of Information Theory in Chinese Word Segmentation, Computer Processing of Chinese & Oriental Languages, Vol. 8, no. 1:115-124

C. H Tung and H J Lee 1994, Identification of unknown words from a corpus. Computer Processing of Chinese & Oriental Languages, Vol. 8 (supplement).

R Sproat, C. Shih 1990, A statistical method for finding word boundaries in Chinese text. Computer Processing of Chinese & Oriental Languages, 4, 4.

W J Teahan, Y Wen, R McNab and I Witten 2000, A compression-based Algorithm for Chinese Word Segmentation, Computational Linguistics, 26, 3.

J Y Nie, W Y Jin and M L Hannan, 1994. A hybrid approach to unknown word detection and segmentation of Chinese. ICCC'94. Singapore.

Swen Bing and Yu Shiwen, 1999, A graded Approach for the Efficient Resolution of Chinese Word Segmentation Ambiguities, NLPPRS, Beijing, China.

Fuchun Peng and Dale Schuurmans 2001, Self-supervised Chinese Word Segmentation, In F. Hoffman et al. (Eds.): Advances in Intelligent Data Analysis, Proceedings of the Fourth International Conference (IDA-01), Cascais, Portugal.

Ponte, J and Croft, W.B 1996, USeg: a retargetable word segmentation procedure for information retrieval. In: Symposium on document analysis and information retrieval (SDAIR '96).

Chang, J. S, Lin Y. C. and su, K. Y. A 1995, Automatic construction of a Chinese Electronic Dictionary. Proceedings of the Third Workshop on Very Large Corpora.

C. L. Yeh and H. J. Lee, 1991, Rule-based word identification for mandarin Chinese sentences – a unification approach, Computer Processing of Chinese and Oriental Languages, Vol. 5, No. 2.

# Chinese Word Segmentation Based on Contextual Entropy

Mr. Jin Hu Huang
School of Informatics and Engineerng
Flinders University of South Australia
Adelaide, Australia, SA 5001
Jin.huang@ist.flinders.edu.au

A/Pro. David Powers
School of Informatics and Engineerng
Flinders University of South Australia
Adelaide, Australia, SA 5001
Powers@ist.flinders.edu.au