Chinese Segmentation and New Word Detection using Conditional Random Fields

Fuchun Peng, Fangfang Feng, Andrew McCallum
Computer Science Department, University of Massachusetts Amherst
140 Governors Drive, Amherst, MA, U.S.A. 01003
{fuchun, feng, mccallum}@cs.umass.edu

Abstract

Chinese word segmentation is a difficult, important and widely-studied sequence modeling problem. This paper demonstrates the ability of linear-chain conditional random fields (CRFs) to perform robust and accurate Chinese word segmentation by providing a principled framework that easily supports the integration of domain knowledge in the form of multiple lexicons of characters and words. We also present a probabilistic new word detection method, which further improves performance. Our system is evaluated on four datasets used in a recent comprehensive Chinese word segmentation competition. State-of-the-art performance is obtained.

1 Introduction

Unlike English and other western languages, many Asian languages such as Chinese, Japanese, and Thai, do not delimit words by white-space. Word segmentation is therefore a key precursor for language processing tasks in these languages. For Chinese, there has been significant research on finding word boundaries in unsegmented sequences (see (Sproat and Shih, 2002) for a review). Unfortunately, building a Chinese word segmentation system is complicated by the fact that there is no standard definition of word boundaries in Chinese.

Approaches to Chinese segmentation fall roughly into two categories: heuristic dictionary-based methods and statistical machine learning methods. In dictionary-based methods, a predefined dictionary is used along with hand-generated rules for segmenting input sequence (Wu, 1999). However these approaches have been limited by the impossibility of creating a lexicon that includes all possible Chinese words and by the lack of robust statistical inference in the rules. Machine learning approaches are more desirable and have been successful in both unsupervised learning (Peng and Schuurmans, 2001) and supervised learning (Teahan et al., 2000).

Many current approaches suffer from either lack of exact inference over sequences or difficulty in incorporating domain knowledge effectively into segmentation. Domain knowledge is either not used, used in a limited way, or used in a complicated way spread across different components. For example, the N-gram generative language modeling based approach of Teahan et al (2000) does not use domain knowledge. Gao et al (2003) uses class-based language for word segmentation where some word category information can be incorporated. Zhang et al (2003) use a hierarchical hidden Markov Model to incorporate lexical knowledge. A recent advance in this area is Xue (2003), in which the author uses a sliding-window maximum entropy classifier to tag Chinese characters into one of four position tags, and then covert these tags into a segmentation using rules. Maximum entropy models give tremendous flexibility to incorporate arbitrary features. However, a traditional maximum entropy tagger, as used in Xue (2003), labels characters without considering dependencies among the predicted segmentation labels that is inherent in the state transitions of finite-state sequence models.

Linear-chain conditional random fields (CRFs) (Lafferty et al., 2001) are models that address both issues above. Unlike heuristic methods, they are principled probabilistic finite state models on which exact inference over sequences can be efficiently performed. Unlike generative N-gram or hidden Markov models, they have the ability to straightforwardly combine rich domain knowledge, for example in this paper, in the form of multiple readily-available lexicons. Furthermore, they are discriminatively-trained, and are often more accurate than generative models, even with the same features. In their most general form, CRFs are arbitrary undirected graphical models trained to maximize the conditional probability of the desired outputs given the corresponding inputs. In the linear-chain special case we use here, they can be roughly understood as discriminatively-trained hidden Markov models with next-state transition functions represented by exponential models (as in maximum en-

tropy classifiers), and with great flexibility to view the observation sequence in terms of arbitrary, overlapping features, with long-range dependencies, and at multiple levels of granularity. These beneficial properties suggests that CRFs are a promising approach for Chinese word segmentation.

New word detection is one of the most important problems in Chinese information processing. Many machine learning approaches have been proposed (Chen and Bai, 1998; Wu and Jiang, 2000; Nie et al., 1995). New word detection is normally considered as a separate process from segmentation. However, integrating them would benefit both segmentation and new word detection. CRFs provide a convenient framework for doing this. They can produce not only a segmentation, but also confidence in local segmentation decisions, which can be used to find new, unfamiliar character sequences surrounded by high-confidence segmentations. Thus, our new word detection is not a stand-alone process, but an integral part of segmentation. Newly detected words are re-incorporated into our word lexicon, and used to improve segmentation. Improved segmentation can then be further used to improve new word detection.

Comparing Chinese word segmentation accuracy across systems can be difficult because many research papers use different data sets and different ground-rules. Some published results claim 98% or 99% segmentation precision and recall, but these either count only the words that occur in the lexicon, or use unrealistically simple data, lexicons that have extremely small (or artificially non-existant) out-of-vocabulary rates, short sentences or many numbers. A recent Chinese word segmentation competition (Sproat and Emerson, 2003) has made comparisons easier. The competition provided four datasets with significantly different segmentation guidelines, and consistent train-test splits. The performance of participating system varies significantly across different datasets. Our system achieves top performance in two of the runs, and a state-of-the-art performance on average. This indicates that CRFs are a viable model for robust Chinese word segmentation.

2 Conditional Random Fields

Conditional random fields (CRFs) are undirected graphical models trained to maximize a conditional probability (Lafferty et al., 2001). A common special-case graph structure is a linear chain, which corresponds to a finite state machine, and is suitable for sequence labeling. A linear-chain CRF with parameters $\Lambda = \{\lambda_1, ...\}$ defines a conditional probability for a state (label) sequence $\mathbf{y} = y_1...y_T$ (for example, labels indicating where words start or have their interior) given an input sequence $\mathbf{x} = x_1...x_T$ (for example, the characters of a Chinese sentence) to be

$$P_\Lambda(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_\mathbf{x}} \exp\left(\sum_{t=1}^{T}\sum_{k}\lambda_k f_k(y_{t-1}, y_t, \mathbf{x}, t)\right), \quad (1)$$

where $Z_\mathbf{x}$ is the per-input normalization that makes the probability of all state sequences sum to one; $f_k(y_{t-1}, y_t, \mathbf{x}, t)$ is a feature function which is often binary-valued, but can be real-valued, and λ_k is a learned weight associated with feature f_k. The feature functions can measure any aspect of a state transition, $y_{t-1} \rightarrow y_t$, and the entire observation sequence, $\mathbf{x}$, centered at the current time step, t. For example, one feature function might have value 1 when y_{t-1} is the state START, y_t is the state NOT-START, and x_t is a word appearing in a lexicon of people's first names. Large positive values for λ_k indicate a preference for such an event; large negative values make the event unlikely.

The most probable label sequence for an input $\mathbf{x}$,

$$\mathbf{y}^* = \arg\max_y P_\Lambda(\mathbf{y}|\mathbf{x}),$$

can be efficiently determined using the Viterbi algorithm (Rabiner, 1990). An N-best list of labeling sequences can also be obtained using modified Viterbi algorithm and A* search (Schwartz and Chow, 1990).

The parameters can be estimated by maximum likelihood—maximizing the conditional probability of a set of label sequences, each given their corresponding input sequences. The log-likelihood of training set $\{(x_i, y_i) : i = 1, ...M\}$ is written

$$\begin{aligned} L_\Lambda &= \sum_i \log P_\Lambda(\mathbf{y}_i|\mathbf{x}_i) \\ &= \sum_i \left(\sum_{t=1}^{T}\sum_{k}\lambda_k f_k(y_{t-1}, y_t, \mathbf{x}, t) - \log Z_{x_i}\right). \end{aligned}$$

Traditional maximum entropy learning algorithms, such as GIS and IIS (della Pietra et al., 1995), can be used to train CRFs. However, our implementation uses a quasi-Newton gradient-climber BFGS for optimization, which has been shown to converge much faster (Malouf, 2002; Sha and Pereira, 2003). The gradient of the likelihood is $\partial P_\Lambda(\mathbf{y}|\mathbf{x})/\partial\lambda_k =$

$$\sum_{i,t} f_k(y_{t-1}, y_t^{(i)}, \mathbf{x}^{(i)}, t) - \sum_{i,\mathbf{y},t} P_\Lambda(\mathbf{y}|\mathbf{x}^{(i)}) f_k(y_{t-1}, y_t, \mathbf{x}^{(i)}, t)$$

CRFs share many of the advantageous properties of standard maximum entropy classifiers, including their convex likelihood function, which guarantees that the learning procedure converges to the global maximum.

2.1 Regularization in CRFs

To avoid over-fitting, log-likelihood is usually penalized by some prior distribution over the parameters. A commonly used prior is a zero-mean Gaussian. With a Gaussian prior, log-likelihood is penalized as follows.

$$L_\Lambda = \sum_i \log P_\Lambda(\mathbf{y}_i|\mathbf{x}_i) - \sum_k \frac{\lambda_k^2}{2\sigma_k^2} \qquad (2)$$

where σ_k^2 is the variance for feature dimension k.

The variance can be feature dependent. However for simplicity, constant variance is often used for all features. We experiment an alternate version of Gaussian prior in which the variance is feature dependent. We bin features by frequency in the training set, and let the features in the same bin share the same variance. The discounted value is set to be $\frac{\lambda_k}{\lceil c_k/M \rceil \times \sigma^2}$ where c_k is the count of features, M is the bin size set by held out validation, and $\lceil a \rceil$ is the ceiling function. See Peng and McCallum (2004) for more details and further experiments.

2.2 State transition features

Varying state-transition structures with different Markov order can be specified by different CRF feature functions, as determined by the number of output labels y examined together in a feature function. We define four different state transition feature functions corresponding to different Markov orders. Higher-order features capture more long-range dependencies, but also cause more data sparseness problems and require more memory for training. The best Markov order for a particular application can be selected by held-out cross-validation.

1. First-order: Here the inputs are examined in the context of the current state only. The feature functions are represented as $f(y_t, \mathbf{x})$. There are no separate parameters for state transitions.

2. First-order+transitions: Here we add parameters corresponding to state transitions. The feature functions used are $f(y_t, \mathbf{x})$, $f(y_{t-1}, y_t)$.

3. Second-order: Here inputs are examined in the context of the current and previous states. Feature function are represented as $f(y_{t-1}, y_t, \mathbf{x})$.

4. Third-order: Here inputs are examined in the context of the current, and two previous states. Feature function are represented as $f(y_{t-2}, y_{t-1}, y_t, \mathbf{x})$.

3 CRFs for Word Segmentation

We cast the segmentation problem as one of sequence tagging: Chinese characters that begin a new word are given the START tag, and characters in the middle and at the end of words are given the NONSTART tag. The task of segmenting new, unsegmented test data becomes a matter of assigning a sequence of tags (labels) to the input sequence of Chinese characters.

Conditional random fields are configured as a linear-chain (finite state machine) for this purpose, and tagging is performed using the Viterbi algorithm to efficiently find the most likely label sequence for a given character sequence.

3.1 Lexicon features as domain knowledge

One advantage of CRFs (as well as traditional maximum entropy models) is its flexibility in using arbitrary features of the input. To explore this advantage, as well as the importance of domain knowledge, we use many open features from external resources. To specifically evaluate the importance of domain knowledge beyond the training data, we divide our features into two categories: closed features and open features, (i.e., features allowed in the competition's "closed test" and "open test" respectively). The open features include a large word list (containing single and multiple-character words), a character list, and additional topic or part-of-speech character lexicons obtained from various sources. The closed features are obtained from training data alone, by intersecting the character list obtained from training data with corresponding open lexicons.

Many lexicons of Chinese words and characters are available from the Internet and other sources. Besides the word list and character list, our lexicons include 24 lists of Chinese words and characters obtained from several Internet sites[1] cleaned and augmented by a local native Chinese speaker independently of the competition data. The list of lexicons used in our experiments is shown in Figure 1.

3.2 Feature conjunctions

Since CRFs are log-linear models, feature conjunctions are required to form complex, non-linear decision boundaries in the original feature space. We

[1] http://www.mandarintools.com, ftp://xcin.linux.org.tw/pub/xcin/libtabe, http://www.geocities.com/hao510/wordlist

noun (e.g., 案, 癌)	verb (e.g., 拿)
adjective (e.g., 脏, 暗)	adverb (e.g., 最, 颇)
auxiliary (e.g., 能, 就)	preposition (e.g., 在)
number (e.g., 一, 二)	negative (e.g., 不, 非)
determiner (e.g., 个, 每, 这)	function (e.g. 兮, 并)
letter (English character)	punctuation (e.g., # $)
last name (e.g., 赵)	foreign name (e.g., 阿)
maybe last-name (e.g., 白, 何)	plural character (e.g., 俩, 们)
pronoun (e.g., 你, 我, 他)	unit character (e.g., 件, 个)
country name (e.g., 中, 美)	Chinese place name (e.g., 京)
organization name	title suffix (e.g., 长, 员)
title prefix (e.g., 总, 副)	date (e.g., 年, 月, 日)

Figure 1: Lexicons used in our experiments

C_{-2}:	second previous character in lexicon
C_{-1}:	previous character in lexicon
C_1:	next character in lexicon
C_2:	second next character in lexicon
C_0C_1:	current and next character in lexicon
$C_{-1}C_0$:	current and previous character in lexicon
$C_{-2}C_{-1}$:	previous two characters in lexicon
$C_{-1}C_0C_1$:	previous, current, and next character in the lexicon

Figure 2: Feature conjunctions used in experiments

use feature conjunctions in both the open and closed tests, as listed Figure 2.

4 Probabilistic New Word Identification

Since no vocabulary list could ever be complete, new word (unknown word) identification is an important issue in Chinese segmentation. Unknown words cause segmentation errors in that these out-of-vocabulary words in input text are often incorrectly segmented into single-character or other overly-short words (Chen and Bai, 1998). Traditionally, new word detection has been considered as a standalone process. We consider here new word detection as an integral part of segmentation, aiming to improve both segmentation and new word detection: detected new words are added to the word list lexicon in order to improve segmentation; improved segmentation can potentially further improve new word detection. We measure the performance of new word detection by its improvements on segmentation.

Given a word segmentation proposed by the CRF, we can compute a confidence in each segment. We detect as new words those that are not in the existing word list, yet are either highly confident segments, or low confident segments that are surrounded by high confident words. A confidence threshold of 0.9 is determined by cross-validation.

Segment confidence is estimated using constrained forward-backward (Culotta and McCallum, 2004). The standard forward-backward algorithm (Rabiner, 1990) calculates Z_x, the total likelihood of all label sequences $\mathbf{y}$ given a sequence $\mathbf{x}$. Constrained forward-backward algorithm calculates Z'_x, total likelihood of all paths passing through a constrained segment (in our case, a sequence of characters starting with a START tag followed by a few NONSTART tags before the next START tag). The confidence in this segment is then $\frac{Z'_x}{Z_x}$, a real number between 0 and 1.

In order to increase recall of new words, we consider not only the most likely (Viterbi) segmentation, but the segmentations in the top N most likely segmentations (an N-best list), and detect new words according to the above criteria in all N segmentations.

Many errors can be corrected by new word detection. For example, person name "吴思华" happens four times. In the first pass of segmentation, two of them are segmented correctly and the other two are mistakenly segmented as "吴 思 华" (they are segmented differently because Viterbi algorithm decodes based on context.). However, "吴思华" is identified as a new word and added to the word list lexicon. In the second pass of segmentation, the other two mistakes are corrected.

5 Experiments and Analysis

To make a comprehensive evaluation, we use all four of the datasets from a recent Chinese word segmentation bake-off competition (Sproat and Emerson, 2003). These datasets represent four different segmentation standards. A summary of the datasets is shown in Table 1. The standard bake-off scoring program is used to calculate precision, recall, F1, and OOV word recall.

5.1 Experimental design

Since CTB and PK are provided in the GB encoding while AS and HK use the Big5 encoding, we convert AS and HK datasets to GB in order to make cross-training-and-testing possible. Note that this conversion could potentially worsen performance slightly due to a few conversion errors.

We use cross-validation to choose Markov-order and perform feature selection. Thus, each training set is randomly split—80% used for training and the remaining 20% for validation—and based on validation set performance, choices are made for model structure, prior, and which word lexicons to include. The choices of prior and model structure shown in Table 2 are used for our final testing.

We conduct closed and open tests on all four datasets. The closed tests use only material from the training data for the particular corpus being tested. Open tests allows using other material, such as lexicons from Internet. In open tests, we use lexicons obtained from various resources as described

Corpus	Abbrev.	Encoding	#Train words	#Test Words	OOV rate (%)
UPenn Chinese Treebank	CTB	GB	250K	40K	18.1
Beijing University	PK	GB	1.1M	17K	6.9
Hong Kong City U	HK	Big 5	240K	35K	7.1
Academia Sinica	AS	Big 5	5.8M	12K	2.2

Table 1: Datasets statistics

	bin-Size M	Markov order
CTB	10	first-order + transitions
PK	15	first-order + transitions
HK	1	first-order
AS	15	first-order + transitions

Table 2: Optimal prior and Markov order setting

in Section 3.1. In addition, we conduct cross-dataset tests, in which we train on one dataset and test on other datasets.

5.2 Overall results

Final results of CRF based segmentation with new word detection are summarized in Table 3. The upper part of the table contains the closed test results, and the lower part contains the open test results. Each entry is the performance of the given metric (precision, recall, F1, and R_{oov}) on the test set.

Closed				
	Precision	Recall	F1	R_{oov}
CTB	0.828	0.870	0.849	0.550
PK	0.935	0.947	0.941	0.660
HK	0.917	0.940	0.928	0.531
AS	0.950	0.962	0.956	0.292
Open				
	Precision	Recall	F1	R_{oov}
CTB	0.889	0.898	0.894	0.619
PK	0.941	0.952	0.946	0.676
HK	0.944	0.948	0.946	0.629
AS	0.953	0.961	0.957	0.403

Table 3: Overall results of CRF segmentation on closed and open tests

To compare our results against other systems, we summarize the competition results reported in (Sproat and Emerson, 2003) in Table 4. XXc and XXo indicate the closed and open runs on dataset XX respectively. Entries contain the F1 performance of each participating site on different runs, with the best performance in bold. Our results are in the last row. Column SITE-AVG is the average F1 performance over the datasets on which a site reported results. Column OUR-AVG is the average F1 performance of our system over the same datasets.

Comparing performance across systems is difficult since none of those systems reported results on all eight datasets (open and closed runs on 4 datasets). Nevertheless, several observations could be made from Table 4. First, no single system achieved best results in all tests. Only one site (S01) achieved two best runs (CTBc and PKc) with an average of 91.8% over 6 runs. S01 is one of the best segmentation systems in mainland China (Zhang et al., 2003). We also achieve two best runs (ASo and HKc), with a comparable average of 91.9% over the same 6 runs, and a 92.7% average over all the 8 runs. Second, performance varies significantly across different datasets, indicating that the four datasets have different characteristics and use very different segmentation guidelines. We also notice that the worst results were obtained on CTB dataset for all systems. This is due to significant inconsistent segmentation in training and testing (Sproat and Emerson, 2003). We verify this by another test. We randomly split the training data into 80% training and 20% testing, and run the experiments for 3 times, resulting in a testing F1 of 97.13%. Third, consider a comparison of our results with site S12, who use a sliding-window maximum entropy model (Xue, 2003). They participated in two datasets, with an average of 93.8%. Our average over the same two runs is 94.2%. This gives some empirical evidence of the advantages of linear-chain CRFs over sliding-window maximum entropy models, however, this comparison still requires further investigation since there are many factors that could affect the performance such as different features used in both systems.

To further study the robustness of our approach to segmentation, we perform cross-testing—that is, training on one dataset and testing on other datasets. Table 5 summarizes these results, in which the rows are the training datasets and the columns are the testing datasets. Not surprisingly, cross testing results are worse than the results using the same

	ASc	ASo	CTBc	CTBo	HKc	HKo	PKc	PKo	SITE-AVG	OUR-AVG
S01	93.8		**88.1**	88.1	90.1		**95.1**	95.3	91.8	**91.9**
S02			87.4	**91.2**					**89.3**	87.2
S03		87.2		82.9		88.6		92.5	87.8	**93.6**
S04							93.9	93.7	93.8	**94.4**
S05	94.2		73.2				89.4		85.6	**91.5**
S06	94.5		82.9		92.4		92.4		90.6	**91.9**
S07								94.0	94.0	**94.6**
S08					90.4	**95.6**	93.6	93.8	93.4	**94.0**
S09	**96.1**						94.6		**95.4**	94.9
S10			83.1	90.1			94.7	**95.9**	**91.0**	90.8
S11		90.4		88.4		87.9		88.6	88.8	**93.6**
S12	95.9				91.6				93.8	**94.2**
	95.6	**95.7**	84.9	89.4	**92.8**	94.6	94.1	94.6		92.7

Table 4: Comparisons against other systems: the first column contains the 12 sites participating in bake-off competition; the second to the ninth columns contain their results on the 8 runs, where a bold entry is the winner of that run; column SITE-AVG contains the average performance of the site over the runs in which it participated, where a bold entry indicates that this site performs better than our system; column OUR-AVG is the average of our system over the same runs, where a bolded entry indicates our system performs better than the other site; the last row is the performance of our system over all the runs and the overall average.

source as training due to different segmentation policies, with an exception on CTB where models trained on other datasets perform better than the model trained on CTB itself. This is due to the data problem mentioned above. Overall, CRFs perform robustly well across all datasets.

From both Table 3 and 5, we see, as expected, improvement from closed tests to open tests, indicating the significant contribution of domain knowledge lexicons.

Closed				
	CTB	PK	HK	AS
CTB		0.822	0.810	0.815
PK	0.816		0.824	0.830
HK	0.790	0.807		0.825
AS	0.890	0.844	0.864	
Open				
	CTB	PK	HK	AS
CTB		0.863	0.870	0.894
PK	0.852		0.862	0.871
HK	0.861	0.871		0.889
AS	0.898	0.867	0.871	

Table 5: Crossing test of CRF segmentation

5.3 Effects of new word detection

Table 6 shows the effect of new word detection on the closed tests. An interesting observation is that the improvement is monotonically related to the OOV rate (OOV rates are listed in Table 1). This is desirable because new word detection is most needed in situations that have high OOV rate. At low OOV rate, noisy new word detection can result in worse performance, as seen in the AS dataset.

	CTB	PK	HK	AS
w/o NWD	0.792	0.934	0.916	0.956
NWD	0.849	0.941	0.928	0.946

Table 6: New word detection effects: *w/o NWD* is the results without new word detection and *NWD* is the results with new word detection.

5.4 Error analysis and discussion

Several typical errors are observed in error analysis. One typical error is caused by inconsistent segmentation labeling in the test set. This is most notorious in CTB dataset. The second most typical error is in new, out-of-vocabulary words, especially proper names. Although our new word detection fixes many of these problems, it is not effective enough to recognize proper names well. One solution to this problem could use a named entity extractor to recognize proper names; this was found to be very helpful in Wu (2003).

One of the most attractive advantages of CRFs (and maximum entropy models in general) is its the flexibility to easily incorporate arbitrary features,

here in the form domain-knowledge-providing lexicons. However, obtaining these lexicons is not a trivial matter. The quality of lexicons can affect the performance of CRFs significantly. In addition, compared to simple models like n-gram language models (Teahan et al., 2000), another shortcoming of CRF-based segmenters is that it requires significantly longer training time. However, training is a one-time process, and testing time is still linear in the length of the input.

6 Conclusions

The contribution of this paper is three-fold. First, we apply CRFs to Chinese word segmentation and find that they achieve state-of-the art performance. Second, we propose a probabilistic new word detection method that is integrated in segmentation, and show it to improve segmentation performance. Third, as far as we are aware, this is the first work to comprehensively evaluate on the four benchmark datasets, making a solid baseline for future research on Chinese word segmentation.

Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval, in part by The Central Intelligence Agency, the National Security Agency and National Science Foundation under NSF grant #IIS-0326249, and in part by SPAWARSYSCEN-SD grant number N66001-02-1-8903.

References

K.J. Chen and M.H. Bai. 1998. Unknown Word Detection for Chinese by a Corpus-based Learning Method. *Computational Linguistics and Chinese Language Processing*, 3(1):27–44, Feburary.

A. Culotta and A. McCallum. 2004. Confidence Estimation for Information Extraction. In *Proceedings of Human Language Technology Conference and North American Chapter of the Association for Computational Linguistics(HLT-NAACL)*.

S. della Pietra, V. della Pietra, and J. Lafferty. 1995. Inducing Features Of Random Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4).

J. Gao, M. Li, and C. Huang. 2003. Improved Source-Channel Models for Chinese Word Segmentation. In *Proceedings of the 41th Annual Meeting of Association of Computaional Linguistics (ACL)*, Japan.

J. Lafferty, A. McCallum, and F. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the 18th International Conf. on Machine Learning*, pages 282–289.

R. Malouf. 2002. A Comparison of Algorithms for Maximum Entropy Parameter Estimation. In *Sixth Workshop on Computational Language Learning (CoNLL)*.

J. Nie, M. Hannan, and W. Jin. 1995. Unknown Word Detection and Segmentation of Chinese using Statistical and Heuristic Knowledge. *Communications of the Chinese and Oriental Languages Information Processing Society*, 5:47–57.

F. Peng and A. McCallum. 2004. Accurate Information Extraction from Research Papers using Conditional Random Fields. In *Proceedings of Human Language Technology Conference and North American Chapter of the Association for Computational Linguistics(HLT-NAACL)*, pages 329–336.

F. Peng and D. Schuurmans. 2001. Self-Supervised Chinese Word Segmentation. In F. Hoffmann et al., editor, *Proceedings of the 4th International Symposium of Intelligent Data Analysis*, pages 238–247. Springer-Verlag Berlin Heidelberg.

L. Rabiner. 1990. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In Alex Weibel and Kay-Fu Lee, editors, *Readings in Speech Recognition*, pages 267–296.

R. Schwartz and Y. Chow. 1990. The N-best Algorithm: An Efficient and Exact Procedure for Finding the N most Likely Sentence Hypotheses. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

F. Sha and F. Pereira. 2003. Shallow Parsing with Conditional Random Fields. In *Proceedings of Human Language Technology Conference and North American Chapter of the Association for Computational Linguistics(HLT-NAACL)*.

R. Sproat and T. Emerson. 2003. First International Chinese Word Segmentation Bakeoff. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*.

R. Sproat and C. Shih. 2002. Corpus-based Methods in Chinese Morphology and Phonology. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING)*.

W. J. Teahan, Y. Wen, R. McNab, and I. H. Witten. 2000. A Compression-based Algorithm for Chinese Word Segmentation. *Computational Linguistics*, 26(3):375–393.

A. Wu and Z. Jiang. 2000. Statistically-Enhanced New Word Identification in a Rule-Based Chinese System. In *Proceedings of the Second Chinese Language Processing Workshop*, pages 46–51, Hong Kong, China.

Z. Wu. 1999. LDC Chinese Segmenter. http://www.ldc.upenn.edu/ Projects/ Chinese/ segmenter/ mansegment.perl.

A. Wu. 2003. Chinese Word Segmentation in MSR-NLP. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, Japan.

N. Xue. 2003. Chinese Word Segmentation as Character Tagging. *International Journal of Computational Linguistics and Chinese Language Processing*, 8(1).

H. Zhang, Q. Liu, X. Cheng, H. Zhang, and H. Yu. 2003. Chinese Lexical Analysis Using Hierarchical Hidden Markov Model. In *Proceedings of the Second SIGHAN Workshop*, pages 63–70, Japan.