Human-Computer Interactive Chinese Word Segmentation: An Adaptive Dirichlet Process Mixture Model Approach

Tongfei Chen[1], **Xiaojun Zou**[2], **Weimeng Zhu**[1], **Junfeng Hu**[2, *]
[1] School of Electronics Engineering and Computer Science,
Peking University, Beijing, 100871, P. R. China
[2] Key Laboratory of Computational Linguistics (Peking University),
Ministry of Education, Beijing, 100871, P. R. China
{ctf,zouxj,zwm,hujf}@pku.edu.cn

Abstract

Previous research shows that Kalman filter based human-computer interactive Chinese word segmentation achieves an encouraging effect in reducing user interventions, but suffers from the drawback of incompetence in distinguishing segmentation ambiguities. This paper proposes a novel approach to handle this problem by using an adaptive Dirichlet process mixture model. By adjusting the hyperparameters of the model, ideal classifiers can be generated to conform to the interventions provided by the users. Experiments reveal that our approach achieves a notable improvement in handling segmentation ambiguities. With knowledge learnt from users, our model outperforms the baseline Kalman filter model by about 0.5% in segmenting homogeneous texts.

1 Introduction

As Chinese text is written without natural delimiters such as whitespaces, word segmentation is often the essential first step in Chinese language processing (Liang, 1987). Over the past two decades, various methods have been developed to address this issue (Nie et al., 1994; Sun et al., 1998; Luo et al., 2002; Zhang et al., 2003; Peng et al., 2004; Goldwater et al., 2006). Generally, supervised statistical learning methods are more adaptive and robust in processing unrestricted texts than the traditional dictionary-based methods.

However, in some domain-specific applications, for example ancient Chinese text processing, there is neither enough homogeneous corpora for training a reliable statistical model, nor a well-defined dictionary. In these tasks, unsupervised word segmentation is preferred to utilize the linguistic knowledge derived from the raw corpus itself. Many researches also enable users to take part in the segmentation process, adding expert knowledge to the system (Wang et al., 2002; Li and Chen, 2007). This is quite reasonable since the criteria of word segmentation are dependent on a user or the destination of use in many applications (Sproat et al., 1996).

Zhu et al. (2013) proposed a Kalman filter based human-computer interactive learning model for segmenting Chinese texts depending upon neither lexicon nor any annotated corpus. This approach enables experts to observe and intervene with the segmentation results, while the segmenter learns and adapts to these knowledge iteratively. At the end of this procedure, a segmentation result that fully matches the demand of the user is returned. However, in some complicated cases where segmentation ambiguities exist, the Kalman filter will not converge and keep swapping in two or more states.

To overcome this drawback, we established an adaptive Dirichlet process mixture model (ADPMM) for human-computer interactive word segmentation. ADPMM gradually adapts itself to the knowledge supplied by users through the process of human-computer interaction, notably reducing human interventions by classifying each occurrence of a bigram into its corresponding class. Each generated class bears a tag *separated* or *combined* derived from user interventions; bigrams classified to a class later is judged as *separated* or *combined* according to the class tag. Knowledge learnt from the user can further

[*] To whom all correspondence should be addressed.

be used to aid the segmentation of homogeneous corpus.

The rest of this paper is organized as follows. The next section reviews related work. The details of our model are elaborated in Section 3. In Section 4, experiments are presented to illustrate the performance of our model. The final section concludes the proposed model and discusses possible future work.

2 Related Work

Unsupervised word segmentation is generally based on some predefined criteria, for example *mutual information* (*mi*), to recognize a substring as a word. Sproat and Shih (1990) studied comprehensively in this direction using mutual information. Many successive researches applied different ensemble methods to mutual information (Chien, 1997; Yamamoto and Kenneth, 2001). Sun et al. (2004) designed an algorithm based on the linear combination of *mi* and *difference of t-score* (*dts*). Other criteria like *description length gain* (Kit and Wilks, 1999), *assessor variety* (Feng et al., 2004) and *branch entropy* (Jin and Tanaka-Ishii, 2006) were also explored.

Any automatic segmentation has limitations in some way and is far from fully matching the particular need of users. Thus, human-computer interactive strategies are explored to allow users to pass their linguistic knowledge to the segmenter by directly intervening the segmentation process. Wang et al. (2002) developed a sentence-based human-computer interaction inductive learning method. Feng et al. (2006) proposed a certainty-based active learning segmentation algorithm to train an *n*-gram language model in an unsupervised learning framework. Li and Chen (2007) further explored a candidate word based human-computer interactive segmentation strategy.

Kalman filters (Kalman, 1960) are based on linear dynamic systems discretized in the time domain. Given parameters, Kalman filters estimates the unobserved state. Zhu et al. (2013) applied Kalman filter model to learn and estimate user intentions in their human-computer interactive word segmentation framework.

A Dirichlet process is a stochastic process that is a distribution whose domain is itself a distribution (Ferguson, 1973). It can also be viewed as an infinite-dimensional generalization of the Dirichlet distribution. It can be used to construct a mixture model with an unknown number of components (West et al., 1993). Dirichlet processes have been used to handle Chinese word segmentation. Goldwater et al. (2006) explored a bigram model built upon a Dirichlet process to discover contextual dependencies.

3 Model

3.1 Baseline Model

Sun et al. (1998) proposed *difference of t-score* (*dts*) as a useful complement to *mutual information* (*mi*). They further designed a compound statistical measurement based on the linear combination of *mi* and *dts*, named *md* (Sun et al., 2004). Given any bigram *xy*, in terms of *md*(*x*,*y*) and a threshold Θ, whether the bigram should be combined or separated can be determined—when *md*(*x*,*y*) is greater than Θ, the bigram *xy* has more chance to be in a word. This model is a reference to our basic model before the human-computer interaction process. The formulae for calculating *md* are as follows:

$$mi*(x,y) = \frac{mi(x,y)-\mu_{mi}}{\sigma_{mi}},$$
$$dts*(x,y) = \frac{dts(x,y)-\mu_{dts}}{\sigma_{dts}}, \qquad (1)$$
$$md(x,y) = mi*(x,y) + \lambda \times dts*(x,y),$$

where μ_{mi} and μ_{dts} are means of *mi* and *dts* in the corpus; σ_{mi} and σ_{dts} are standard deviations. *mi** and *dts** are normalized versions of measure *mi* and *dts*; λ is an empirical value.

Meanwhile, there is an optimization where local maxima and minima of *md* appear (Sun et al., 2004). Consider a character string *abcd*. If *md*(*b*,*c*) > *md*(*a*,*b*) and *md*(*b*,*c*) > *md*(*c*,*d*), then *bc* is considered a *local maximum*. *Local minimum* follows a similar definition. Obviously, local maxima are more likely to form words, while local minima are more likely to be separated. To reflect this kind of tendency, we increase the *md* values at local maxima by a constant *s*, and decrease the *md* values at local minima by *s*.

Based on the compound statistical measure *md*, Zhu et al. (2013) further developed a human-computer interactive word segmentation framework. In their model, the human interaction process is mapped to a time series process, and user judgments are treated as measurements of the true *md* value of bigrams. Each bigram is modeled by a Kalman filter independently to learn and estimate user intentions from user interventions (which may contain noise). Linguistic knowledge is gradually accumulated from the interactions, and eventually, a segmentation that fully matches the specific use is returned.

Both baseline models above use a threshold value Θ to classify each bigram into two classes, namely *combined* and *separated*. Both approaches are inherently binary classifiers which seek to classify occurrences of bigrams into classes.

3.2 Problem of Segmentation Ambiguity

In the scenario of human-computer interactive Chinese word segmentation, the Kalman filter approach proposed by Zhu et al. (2013) encounters the problem of segmentation ambiguity, rendering it unsuccessful in converging in some special cases. If segmentation ambiguity exists, human interventions would be swapping, which in turn results in swapping states of the Kalman filter.

Take the bigram 及其 used in Zhu et al. (2013) as an example. It exhibits at least two types of segmentation in the corpus (e.g., *separated* in 以及/其他 'and others', and *combined* in 及其/浮动 'and its fluctuations'). The Kalman filter approach will not converge, and will keep swapping between two or more states as shown in Figure 1.

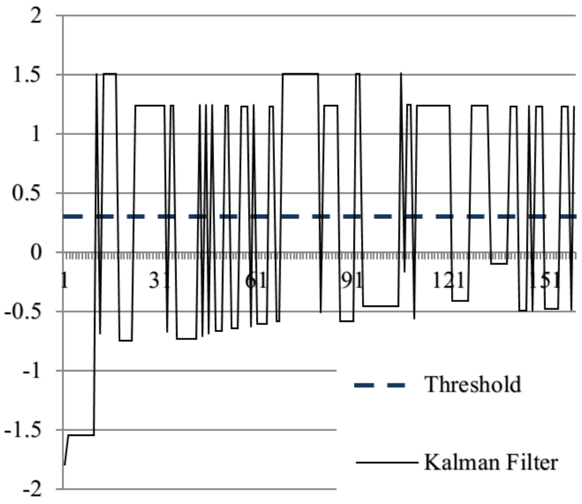

Figure 1. Problem encountered in Kalman Filter model on the bigram 及其. The vertical axis denotes the *md* value, and the horizontal axis denotes the occurrence of 及其 in the text. An increase in the value denotes that there exist interventions tagged by the user as combined; whereas a decrease in the value indicates the presence of interventions tagged as separated.

To address this problem, we adopt the *md* measure described by Sun et al. (2004) to construct a Dirichlet process mixture model to classify each occurrence of a bigram into its corresponding class. Each class bears a tag *separated* or *combined* derived from user interventions. This model gradually adapts itself to the knowledge supplied by the user's interventions through human-computer interaction, making it more robust in distinguishing segmentation ambiguities through the process of classifying them into different classes.

3.3 Adaptive Dirichlet Process Mixture Model

To address the problem mentioned above, classification of each occurrence of a bigram into its corresponding class is required. Since we cannot predict the exact number of classes, a Dirichlet process mixture model (West et al., 1993) would suffice. Similar to the Kalman filter based approach, we also assume that each bigram is independent, i.e., if the model for one bigram changes, other bigrams is not affected. To simplify our discussion, we focus on only one bigram in this section. Notations used in this paper are listed in Table 1.

Symbol	Definition
Θ	Threshold *md* value
x_i	The *md* value of the *i*th occurrence of the specific bigram
μ_k	The expectation of the *k*th class
σ_k^2	The variance of the *k*th class
z_i	The class indicator of sample x_i, i.e., x_i belongs to class z_i
α	Concentration parameter of the Dirichlet process mixture model
H	Prior base distribution of the Dirichlet process mixture model
$N(x \mid \mu, \sigma^2)$	Probability density of $N(\mu, \sigma^2)$ at x
$H(\mu, \sigma^2)$	Probability density of H at (μ, σ^2)
$\text{N-}\Gamma^{-1}$	Normal-inverse-gamma distribution
ψ	Prior sum of squared deviations of the mixture model

Table 1. Notations used in this paper.

We consider the *md* value of each occurrence of a bigram as a *sample* of the bigram. Initially, samples are classified into class *separated* or *combined* according to threshold value Θ. During the interaction process, more classes should be generated to handle complex situations when binary classifiers are unable to produce correct segmentation result. As the exact number of classes cannot be predicted, the model used for the generation of multiple classes can be formulated as an infinite Gaussian mixture model, in which each sample belongs to a class that follows a Gaussian distribution, and each distribution is specified by a mean and a variance.

Infinite Gaussian mixture models can be formulated by a Dirichlet process with concentration parameter α and base distribution H (West et al., 1993):

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H)\,, \\ (\mu_k, \sigma_k^2) &\sim G\,, \qquad (2) \\ x_{i,\cdots,N} &\sim N(\mu_k, \sigma_k^2)\,. \end{aligned}$$

For simplicity, we choose the prior base distribution H to be the conjugate prior of $N(\mu, \sigma^2)$. The conjugate prior of a Gaussian distribution with unknown expectation and variance is the normal-inverse-gamma distribution:

$$(\mu, \sigma^2) \sim H = \mathrm{N}\text{-}\Gamma^{-1}(\mu_0, \kappa, v, \psi)\,, \qquad (3)$$

where μ_0 is the prior expectation of μ estimated from κ observations, and ψ is the prior sum of squared deviations estimated from v observations (O'Hagan et al., 2004).

The prior parameter α and ψ are of special interest here. Parameter α is the concentration of the Dirichlet process. The greater α is, the probability of producing more classes increases. Parameter ψ represents the prior sum of squared deviations of each class. The lesser ψ is, the higher precision a class is, and the range the class covers becomes smaller.

To produce more classes that covers smaller ranges, we increase α and decrease ψ. We define this step as ADJUSTPARAMETER, which is implemented by multiplying a constant value to α and ψ respectively.

In the scenario of human-computer interactive word segmentation, humans can judge whether the segmentation result produced by the segmenter is correct or not. These judgments act as constraints over samples. Classes produced shall conform to these judgments, i.e., samples within each class are uniformly judged as *separated* or uniformly judged as *combined*. If the initial result does not conform to human judgments, more classes with smaller ranges should be generated. Thus ADJUSTPARAMETER should be performed.

Our adaptive Dirichlet process mixture model works as follows: In the initial state of a bigram, we construct a classifier such that all samples below the threshold Θ are marked *separated*, while all samples above Θ are marked *combined*. Whenever a user intervention occurred, implying that the current classifier cannot distinguish certain segmentation ambiguities, we increase the concentration parameter α, i.e. increase the probability to generate more classes, and decrease the prior class sum of squared deviations ψ, i.e. increase the precision of a class. With these parameters adjusted, re-cluster all the samples to date. Since the prior parameters α and ψ are adjusted, the segmenter tends to produce more classes. Iterate this process until all classes conform to human judgments, i.e., samples within each produced class share the same human judgment. Then tag each class with *separated* or *combined* according to the human judgments of the samples in that class. In this way, the Dirichlet process mixture model will adapt itself to conform to human judgments upon samples. This algorithm is illustrated in Algorithm 1.

In Algorithm 1, Line 5 and 6 implements ADJUSTPARAMETER. Empirical values 2.0 and 0.9 are assigned to coefficients p_α and p_ψ. The algorithm CLUSTERBYDPMM will be elaborated in Section 3.4.

Algorithm 1. ADAPTIVEDPMM

```
   Input: Sample set $X$, human judgments $J$
   Output: Clustering result $C$
1: begin
2:   do
3:     $C \leftarrow$ CLUSTERBYDPMM($X$, $\alpha$, $\psi$)
4:     if $C$ conforms to human judgments $J$ break
5:     $\alpha \leftarrow \alpha \times p_\alpha$
6:     $\psi \leftarrow \psi \times p_\psi$
7:   while maximum iteration count not reached
8:   Tag each class in $C$ according to judgments $J$
9: end
```

Whenever a user intervention occurred, the algorithm above is run once, and it returns the expectation and variance of each class, along with the class tag. Use the expectation and variance to construct a naïve Bayes classifier from these data, namely

$$z = \arg\max_k P(k) N(x \mid \mu_k, \sigma_k^2)\,, \qquad (4)$$

where x is a new sample, z is the class which x belongs to, and μ_k, σ_k^2 are the expectation and variance of class k. $P(k)$ is the class-prior, i.e. the proportion class k takes in the whole set of samples. Bigram with *md* value x is judged *separated* or *combined* according the tag associated with class z. This naïve Bayes classifier is used to classify new occurrences of the bigram until the user intervenes again.

3.4 Inference of the Dirichlet Process Mixture Model

Each time a user intervenes in the segmentation process, implying that the samples should be re-clustered, we use a Gibbs sampler to perform the clustering task (MacEachern, 1994; Neal, 2000; Rasmussen, 2000). The algorithm below adopts Algorithm 3 described by Neal (2000).

Set up a Markov chain whose state consists of $\mathbf{z} = (z_1, \cdots, z_n)$, i.e., the class indicator of current samples. Repeatedly sample as follows:

For *i* = 1, …, *n*: Draw a new value for z_i from:

$$P(z_i = z \mid z_{-i}, x_i) \propto \begin{cases} \dfrac{n_{-i,z}}{n-1+\alpha} \displaystyle\int N(x_i \mid \varphi) H_{-i,z}(\varphi)\,\mathrm{d}\varphi \\ \qquad \text{if } z = z_j \text{ for some } j \neq i \\ \dfrac{\alpha}{n-1+\alpha} \displaystyle\int N(x_i \mid \varphi) H(\varphi)\,\mathrm{d}\varphi \\ \qquad \text{if } z_i \neq z_j \text{ for all } j \neq i \end{cases}, \qquad (5)$$

where φ indicates the parameter pair (μ, σ^2) ; $n_{-i,z}$ is the number of samples in class *z* except x_i ; and $H_{-i,z}$ is the posterior distribution of φ based on the prior *H* and all observations x_j for which $j \neq i$ and $z_j = z$.

Since *H* is chosen to be the conjugate prior of Gaussian distribution, i.e. the normal-inverse-gamma distribution mentioned in Section 3.3, the integral term in Equation (5) is analytically feasible, thus the sampling method presented here is feasible.

4 Experiments

In this section, we conducted several experiments to evaluate the performance of our segmentation model. Firstly, we analyzed the performance of segmentation ambiguity handling through a case study. Secondly, we verified the improvement in reducing human intervention after introducing our model. Thirdly, we tested the reusability of knowledge learnt from human interaction. The experiments are based on the *People's Daily* corpus from Jan. 1998 to Jun. 1998 provided by the Institute of Computational Linguistics, Peking University.

Several baseline models are used in this section. One is the approach proposed by Sun et al. (2004) (abbreviated as *Sun's Appr.*) mentioned in Section 3.1, and the other is the Kalman Filter based approach proposed by Zhu et al. (2013) (abbreviated as *Zhu's Appr.*). In addition, the memory approach (abbreviated as *Memory Appr.*), a bigram based human interactive model whose initial segmentation is exactly the same as Sun's Approach but its prediction of the bigram is taken from the latest human intervention (i.e., the latest correct segmentation result judged by human), is also compared in Sections 4.2 and 4.3. Our adaptive Dirichlet process mixture model is abbreviated as ADPMM.

4.1 Case Study

In this part, we took the aforementioned bigram 及其 as an example, and examined the exact number of interventions by users during the human-computer interaction process. Models used for comparison are Memory Appr., Zhu's Appr. and ADPMM. The simulation of the segmentation process was performed by using the correct segmentation text as input to the model. We define an *intervention rate* (IR) of a specific bigram to measure the human effort in a corpus. The IR of bigram *xy* is defined as

$$\mathrm{IR}[\%] = \frac{\#\text{ of interventions of } xy}{\#\text{ of occurrences of } xy} \times 100\% . \qquad (6)$$

Table 2 shows the number of interventions (denoted by NI) and the IR of bigram 及其 under each model with *People's Daily* Jan. 1998 to Mar. 1998 as test text. It can be seen from the table that ADPMM significantly reduced the number of interventions of bigram 及其 under all three corpora. In Feb. 1998, ADPMM reduced the NI from 36 in Zhu's Appr. to 16 (about 55.56% reduction in percentage), while in Mar. 1998, from 36 to 10 (about 72.22% reduction in percentage). This experiment shows that our model greatly reduced the number of interventions in the case of the segmentation-ambiguous word 及其.

Corpus		Memory Appr.	Zhu's Appr.	**ADPMM**
Jan.	NI	63	43	**17**
	IR	39.38	26.88	**10.63**
Feb.	NI	63	36	**16**
	IR	42.00	24.00	**10.67**
Mar.	NI	56	36	**10**
	IR	25.00	16.07	**4.46**

Table 2. Number of interventions (NI) and IR[%] of bigram 及其 under different corpora.

4.2 Simulating the Human-Computer Interactive Segmentation Process

In this part, we simulated the human-computer interaction by using the correct segmentation text as input to the model. We adopted the binary prediction rate (BPR) described by Zhu et al. (2013) to quantify the conformity of the prediction of the model to user intention. BPR is defined as

$$\text{BPR}[\%] = \frac{\#\text{of correct predictions}}{\#\text{of all predictions}} \times 100\% . \quad (7)$$

The result of the experiment is shown in Table 3. It can be seen that our model gained a slightly higher BPR than both Zhu’s Appr. and Memory Appr. (this is because segmentation ambiguities are relatively rare in corpora), which indicates that our model can reduce user interventions more effectively than Zhu’s Appr.

Corpus	Sun’s Appr.	Memory Appr.	Zhu’s Appr.	**ADPMM**
Jan.	84.22	94.55	94.66	**94.95**
Feb.	84.58	94.74	94.83	**95.14**
Mar.	84.59	95.04	95.17	**95.46**

Table 3. BPR[%] of different approaches under different corpora.

4.3 Knowledge Reusability Test

After the experiment in Section 4.1, we obtained the classification information for each bigram, and we assumed that this information can be viewed as a kind of learnt knowledge that could be used to aid further word segmentation on homogeneous corpus.

In this part, we performed an incremental test on knowledge reusability. This is done by applying the model with knowledge learnt from text of previous months to segment the text of the current month, from Jan. to Jun., respectively. For example, we took the model with knowledge learnt from Jan. to segment the text of Feb.; the model with knowledge learnt from both Jan. and Feb. to segment text of Mar.; and so on. The BPRs of Memory Appr., Zhu’s Appr. and ADPMM are recorded using the testing scheme described above. As is shown in Figure 2, with the knowledge accumulating, the advantage of our model increases significantly: on Jan. (no previous knowledge exists), the advantage of our model is 0.29% and 0.40% over Zhu’s Appr. and Memory Appr. respectively; on Jun. this advantage is enlarged to 0.47% and 0.68%; on May., this advantage reached 0.56% and 0.80%. This experiments shows that when a large training corpus is present, knowledge of segmentation ambiguities will be stored in our model through the form of different classes of a bigram, making it more robust in handling future segmentation ambiguities.

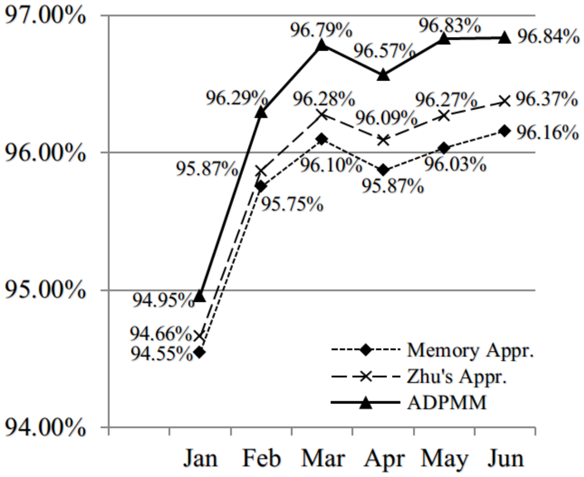

Figure 2. BPR[%] of different word segmentation approaches using an incremental testing scheme.

5 Conclusions and Future Work

Research shows that Kalman filter based human-computer interactive Chinese word segmentation framework suffers from the drawback of ineptitude in handling segmentation ambiguities. This paper proposes an adaptive Dirichlet process mixture model (ADPMM). ADPMM adjusts the hyperparameters so that ideal classifiers can be generated to conform to the interventions provided by the users. Experiments showed that our approach achieves a notable improvement (more than 55.56% in a case study) in handling segmentation ambiguities, therefore effective in reducing human effort. In the knowledge reusability test, our model outperforms the baseline Kalman filter model by about 0.5% in segmenting homogeneous texts with knowledge learnt from users.

Our future work will concentrate on improving statistics criteria that would reflect contexts more precisely. As in the experiments, we found that the number of classes may grow rapidly. This is caused by the ineffectiveness of the *md* measure to distinguish different contexts.

Acknowledgements

We would like to thank Professor Sujian Li for her valuable advice on writing this paper. This work is partially supported by Open Project Program of the National Laboratory of Pattern Recognition (NLPR) and the Opening Project of Beijing Key Laboratory of Internet Culture and Digital Dissemination Research (ICDD201102).

References

Lee-Feng Chien. 1997. Pat-Tree-Based Keyword Extraction for Chinese Information Retrieval. In *ACM SIGIR Forum*, pages 50-58.

Michael D. Escobar. 1994. Estimating Normal Means with a Dirichlet Process Prior. *Journal of the American Statistical Association*, 89(425): 268-277.

Chong Feng, Zhaoxiong Chen, Heyan Huang, and Zhenzhen Guan. 2006. Active Learning in Chinese Word Segmentation Based on Multigram Language Model. *Journal of Chinese Information Processing*, 20(1): 50-58 (in Chinese)

Haodi Feng, Kang Chen, Xiaotie Deng, and Weimin Zheng. 2004. Accessor Variety Criteria for Chinese Word Extraction. *Computational Linguistics*, 30(1): 75-93.

Thomas S. Ferguson. 1973. A Bayesian Analysis of Some Nonparametric Problems. *The Annals of Statistics*, pages 209-230.

Sharon Goldwater, Thomas L. Griffiths, and Mark Johnson. 2006. Contextual Dependencies in Unsupervised Word Segmentation. In *COLING/ACL* 2006, pages 673-680.

Zhihui Jin and Kumiko Tanaka-Ishii. 2006. Unsupervised Segmentation of Chinese Text by Use of Branching Entropy. In *COLING/ACL* 2006, pages 428-435.

Chunyu Kit and Yorick Wilks. 1999. Unsupervised Learning of Word Boundary with Description Length Gain. In *Proceedings of the CoNLL99 ACL Workshop*, pages 1-6.

Bin Li and Xiaohe Chen. 2007. A Human-Computer Interaction Word Segmentation Method Adapting to Chinese Unknown Texts. *Journal of Chinese Information Processing*, 21(3): 92-98. (in Chinese)

Rudolph E. Kalman. 1960. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1): 35-45.

Nanyuan Liang. 1987. CDWS: An Automatic Word Segmentation System for Written Chinese Texts. *Journal of Chinese Information Processing,* 1(2): 44-52. (in Chinese)

Xiao Luo, Maosong Sun, and Benjamin K. Tsou. 2002. Covering Ambiguity Resolution in Chinese Word Segmentation Based on Contextual Information. In *COLING* 2002, pages 1-7.

Radford M. Neal. 2000. Markov Chain Sampling Methods for Dirichlet Process Mixture Models. *Journal of Computational and Graphical Statistics*, 9(2): 249-265.

Jian-Yun Nie, Wanying Jin, and Marie-Louise Hannan. 1994. A Hybrid Approach to Unknown Word Detection and Segmentation of Chinese. In *Proceedings of International Conference on Chinese Computing*, pages 326-335.

Anthony O'Hagan, Jonathan Forster, and Maurice G. Kendall. 2004. *Bayesian Inference*. London: Arnold.

Fuchun Peng, Fangfang Feng, and Andrew McCallum. 2004. Chinese Segmentation and New Word Detection Using Conditional Random Fields. In *COLING* 2004, pages 23-27.

Carl E. Rasmussen. 2000. The Infinite Gaussian Mixture Model. *Advances in Neural Information Processing Systems*, 12(5.2): 2.

Richard Sproat, William Gale, Chilin Shih, and Nancy Change. 1996. A Stochastic Finite-State Word Segmentation Algorithm for Chinese. *Computational Linguistics*, 22(3): 377-404.

Richard Sproat and Chilin Shih. 1990. A Statistical Method for Finding Word Boundaries in Chinese Text. *Computer Processing of Chinese and Oriental Languages*, pages 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* 1998, (1998) 1265-1271.

Maosong Sun, Ming Xiao, and Benjamin K. Tsou. 2004. Chinese Word Segmentation without Using Dictionary Based on Unsupervised Learning Strategy. *Chinese Journal of Computers*, 27(6): 736-742 (in Chinese)

Zhongjian Wang, Kenji Araki, and Koji Tochinai. 2002. A Word Segmentation Method with Dynamic Adapting to Text Using Inductive Learning. In *Proceedings of the First SIGHAN Workshop on Chinese Language Processing*, pages 1-5.

Mike West, Peter Müller, and Michael D. Escobar. 1993. *Hierarchical Priors and Mixture Models, with Application in Regression and Density Estimation*. Institute of Statistics and Decision Sciences, Duke University.

Mikio Yamamoto, and Church W. Kenneth. 2001. Using Suffix Arrays to Compute Term Frequency and Document Frequency for All Substrings in a Corpus. *Computational Linguistics*, 27(1): 1-30.

Hua-Ping Zhang, Qun Liu, Xue Q. Cheng, and Hong K Yu. 2003. Chinese Lexical Analysis Using Hierarchical Hidden Markov Model. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, pages 63-70.

Weimeng Zhu, Ni Sun, Xiaojun Zou, and Junfeng Hu. 2013. The Application of Kalman Filter Based Human-Computer Learning Model to Chinese-Word Segmentation. In *Computational Linguistics and Intelligent Text Processing*, pages 218-230.