An Empirical Comparison of Goodness Measures for Unsupervised Chinese Word Segmentation with a Unified Framework

Hai Zhao*and Chunyu Kit
Department of Chinese, Translation and Linguistics,
City University of Hong Kong,
83 Tat Chee Avenue, Kowloon, Hong Kong, China
Email: haizhao@cityu.edu.hk, ctckit@cityu.edu.hk

Abstract

This paper reports our empirical evaluation and comparison of several popular goodness measures for unsupervised segmentation of Chinese texts using Bakeoff-3 data sets with a unified framework. Assuming no prior knowledge about Chinese, this framework relies on a goodness measure to identify word candidates from unlabeled texts and then applies a generalized decoding algorithm to find the optimal segmentation of a sentence into such candidates with the greatest sum of goodness scores. Experiments show that description length gain outperforms other measures because of its strength for identifying short words. Further performance improvement is also reported, achieved by proper candidate pruning and by assemble segmentation to integrate the strengths of individual measures.

1 Introduction

Unsupervised Chinese word segmentation was explored in a number of previous works for various purposes and by various methods (Ge et al., 1999; Fu and Wang, 1999; Peng and Schuurmans, 2001; SUN et al., 2004; Jin and Tanaka-Ishii, 2006). However, various heuristic rules are often involved in most existing works, and there has not been a comprehensive comparison of their performance in a unified way with available large-scale "gold standard" data sets, especially, multi-standard ones since Bakeoff-1 [1].

In this paper we will propose a unified framework for unsupervised segmentation of Chinese text. Four existing approaches to unsupervised segmentations or word extraction are considered as its special cases, each with its own goodness measurement to quantify word likelihood. The output by each approach will be evaluated using benchmark data sets of Bakeoff-3[2] (Levow, 2006). Note that unsupervised segmentation is different from, if not more complex than, word extraction, in that the former must carry out the segmentation task for a text, for which a segmentation (decoding) algorithm is indispensable, whereas the latter only acquires a word candidate list as output (Chang and Su, 1997; Zhang et al., 2000).

2 Generalized Framework

We propose a generalized framework to unify the existing methods for unsupervised segmentation, assuming the availability of a list of word candidates each associated with a goodness for how likely it is to be a true word. Let $W = \{\{w_i, g(w_i)\}_{i=1,...,n}\}$ be such a list, where w_i is a word candidate and $g(w_i)$

The research described in this paper was supported by the Research Grants Council of Hong Kong S.A.R., China, through the CERG grant 9040861 (CityU 1318/03H) and by City University of Hong Kong through the Strategic Research Grant 7002037. Dr. Hai Zhao was supported by a postdoctoral Research Fellowship in the Department of Chinese, Translation and Linguistics, City University of Hong Kong. Thanks four anonymous reviewers for their insightful comments!

[1] First International Chinese Word Segmentation Bakeoff, at http://www.sighan.org/bakeoff2003

[2] The Third International Chinese Language Processing Bakeoff, at http://www.sighan.org/bakeoff2006.

its goodness function.

Two generalized decoding algorithms, (1) and (2), are formulated for optimal segmentation of a given plain text. The first one, decoding algorithm (1), is a Viterbi-style one to search for the best segmentation S^* for a text T, as follows,

$$S^* = \underset{w_1 \cdots w_i \cdots w_n = T}{\operatorname{argmax}} \sum_{i=1}^{n} g(w_i), \qquad (1)$$

with all $\{w_i, g(w_i)\} \in W$.

Another algorithm, decoding algorithm (2), is a maximal-matching one with respect to a goodness score. It works on T to output the best current word w^* repeatedly with T=t^* for the next round as follows,

$$\{w^*, t^*\} = \underset{wt = T}{\operatorname{argmax}}\ g(w) \qquad (2)$$

with each $\{w, g(w)\} \in W$. This algorithm will back off to forward maximal matching algorithm if the goodness function is set to word length. Thus the former may be regarded as a generalization of the latter. Symmetrically, it has an inverse version that works the other way around.

3 Goodness Measurement

An unsupervised segmentation strategy has to rest on some predefined criterion, e.g., mutual information (MI), in order to recognize a substring in the text as a word. Sproat and Shih (1990) is an early investigation in this direction. In this study, we examine four types of goodness measurement for a candidate substring[3]. In principle, the higher goodness score for a candidate, the more possible it is to be a true word.

Frequency of Substring with Reduction A linear algorithm was proposed in (Lü et al., 2004) to produce a list of such reduced substrings for a given corpus. The basic idea is that if two partially overlapped n-grams have the same frequency in the input corpus, then the shorter one is discarded as a redundant word candidate. We take the logarithm of FSR as the goodness for a word candidate, i.e.,

$$g_{FSR}(w) = \log(\hat{p}(w)) \qquad (3)$$

where $\hat{p}(w)$ is w's frequency in the corpus. This allows the arithmetic addition in (1). According to Zipf's Law (Zipf, 1949), it approximates the use of the rank of w as its goodness, which would give it some statistical significance. For the sake of efficiency, only those substrings that occur more than once are considered qualified word candidates.

Description Length Gain (DLG) The goodness measure is proposed in (Kit and Wilks, 1999) for compression-based unsupervised segmentation. The DLG from extracting all occurrences of $x_i x_{i+1} ... x_j$ (also denoted as $x_{i..j}$) from a corpus $X = x_1 x_2 ... x_n$ as a word is defined as

$$DLG(x_{i..j}) = L(X) - L(X[r \rightarrow x_{i..j}] \oplus x_{i..j}) \quad (4)$$

where $X[r \rightarrow x_{i..j}]$ represents the resultant corpus from replacing all instances of $x_{i..j}$ with a new symbol r throughout X and $\oplus$ denotes the concatenation of two substrings. $L(\cdot)$ is the empirical description length of a corpus in bits that can be estimated by the Shannon-Fano code or Huffman code as below, following classic information theory (Shannon, 1948).

$$L(X) \doteq -|X| \sum_{x \in V} \hat{p}(x) \log_2 \hat{p}(x) \qquad (5)$$

where $|\cdot|$ denotes string length, V is the character vocabulary of X and $\hat{p}(x)$ x's frequency in X. For a given word candidate w, we define $g_{DLG}(w) = DLG(w)$. In principle, a substring with a negative DLG do not bring any positive compression effect by itself. Thus only substrings with a positive DLG value are added into our word candidate list.

Accessor Variety (AV) Feng et al. (2004) propose AV as a statistical criterion to measure how likely a substring is a word. It is reported to handle low-frequent words particularly well. The AV of a substring $x_{i..j}$ is defined as

$$AV(x_{i..j}) = \min\{L_{av}(x_{i..j}), R_{av}(x_{i..j})\} \qquad (6)$$

where the left and right accessor variety $L_{av}(x_{i..j})$ and $R_{av}(x_{i..j})$ are, respectively, the number of distinct predecessor and successor characters. For a similar reason as to FSR, the logarithm of AV is used

[3] Although there have been many existing works in this direction (Lua and Gan, 1994; Chien, 1997; Sun et al., 1998; Zhang et al., 2000; SUN et al., 2004), we have to skip the details of comparing MI due to the length limitation of this paper. However, our experiments with MI provide no evidence against the conclusions in this paper.

as goodness measure, and only substrings with AV > 1 are considered word candidates. That is, we have $g_{AV}(w) = \log AV(w)$ for a word candidate w.

Boundary Entropy (Branching Entropy, BE) It is proposed as a criterion for unsupervised segmentation in some existing works (Tung and Lee, 1994; Chang and Su, 1997; Huang and Powers, 2003; Jin and Tanaka-Ishii, 2006). The local entropy for a given $x_{i..j}$, defined as

$$h(x_{i..j}) = -\sum_{x \in V} p(x|x_{i..j}) \log\, p(x|x_{i..j}), \qquad (7)$$

indicates the average uncertainty after (or before) $x_{i..j}$ in the text, where $p(x|x_{i..j})$ is the co-occurrence probability for x and $x_{i..j}$. Two types of $h(x_{i..j})$, namely $h_L(x_{i..j})$ and $h_R(x_{i..j})$, can be defined for the two directions to extend $x_{i..j}$ (Tung and Lee, 1994). Also, we can define $h_{min} = \min\{h_R, h_L\}$ in a similar way as in (6). In this study, only substrings with BE > 0 are considered word candidates. For a candidate w, we have $g_{BE}(w) = h_{min}(w)$[4].

4 Evaluation

The evaluation is conducted with all four corpora from Bakeoff-3 (Levow, 2006), as summarized in Table 1 with corpus size in number of characters. For unsupervised segmentation, the annotation in the training corpora is not used. Instead, they are used for our evaluation, for they are large and thus provide more reliable statistics than small ones. Segmentation performance is evaluated by word F-measure $F = 2RP/(R+P)$. The recall R and precision P are, respectively, the proportions of the correctly segmented words to all words in the gold-standard and a segmenter's output[5].

Note that a decoding algorithm always requires the goodness score of a single-character candidate for computation. There are two ways to get this score: (1) computed by the goodness measure, which is applicable only if the measure allows; (2) set to zero as default value, which is always applicable even to single-character candidates not in the word candidate list in use. For example, all single-character candidates given up by DLG because of their negative DLG scores will have a default value during decoding. We will use a '/d' to indicate experiments using such a default value.

Table 1: Bakeoff-3 Corpora

Corpus	AS	CityU	CTB	MSRA
Training(M)	8.42	2.71	0.83	2.17
Test(K)	146	364	256	173

Table 2: Performance with decoding algorithm (1)

M. L.[a]	Good-ness	Training corpus			
		AS	CityU	CTB	MSRA
2	FSR	.400	.454	.462	.432
	DLG/d	**.592**	**.610**	**.604**	**.603**
	AV	.568	.595	.596	.577
	BE	.559	.587	.592	.572
7	FSR	.193	.251	.268	.235
	DLG/d	.331	.397	.409	.379
	AV	**.399**	**.423**	**.430**	**.407**
	BE	.390	.419	.428	.403

[a]M.L.: Maximal length allowable for word candidates.

4.1 Comparison

We apply the decoding algorithm (1) to segment all Bakeoff-3 corpora with the above goodness measures. Both word candidates and goodness values are derived from the raw text of each training corpus. The performance of these measures is presented in Table 2. From the table we can see that DLG and FSR have the strongest and the weakest performance, respectively, whereas AV and BE are highly comparable to each other.

Decoding algorithm (2) runs the forward and backward segmentation with the respective AV and BE criteria, i.e., L_{AV}/h_L for backward and R_{AV}/h_R forward, and the output is the union of two segmentations [6]. A performance comparison of AV and BE with both algorithms (1) and (2) is presented in Table 3. We can see that the former has a rela-

[4]Both AV and BE share a similar idea from Harris (1970): If the uncertainty of successive token increases, then it is likely to be at a boundary. In this sense, one may consider them the discrete and continuous formulation of the same idea.

[5]All evaluations will be represented in terms of word F-measure if not otherwise specified. A standard scoring tool with this metric can be found in SIGHAN website, http://www.sighan.org/bakeoff2003/score. However, to compare with related work, we will also adopt boundary F-measure $F_b = 2R_bP_b/(R_b+P_b)$, where the boundary recall R_b and boundary precision P_b are, respectively, the proportions of the correctly recognized boundaries to all boundaries in the gold-standard and a segmenter's output (Ando and Lee, 2000).

[6]That is, all segmented points by either segmentation will be accounted into the final segmentation.

Table 3: Performance comparison: AV vs. BE

M. L.	Goodness	Training corpus AS	CityU	CTB	MSRA
2	$AV_{(1)}$	**.568**	**.595**	**.596**	**.577**
	$AV_{(2)}$/d	.485	.489	.508	.471
	$AV_{(2)}$	.445	.366	.367	.387
	$BE_{(1)}$	**.559**	**.587**	**.592**	**.572**
	$BE_{(2)}$/d	.485	.489	.508	.471
	$BE_{(2)}$	.504	.428	.446	.446
7	$AV_{(1)}$	.399	.423	.430	.407
	$AV_{(2)}$/d	**.570**	**.581**	**.588**	**.572**
	$AV_{(2)}$	.445	.366	.368	.387
	$BE_{(1)}$	.390	.419	.428	.403
	$BE_{(2)}$/d	**.597**	**.604**	**.605**	**.593**
	$BE_{(2)}$	.508	.431	.449	.446

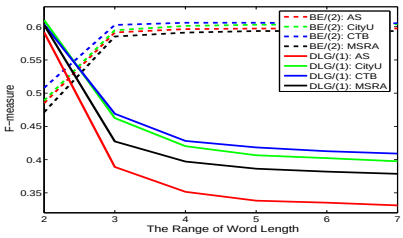

Figure 1: Performance vs. word length

tively better performance on shorter words and the latter outperforms on longer ones.

How segmentation performance varies along with word length is exemplified with DLG and BE as examples in Figure 1, with (1) and (2) indicating a respective decoding algorithm in use. It shows that DLG outperforms on two-character words and BE on longer ones.

4.2 Word Candidate Pruning

Up to now, word candidates are determined by the default goodness threshold 0. The number of them for each of the four goodness measures is presented in Table 4. We can see that FSR generates the largest set of word candidates and DLG the smallest. More interestingly or even surprising, AV and BE generate exactly the same candidate list for all corpora.

In addition to word length, another crucial factor to affect segmentation performance is the quality of the word candidates as a whole. Since each candidate is associated with a goodness score to indicate how good it is, a straightforward way to ensure, and further enhance, the overall quality of a candidate set is to prune off those with low goodness scores.

Table 4: Word candidate number by threshold 0

Goodness	Training Corpus AS	CityU	CTB	MSRA
FSR	2,009K	832K	294K	661K
DLG	543K	265K	96K	232K
AV	1,153K	443K	160K	337K
BE	1,153K	443K	160K	337K

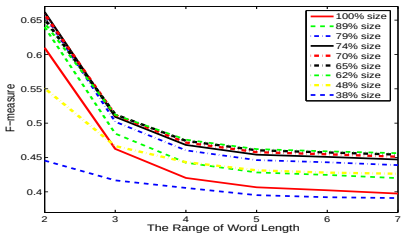

Figure 2: Performance by candidate pruning: DLG

To examine how segmentation performance changes along with word candidate pruning and decide the optimal pruning rate, we conduct a series of experiments with each goodness measurements. Figures 2 and 3 present, as an illustration, the outcomes of two series of our experiments with DLG by decoding algorithm (1) and BE by decoding algorithm (1) and (2) on CityU training corpus. We find that appropriate pruning does lead to significant performance improvement and that both DLG and BE keep their superior performance respectively on two-character words and others. We also observe that each goodness measure has a stable and similar performance in a range of pruning rates around the optimal one, e.g., 79-62% around 70% in Figure 2.

The optimal pruning rates found through our experiments for the four goodness measures are given in Table 5, and their correspondent segmentation performance in Table 6. These results show a remarkable performance improvement beyond the de-

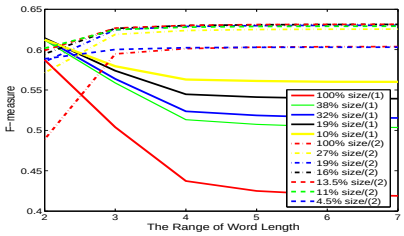

Figure 3: Performance by candidate pruning: BE

Table 5: Optimal rates for candidate pruning (%)

Decoding algorithm	Goodness measure			
	FSR	DLG	AV	BE
(1)	1.8	70	12.5	20
(2)	–	–	8	12.5

Table 6: Performance via optimal candidate pruning

M. L.	Goodness	Training corpus			
		AS	CityU	CTB	MSRA
2	$FSR_{(1)}$	.501	.525	.513	.522
	$DLG_{(1)}$/d	**.710**	**.650**	**.664**	**.638**
	$AV_{(1)}$	.616	.625	.609	.618
	$BE_{(1)}$	.613	.614	.605	.611
	$AV_{(2)}$/d	.585	.602	.589	.599
	$BE_{(2)}$/d	.591	.599	.596	.593
7	$FSR_{(1)}$	.444	.491	.486	.486
	$DLG_{(1)}$/d	.420	.447	.460	.423
	$AV_{(1)}$	.517	.568	.549	.544
	$BE_{(1)}$	.501	.539	.510	.519
	$AV_{(2)}$/d	.623	.624	.604	.615
	$BE_{(2)}$/d	**.630**	**.631**	**.620**	**.622**

fault threshold setting. What remains unchanged is the advantage of DLG for two-character words and that of AV/BE for longer words. However, DLG achieves the best overall performance among the four, although it uses only single- and two-character word candidates. The overwhelming number of two-character words in Chinese allows it to triumph.

4.3 Ensemble Segmentation

Although proper pruning of word candidates brings amazing performance improvement, it is unlikely for one to determine an optimal pruning rate in practice for an unlabeled corpus. Here we put forth a parameter-free method to tackle this problem with the aids of all available goodness measures.

The first step of this method to do is to derive an optimal set of word candidates from the input. We have shown above that quality candidates play a critical role in achieving quality segmentation. Without any better goodness criterion available, the best we can opt for is the intersection of all word candidate lists generated by available goodness measures with the default threshold. A good reason for this is that the agreement of them can give a more reliable decision than any individual one of them. In fact, we only need DLG and AV/BE to get this intersection, because AV and BE give the same word candidates

Table 7: Performances of ensemble segmentation

M. L.	Goodness	Training corpus			
		AS	CityU	CTB	MSRA
2	$FSR_{(1)}$	.629	.635	.624	.623
	$DLG_{(1)}$/d	**.664**	**.653**	**.643**	**.650**
	$AV_{(1)}$	.641	.644	.631	.634
	$BE_{(1)}$	.640	.643	.632	.634
7	$AV_{(2)}$/d	.595	.637	.624	.610
	$BE_{(2)}$/d	.593	.635	.620	.609
$DLG_{(1)}$/d+$AV_{(2)}$/d		**.672**	**.684**	**.663**	**.665**
$DLG_{(1)}$/d+$BE_{(2)}$/d		.660	.681	.656	.653

and DLG generates only a subset of what FSR does.

The next step is to use this intersection set of word candidates to perform optimal segmentation with each goodness measures, to see if any further improvement can be achieved. The best results are given in Table 7, showing that decoding algorithm (1) achieves marvelous improvement using short word candidates with all other goodness measures than DLG. Interestingly, DLG still remains at the top by performance despite of some slip-back.

To explore further improvement, we also try to combine the strengths of DLG and AV/BE respectively for recognizing two- and multi-character word. Our strategy to combine them together is to enforce the multi-character words in AV/BE segmentation upon the correspondent parts of DLG segmentation. This ensemble method gives a better overall performance than all others that we have tried so far, as presented at the bottom of Table 7.

4.4 Yet Another Decoding Algorithm

Jin and Tanaka-Ishii (2006) give an unsupervised segmentation criterion, henceforth referred to as decoding algorithm (3), to work with BE. It works as follows: if $g(x_{i..j+1}) > g(x_{i..j})$ for any two overlapped substrings $x_{i..j}$ and $x_{i..j+1}$, then a segmenting point should be located right after $x_{i..j+1}$. This algorithm has a forward and a backward version. The union of the segmentation outputs by both versions is taken as the final output of the algorithm, in exactly the same way as how decoding algorithm (2) works[7]. This algorithm is evaluated in (Jin and Tanaka-Ishii, 2006) using Peking University (PKU)

[7]Three segmentation criteria are given in (Jin and Tanaka-Ishii, 2006), among which the entropy increase criterion, namely, decoding algorithm (3), proves to be the best. Here we would like to thank JIN Zhihui and Prof. Kumiko Tanaka-Ishii for presenting the details of their algorithms.

Table 8: Performance comparison by word and boundary F-measure on PKU corpus (M. L. = 6)

	Good-ness	Decoding algorithm					
		(1)/d	(1)	(2)/d	(2)	(3)/d	(3)
F	AV	.313	.325	.588	.373	.376	.453
	AV*	.372	.372	**.663**	.663	.445	.445
	BE	.309	.319	.624	.501	.376	.624
	BE*	.370	.370	**.676**	.676	.447	.447
F_b	AV	.695	.700	.830	.762	.762	.728
	AV*	.728	.728	**.865**	.865	.783	.783
	BE	.696	.699	.849	.810	.762	.837[a]
	BE*	.728	.728	**.872**	.872	.784	.784

[a]With the same hyperparameters, (Jin and Tanaka-Ishii, 2006) report their best result of boundary precision 0.88 and boundary recall 0.79, equal to boundary F-measure 0.833.

Corpus of 1.1M words[8] as gold standard with a word candidate list extracted from the 200M Contemporary Chinese Corpus that mostly consists of several years of Peoples' Daily[9]. Here, we carry out evaluation with similar data: we extract word candidates from the unlabeled texts of People's Daily (1993 - 1997), of 213M and about 100M characters, in terms of the AV and BE criteria, yielding a list of 4.42 million candidates up to 6-character long[10] for each criterion. Then, the evaluation of the three decoding algorithms is performed on PKU corpus.

The evaluation results with both word and boundary F-measure are presented for the same segmentation outputs in Table 8, with "*" to indicate candidate pruning by DLG > 0 as reported before. Note that boundary F-measure gives much more higher score than word F-measure for the same segmentation output. However, in either of metric, we can find no evidence in favor of decoding algorithm (3). Undesirably, this algorithm does not guarantee a stable performance improvement with the BE measure through candidate pruning.

4.5 Comparison against Supervised Segmentation

Huang and Zhao (2007) provide empirical evidence to estimate the degree to which the four segmentation standards involved in the Bakeoff-3 differ from each other. As quoted in Table 9, a consistency rate

[8]http://icl.pku.edu.cn/icl_groups/corpus/dwldform1.asp

[9]http://ccl.pku.edu.cn:8080/ccl_corpus/jsearch/index.jsp

[10]This is to keep consistence with (Jin and Tanaka-Ishii, 2006), where 6 is set as the maximum n-gram length.

Table 9: Consistency rate among Bakeoff-3 segmentation standards (Huang and Zhao, 2007)

Test corpus	Training corpus			
	AS	CityU	CTB	MSRA
AS	1.000	0.926	0.959	0.858
CityU	0.932	1.000	0.935	0.849
CTB	0.942	0.910	1.000	0.877
MSRA	0.857	0.848	0.887	1.000

beyond 84.8% is found among the four standards. If we do not over-expect unsupervised segmentation to achieve beyond what these standards agree with each other, it is reasonable to take this figure as the topline for evaluation. On the other hand, Zhao et al. (2006) show that the words of 1 to 2 characters long account for 95% of all words in Chinese texts, and single-character words alone for about 50%. Thus, we can take the result of the brute-force guess of every single character as a word as a baseline.

To compare to supervised segmentation, which usually involves training using an annotated training corpus and, then, evaluation using test corpus, we carry out unsupervised segmentation in a comparable manner. For each data track, we first extract word candidates from both the training and test corpora, all unannotated, and then evaluate the unsupervised segmentation with reference to the gold-standard segmentation of the test corpus. The results are presented in Table 10, together with best and worst official results of the Bakeoff closed test. This comparison shows that unsupervised segmentation cannot compete against supervised segmentation in terms of performance. However, the experiments generate positive results that the best combination of the four goodness measures can achieve an F-measure in the range of 0.65-0.7 on all test corpora in use without using any prior knowledge, but extracting word candidates from the unlabeled training and test corpora in terms of their goodness scores.

5 Discussion: How Things Happen

Note that DLG criterion is to perform segmentation with the intension to maximize the compression effect, which is a global effect through the text. Thus it works well incorporated with a probability maximization framework, where high frequent but independent substrings are effectively extracted and re-

Table 10: Comparison of performances against supervised segmentation

Type		Test corpus			
		AS	CityU	CTB	MSRA
Baseline		.389	.345	.337	.353
2	$DLG_{(1)}/d$	.597	.616	.601	.602
	$DLG^{*}_{(1)}/d$	.655	.659	.632	.655
	$AV_{(1)}$	.577	.603	.597	.583
	$AV^{*}_{(1)}$	.630	.650	.618	.638
	$BE_{(1)}$	.570	.598	.594	.580
	$BE^{*}_{(1)}$	.629	.649	.618	.638
7	$AV_{(2)}/d$	.512	.551	.543	.526
	$AV^{*}_{(2)}/d$	.591	.644	.618	.604
	$BE_{(2)}/d$	.518	.554	.546	.533
	$BE^{*}_{(2)}/d$	.587	.641	.614	.605
$DLG^{*}_{(1)}/d + AV^{*}_{(2)}/d$		**.663**	**.692**	**.658**	**.667**
$DLG^{*}_{(1)}/d + BE^{*}_{(2)}/d$		.650	.689	.650	.656
Worst closed		.710	.589	0.818	.819
Best closed		.958	.972	0.933	.963

combined. We know that most unsupervised segmentation criteria will bring up long word bias problem, so does DLG measure. This explains why it gives the worse results as long candidates are added.

As for AV and BE measures, both of them give the metric of the uncertainty before or after the current substring. This means that they are more concerned with local uncertainty information near the current substring, instead of global information among the whole text as DLG. Thus local greedy search in maximal matching style is more suitable for these two measures than Viterbi search.

Our empirical results about word candidate list with default threshold 0, where the same list is from AV and BE, give another proof that both AV and BE reflect the same uncertainty. The only difference is behind the fact that the former and the latter is in the discrete and continuous formulation, respectively.

6 Conclusion and Future Work

This paper reported our empirical comparison of a number of goodness measures for unsupervised segmentation of Chinese texts with the aid two generalized decoding algorithms. We learn no previous work by others for a similar attempt. The comparison is carried out with Bakeoff-3 data sets, showing that all goodness measures exhibit their strengths for recognizing words of different lengths and achieve a performance far beyond the baseline. Among them, DLG with decoding algorithm (1) can achieve the best segmentation performance for single- and two-character words identification and the best overall performance as well. Our experiments also show that the quality of word candidates plays a critical role in ensuring segmentation performance [11]. Proper pruning of candidates with low goodness scores to enhance this quality enhances the segmentation performance significantly. Also, the success of unsupervised segmentation depends strongly on an appropriate decoding algorithm. Generally, Viterbi-style decoding produces better results than best-first maximal-matching. But the latter is not shy from exhibiting its particular strength for identifying multi-character words.

Finally, the ensemble segmentation we put forth to combine the strengths of different goodness measures proves to be a remarkable success. It achieves an impressive performance improvement on top of individual goodness measures.

As for future work, it would be natural for researchers to enhance supervised learning for Chinese word segmentation with goodness measures introduced here. There does be two successful examples in our existing work (Zhao and Kit, 2007). This is still an ongoing work.

References

Rie Kubota Ando and Lillian Lee. 2000. Mostly-unsupervised statistical segmentation of Japanese: Applications to kanji. In *Proceedings of the first Conference on North American Chapter of the Association for Computational Linguistics and the 6th Conference on Applied Natural Language Processing*, pages 241–248, Seattle, Washington, April 30.

Jing-Shin Chang and Keh-Yih Su. 1997. An unsupervised iterative method for Chinese new lexicon extraction. *Computational Linguistics and Chinese Language Processing*, 2(2):97–148.

Lee-Feng Chien. 1997. PAT-tree-based keyword extraction for Chinese information retrieval. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–58, Philadelphia.

Haodi Feng, Kang Chen, Xiaotie Deng, and Weimin Zheng. 2004. Accessor variety criteria for Chinese word extraction. *Computational Linguistics*, 30(1):75–93.

[11] This observation is shared by other researchers, e.g., (Peng et al., 2002).

Guo-Hong Fu and Xiao-Long Wang. 1999. Unsupervised Chinese word segmentation and unknown word identification. In *5th Natural Language Processing Pacific Rim Symposium 1999 (NLPRS'99), "Closing the Millennium"*, pages 32–37, Beijing, China, November 5-7.

Xianping Ge, Wanda Pratt, and Padhraic Smyth. 1999. Discovering Chinese words from unsegmented text. In *SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 271–272, Berkeley, CA, USA, August 15-19. ACM.

Zellig Sabbetai Harris. 1970. Morpheme boundaries within words. In *Papers in Structural and Transformational Linguistics*, page 68 ‐ 77.

Jin Hu Huang and David Powers. 2003. Chinese word segmentation based on contextual entropy. In Dong Hong Ji and Kim-Ten Lua, editors, *Proceedings of the 17th Asian Pacific Conference on Language, Information and Computation*, pages 152–158, Sentosa, Singapore, October, 1-3. COLIPS Publication.

Chang-Ning Huang and Hai Zhao. 2007. Chinese word segmentation: A decade review. *Journal of Chinese Information Processing*, 21(3):8–20.

Zhihui Jin and Kumiko Tanaka-Ishii. 2006. Unsupervised segmentation of Chinese text by use of branching entropy. In *COLING/ACL 2006*, pages 428–435, Sidney, Australia.

Chunyu Kit and Yorick Wilks. 1999. Unsupervised learning of word boundary with description length gain. In M. Osborne and E. T. K. Sang, editors, *CoNLL-99*, pages 1–6, Bergen, Norway.

Gina-Anne Levow. 2006. The third international Chinese language processing bakeoff: Word segmentation and named entity recognition. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 108–117, Sydney, Australia, July.

Xueqiang Lü, Le Zhang, and Junfeng Hu. 2004. Statistical substring reduction in linear time. In Keh-Yih Su et al., editor, *Proceeding of the 1st International Joint Conference on Natural Language Processing (IJCNLP-2004)*, volume 3248 of *Lecture Notes in Computer Science*, pages 320–327, Sanya City, Hainan Island, China, March 22-24. Springer.

Kim-Teng Lua and Kok-Wee Gan. 1994. An application of information theory in Chinese word segmentation. *Computer Processing of Chinese and Oriental Languages*, 8(1):115–123.

Fuchun Peng and Dale Schuurmans. 2001. Self-supervised Chinese word segmentation. In *The Fourth International Symposium on Intelligent Data Analysis*, pages 238–247, Lisbon, Portugal, September, 13-15.

Fuchun Peng, Xiangji Huang, Dale Schuurmans, Nick Cercone, and Stephen Robertson. 2002. Using self-supervised word segmentation in Chinese information retrieval. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 349–350, Tampere, Finland, August, 11-15.

Claude E. Shannon. 1948. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656, July, October.

Richard Sproat and Chilin Shih. 1990. A statistical method for finding word boundaries in Chinese text. *Computer Processing of Chinese and Oriental Languages*, 4(4):336–351.

Maosong Sun, Dayang Shen, and Benjamin K. Tsou. 1998. Chinese word segmentation without using lexicon and hand-crafted training data. In *COLING-ACL '98, 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, volume 2, pages 1265–1271, Montreal, Quebec, Canada.

Mao Song SUN, Ming XIAO, and Benjamin K. Tsou. 2004. Chinese word segmentation without using dictionary based on unsupervised learning strategy (in Chinese) (基于无指导学习策略的无词表条件下的汉语自动分词). *Chinese Journal of Computers*, 27(6):736–742.

Cheng-Huang Tung and His-Jian Lee. 1994. Identification of unknown words from corpus. *Computational Proceedings of Chinese and Oriental Languages*, 8:131–145.

Jian Zhang, Jianfeng Gao, and Ming Zhou. 2000. Extraction of Chinese compound words – an experimental study on a very large corpus. In *Proceedings of the Second Chinese Language Processing Workshop*, pages 132–139, Hong Kong, China.

Hai Zhao and Chunyu Kit. 2007. Incorporating global information into supervised learning for Chinese word segmentation. In *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics*, pages 66–74, Melbourne, Australia, September 19-21.

Hai Zhao, Chang-Ning Huang, Mu Li, and Bao-Liang Lu. 2006. Effective tag set selection in Chinese word segmentation via conditional random field modeling. In *Proceedings of the 20th Asian Pacific Conference on Language, Information and Computation*, pages 87–94, Wuhan, China, November 1-3.

George Kingsley Zipf. 1949. *Human Behavior and the Principle of Least Effort*. Addison-Wesley, Cambridge, MA.