Chinese Word Segmentation Based on Contextual Entropy

Jin Hu Huang School of Informatics and Engineering Flinders University of South Australia GPO Box 2100, Adelaide South Australia 5001 Jin.huang@infoeng.flinders.edu.au **David Powers**

School of Informatics and Engineering Flinders University of South Australia GPO Box 2100, Adelaide South Australia 5001 powers@infoeng.flinders.edu.au

Abstract

Chinese is written without word delimiters so word segmentation is generally considered a key step in processing Chinese texts. This paper presents a new statistical approach to segment Chinese sequences into words 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 segment by finding the word boundaries instead of the words. 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, 1990,1996, , Teaban, Wen, McNab and Witten, 2000, Peng and Schuurmans, 2001) and lexicon-based approach (Yeh and Lee, 1991, Palmer, 1997, Cheng, Yong and Wong, 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, Wen, McNab and Witten, 2000), which is too expensive to construct at present. Rule-based approaches require a pre-defined word list (dictionary, or lexicon). The coverage of the dictionary is critical for these approaches. Many researches use a combination of approaches (Nie, Jin and Hanna 1994). These are supervised approaches that require extensive human involvement. Some (Sproat and Shih, 1990, de Marcken, 1996, Peng and Schuurmans, 2001) used unsupervised approaches and required little human intervention.

It has been long known that contextual information can be used for segmentation (Harris 1955). Dai, Kgoo and Loh (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, Lin & Su (1995) and Ponte & Croft (1996) used contextual entropy for automatic lexical acquisition. Hutchens & 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 given 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.

Contextual Entropy & Mutual Information



Fig. 1 Contextual Entropy and Mutual Information for "本世纪发生的两次世界大战,曾给包括中国在内的全人类带来巨大灾难。" "The two world wars happened this century had brought great disasters to human being including China."

Monitoring entropy in the figure 1 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.

Contextual entropy finds a left boundary if there is a high branching factor (perplexity & choice) to the left and a right boundary if there is a high branching factor.



3 Algorithm

3.1 Contextual Entropy

To find Chinese words we look for character sequences that are stable in the corpus in the sense that 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 or right adjacent context. This can save a lot of space to store bigrams with zero value.

From Figure 1 above we know that there is a word boundary at a peak for both entropy values. On the contrary there is no boundary at a trough. For a punctuation mark or a Chinese word marker, there is a peak preceding it given the right context and a peak following it given the left context. In other words, after having seen a punctuation mark or a word marker we do not know what occurs before and after it. This is very useful for detecting punctuation marks and word markers. Most other work did not treat the punctuation as an unknown character (Peng and Schuurmans, 2001, Dai, Khoo and Loh, 1999) or could not detect word markers well based on statistical methods (Ge, Pratt and Smyth, 1999). They treated punctuation marks or characters as separators for sentences.

In order to segment the text we simply need to find the word boundaries. Across a word boundary there is a significant change in the contextual entropy. We apply the following thresholds 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 word markers or punctuation mark, there is a boundary before and after it. We call these function characters and apply the following thresholds to determine if c is a function character in the string ABCDE.

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

where LH is the left contextual entropy, RH is the right contextual entropy. h1, h2, h3, h4, h5, h6, h7, h8 are the threshold values.

9. LH_{BC} >h9 10. RH_{DE} >h10 11. 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.

3.2 Mutual Information

The work by Sproat and Shih (1990) has a similar goal using a different measure, Mutual Information.

$$MI(x, y) = -\log_2 \frac{p(x, y)}{p(x)p(y)}$$

From Fig. 1 we know that there is a high mutual information between characters in a word and a low mutual information across a boundary. They found the pair of adjacent characters with mutual information greater than some threshold (2.5) is a word and grouped them together. They iterated it until there were no more characters to group.

We formulate this in our model as well and consider it on its own and in combination with Contextual Entropy. Instead of grouping characters together as a word we try to find the boundary between characters. We use (9) (10) to test whether there is a minimum value at a boundary between C and D for a string ABCDE.

 $\begin{array}{l} 12. \ MI_{CD} < m1 \\ 13. \ MI_{BC} - MI_{CD} > m2 \\ 14. \ MI_{DE} - MI_{CD} > m3 \end{array}$

where MI is the mutual information, m1, m2 are the threshold values.

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 validate variations on our algorithm, we used a small corpus 100 articles of 325 articles from People Daily (94-98) included in the Penn Treebank Tagged Chinese Corpus (3.3M) to set the thresholds h1 ... h11, m1 ... m3 and find the best way of combining these. 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			
	Recall	Precision	F-measure	Recall	Precision	F-measure	
1(h1=0)	75.8%	85.7%	80.5%	53.5%	60.5%	56.8%	
2(h2=0)	72.6%	84.8%	78.2%	43.5%	50.8%	46.9%	
3(h3=0)	73.0%	85.0%	78.6%	44.7%	52.1%	48.1%	
4(h4=0)	78.0%	87.5%	82.5%	56.2%	63.0%	59.4%	
AND(1,2,3,4),h1,h2,h3,h4=0	36.4%	96.0%	52.8%	17.5%	46.1%	25.3%	
OR(1,2,3,4),h1,h2,h3,h4=0	97.0%	77.1%	85.9%	75.7%	60.1%	67.0%	
OR(1,2,3,4),h1,h2,h3,h4=1	94.1%	82.5%	87.9%	76.1%	66.7%	71.1%	
OR(1,2,3,4),h1,h2,h3,h4=2	89.6%	87.0%	88.36%	72.3%	70.3%	71.3%	
OR(1,2,3,4),h1,h2,h3,h4=3	82.0%	90.7%	86.1%	63.9%	70.7%	67.1%	

AND(1,2),h1,h2=0	59.2%	90.0%	71.5%	29.2%	44.5%	35.3%
AND(1,2),h1,h2=1	48.5%	94.3%	64.1%	22.0%	42.9%	29.1%
AND(3,4),h3,h4=0	62.3%	93.3%	74.7%	33.8%	50.6%	40.5%
AND(3,4),h3,h4=1	49.8%	96.1%	65.6%	25.1%	48.4%	33.1%
OR(AND(1,2),AND(3,4)),h=0	85.0%	90.1%	87.5%	67.5%	71.5%	69.5%
OR(AND(1,2),AND(3,4)),h=1	71.5%	94.0%	81.2%	50.6%	66.5%	57.4%

Table 1 Validation results for ac	cording to Equation (1)(2)(3)(4)
-----------------------------------	----------------------------------

	Boundaries Recall Precision F-measure			Words		
				Recall	Precision	F-measure
AND(5,6,7,8)h5,h6,h7,h8=0	94.8%	29.9%	45.5%	16.5%	52.3%	25.1%
AND(5,6,7,8)h5,h6,h7,h8=1	99.0%	18.5%	31.3%	9.6%	51.5%	16.2%

Table 2 Validation results for Equation (5)(6)(7)(8)

	Boundaries			Words			
	Recall	Precision	F-measure	Recall	Precision	F-measure	
9(h9=3)	89.4%	79.4%	84.1%	66.0%	58.6%	62.1%	
9(h9=4)	79.1%	85.5%	82.2%	57.4%	62.1%	59.7%	
10(h10=3)	88.5%	79.0%	83.5%	63.1%	56.3%	59.5%	
10(h10=4)	82.0%	86.9%	84.4%	60.7%	64.3%	62.5%	
OR(9,10)(h9,h10=3)	98.4%	72.7%	83.6%	68.8%	50.8%	58.4%	
OR(9,10)(h9,h10=4)	94.5%	· 80.7%	87.1%	73.9%	63.1%	68.1%	
OR(9,10)(h9,h10=5)	86.6%	89.5%	88.0%	69.3%	71.6%	70.5%	
AND(9,10)(h9,h10=3)	79.5%	89.1%	84.0%	61.6%	69.1%	65.1%	
AND(9,10)(h9,h10=4)	66.6%	95.4%	78.5%	47.7%	68.3%	56.2%	
11(h11=6)	94.2%	82.2%	87.8%	75.2%	65.6%	70.1%	
11(h11=7)	90.2%	87.2%	88.7%	74.2%	71.7%	72.9%	
11(h11=8)	85.0%%	91.0%	87.9%	69.4%	74.3%	71.8%	

Table 3 Validation results according to Equation (9)(10)(11)

	Boundaries			Words			
	Recall	Precision	F-measure	Recall	Precision	F-measure	
12(m1=2)	70.1%	91.7%	79.5%	46.7%	61.1%	53.0%	
12(m1=3)	82.6%	88.9%	85.7%	62.8%	67.5%	65.1%	
12(m1=4)	90.0%	83.2%	86.5%	69.9%	64.7%	67.2%	
12(m1=5)	95.1%	77.6%	85.5%	71.6%	58.5%	64.4%	
13(m2=0)	84.7%	79.2%	81.8%	59.3%	55.4%	57.3%	
13(m2=1)	77.3%	82.6%	79.9%	51.9%	55.5%	53.7%	
14(m3=0)	89.8%	88.0%	88.9%	72.9%	71.5%	72.2%	
14(m3=1)	86.0%	91.4%	88.7%	69.4%	73.8%	71.5%	
OR(13,14),m2,m3=0	94.3%	78.2%	85.5%	70.5%	58.5%	63.9%	
OR(13,14),m2,m3=1	91.6%	82.6%	86.9%	70.2%	63.4%	66.6%	
OR(13,14),m2,m3=2	86.6%	86.7%	86.6%	66.3%	66.4%	66.3%	
AND(12,13),m1,m2=0	80.1%	90.6%	85.1%	61.2%	69.2%	65.0%	
AND(12,13),m1,m2=1	71.7%	93.4%	81.1%	50.4%	65.6%	57.0%	
AND(12,13),m1,m2=2	61.1%	95.4%	74.5%	37.0%	57.8%	45.2%	

Table 4 Validation results for Equation (12)(13)(14)

		Boundar	у	Word			
	Recall	Precision	F-measure	Recall	Precision	F-measure	
100 articles (Penn)	93.2%	93.1%	93.1%	81.1%	81.2%	81.1%	
225 articles (Penn)	92.4%	93.3%	92.3%	80.4%	81.3%	80.8%	
BJ corpus	91.5%	89.4	90.4%	76.8%	75.0%	75.9%	
			20.170	/0.0/0	1.0.070	13.970	

Table 5 Results on testing corpus

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 validation corpus.

Table 2 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 3 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). Table 4 shows the best threshold for grouping characters together is 4 for Penn Treebank corpus, greater than 2.5 that Sproat and Shih (1990) used in their work.

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

- 1. OR(AND(1,2),AND(3,4)),h1,h2,h3,h4=2
- 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. We tested on another corpus tagged by Beijing University from People Daily (Jan 1998, 8.8M). We obtained 89.4% precision with 91.5% recall on discovering word boundaries and 75.0% precision with 76.8% recall on discovering words.

Peng and Schuurmans (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 Schuurmans (2001) mentioned and had the same errors as most segmenters had to recognise the Chinese names. Most errors caused with our approaches relate to numbers and dates. In the training corpus, numbers written in full-width Arabic digits were replaced by a special character but in Penn corpus numbers are written in Chinese character. The other main kind of errors concerns 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 are as expected 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.

1 References

- Chang, J. S., Y. C Lin and K. Y. Su. 1995. Automatic construction of a Chinese electronic dictionary. Proceedings of the Third Workshop on Very Large Corpora.
- Dai, Y. B., C. Kgoo and T. Loh, 1999, A new statistical formula for Chinese text segmentation incorporating contextual information, SIGIR'99 Berkley, CA USA

de Marcken, C. 1996, Unsupervised Language Acquisition. Ph.D thesis, MIT.

- Ge, X., W. Pratt and P. Smyth. 1999. Discovering Chinese Words from Unsegmented Text. SIGIR 99. Berkeley.
- Harris, Z.S. 1955. From phoneme to morpheme. Language, 31(2), 1955
- Hutchens, J. and M. Alder, 1998, Finding structure via compression. In D. Powers, ed., NeMLap3/CoNLL98, Sydney, Australia, 1998.
- Kempe, A. 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
- Lua, K. T. 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
- Nie, J Y, W Y Jin and M L Hannan. 1994. A hybrid approach to unknown word detection and segmentation of Chinese. ICCC'94. Singapore.
- Palmer, D. 1997. A trainable rule-based algorithm for word segmentation. Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL97). Madrid.
- Peng, F. and D. 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 W. B. Croft. 1996. USeg: a retargetable word segmentation procedure for information retrieval. In: Symposium on document analysis and information retrieval (SDAIR '96).
- Sproat, R. and C. Shih. 1990. A statistical method for finding word boundaries in Chinese text. Computer Processing of Chinese & Oriental Languages, 4, 4.
- Sproat, R., C. Shih, W. Gale and N. Chang. 1996. A stochastic finite-state word-segmentation algorithm for Chinese. *Computational Linguistics*, 22(3).
- Teahan, W. J, Y. Wen, R. McNab and I. Witten 2000, A compression-based algorithm for Chinese word segmentation, *Computational Linguistics*, 26, 3.
- Tung, C. H. and H. J. Lee. 1994. Identification of unknown words from a corpus. Computer Processing of Chinese & Oriental Languages, Vol. 8 (supplement).
- Yeh, C. L. 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.
- Zhang, J., J. Gao and M. 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.