Comparison of Dictionary Integration Methods for Chinese Word Segmentation Models: Post-print

Authors: Feng Xue

Date: 2018-05-20T00:00:00+00:00

Abstract

Currently, the prevalent Chinese word segmentation methods are machine learning approaches based on statistical models. Statistical-based methods typically employ manually annotated sentence-level corpora for training, yet this approach often neglects existing manually annotated dictionary information accumulated over many years. Such information becomes particularly precious, especially in cross-domain scenarios, due to the scarcity of sentence-level annotated resources in the target domain. Therefore, how to fully and effectively leverage dictionary information within statistical-based models constitutes a highly noteworthy research direction. Recent research has investigated this problem, which can be broadly categorized into two approaches based on how dictionary information is incorporated: one incorporates dictionary features into character-based sequence labeling models, while the other integrates features into word-based beam search models. This work compares these two categories of methods and further combines them. Experimental results demonstrate that after combining these two approaches, dictionary information can be exploited more comprehensively, ultimately yielding superior performance on both in-domain and cross-domain testing.

Full Text

Preamble

http://www.arocmag.com/article/02-2019-01-001.html
ChinaXiv Cooperative Journal *Application Research of Computers*

Comparison of Methods for Integrating Lexicon Information in Chinese Word Segmentation Models

Feng Xue

(School of Computer Science, Beijing Information Science & Technology University, Beijing 100192, China)

Abstract: Currently, the most popular Chinese word segmentation methods are based on statistical machine learning models. These statistical approaches typically use manually annotated sentence-level corpora for training, but often neglect existing manually annotated lexicon resources that have been accumulated over many years. Such resources become particularly valuable when facing cross-domain scenarios, where target domain sentence-level annotated resources are scarce. Therefore, how to fully and effectively utilize lexicon information in statistical models is a highly important research question. Recent work has begun to address this problem, with integration methods broadly falling into two categories: one incorporates lexicon features into character-based sequence labeling models, while the other incorporates features into word-based beam search models. This paper compares these two types of methods and further combines them. Experiments demonstrate that after combination, lexicon information can be more fully exploited, ultimately achieving superior performance on both in-domain and cross-domain tests.

Keywords: Chinese word segmentation; conditional random field; beam search; domain adaptation
Chinese Library Classification: TP391.1
doi: 10.3969/j.issn.1001-3695.2017.05.0643

Comparison of methods for integrating lexicon information in Chinese word segmentation

(School of Computer, Beijing Information Science & Technology University, Beijing 100192, China)

Abstract: Chinese word segmentation is a fundamental task in Chinese natural language processing. Currently, mainstream methods for Chinese word segmentation exploit statistical machine learning models. These methods usually require manually annotated segmented sentences as training corpus, yet have neglected the annotated large-scale lexicon resources which have been built previously—resources that can be highly valuable when cross-domain evaluation is conducted, as gold-standard sentence-level annotations are rare. Recently, the integration of lexicon information into word segmentation models has gained increasing interest. As a whole, integration methods can be classified into two categories: one being based on character-based models that cast the word segmentation problem as sequence labeling, and the other being based on word-based models that use beam-search to decode. In this paper, we compare these two models and combine them. Experimental results on benchmark datasets show that lexicon information can be more fully explored after combination, and finally the combined model can achieve better performances with both in- and cross-domain settings.

Key Words: Chinese word segmentation; conditional random field; beam-search; domain adaptation

0 Introduction

A major difference between Chinese and alphabetic languages is that Chinese text lacks explicit word delimiters. Therefore, Chinese word segmentation is generally the primary task in Chinese natural language processing. Once segmentation is completed, subsequent analyses and applications can proceed within a unified framework similar to other languages. Research on Chinese word segmentation has continued for a long time, and currently the most popular and high-performance methods are based on statistical machine learning.

In statistical models for Chinese word segmentation, mainstream approaches fall into two categories: character-based sequence labeling methods [1-3] and word-based search algorithms [4-5]. The character-based sequence classification approach transforms Chinese word segmentation into a sequence labeling task: each character in a sentence is represented by one of four label types—BMES—where B indicates the first character of a word, M indicates a middle character, E indicates the last character, and S indicates a single-character word. After this transformation, a conditional random field (CRF) model can be employed for training and decoding to complete the Chinese word segmentation task.

Since word-related features are difficult to incorporate in character-based sequence labeling methods, subsequent researchers proposed word-based methods. In this approach, when processing the next character during decoding, the model takes one of two actions: either concatenate it with the previous word or make it the beginning of the next word. During decoding, the model can thus know what specific words have been generated, and these words can also serve as features for the model. Since each character faces two choices, the search space grows exponentially with the number of processed characters; therefore, this method typically employs a beam search algorithm for decoding.

Both of these typical methods utilize sentence-level segmentation information for model training, often neglecting long-accumulated lexicon resources. Moreover, lexicon annotation is actually much easier than sentence-level segmentation annotation. How to fully utilize lexicon resources in statistical Chinese word segmentation models is also an important research question. Obviously, due to significant model differences, the ways to integrate lexicon information in these two typical methods must be different. Zhang Meishan et al. proposed a lexicon integration method for sequence labeling models in 2011 [6]; while in word-based models, lexicon integration is relatively easier—simply checking whether a recently formed word is in the lexicon—an approach also used by Zhang et al. in a joint segmentation and POS-tagging model in 2014 [7].

The research objective of this paper is to systematically compare these two methods of integrating lexicon information, explore their differences, and further combine them to observe whether this combination can yield better results. Finally, this paper compares the two methods experimentally under two settings: same-domain and cross-domain.

1 Related Work

Chinese word segmentation has been extensively studied by researchers [1-5, 8-10]. Early work mainly employed rule-based approaches and language models to score segmentation results. In recent years, statistical machine learning methods have gradually achieved leading performance. This approach includes two categories: character-based sequence labeling methods and word-based methods. These two approaches have their respective advantages and disadvantages. Sun Weiwei conducted a detailed comparison and analysis of these two methods at COLING 2010 [11], and further combined them. This comparative and fusion 思想 is similar to our work. However, the main difference from the aforementioned work is that this paper focuses on the full utilization of lexicon information, proposes fusion specifically for lexicon-based features, and particularly examines the performance of lexicon features under cross-domain conditions.

Integrating lexicon information into statistical models was first proposed by Zhao Hai et al. [12], and further extended by Zhang Meishan et al. [6] to enable this method to function in cross-domain scenarios. Their method was primarily developed on character-based sequence labeling approaches. Subsequently, Zhang et al. [7] also attempted to integrate lexicon features in word-based methods. The 思路 of this paper is consistent with Sun Weiwei et al., but focuses specifically on detailed comparison and further combination of lexicon information integration. Lexicon information integration is crucial for Chinese word segmentation, especially in cross-domain scenarios where training corpora are insufficient, where reasonable integration of this information can yield better performance.

2 Character-Based Sequence Labeling Model

2.1 CRF Chinese Word Segmentation Model

Treating Chinese word segmentation as a sequence labeling problem was first proposed by Xue Nianwen et al. in 2003, with subsequent improvements by other researchers. Currently, a popular setting uses the BMES four-label approach, classifying each character in a sentence according to its position in a word: B indicates the beginning character of a word, M indicates a middle character, E indicates the ending character, and S indicates a word composed of a single independent character.

CRF is the abbreviation for Conditional Random Field, which is currently the most mainstream sequence labeling algorithm because it can achieve leading results on sequence labeling problems. For a given sentence and a segmentation result, where the characters are Chinese characters, the score can be calculated using the following method. This score is actually also a probability value:

$$P(y|x) = \frac{1}{Z(x)} \exp\left(\sum_{i=1}^{n} W \cdot \Phi(x, y_{i-1}, y_i)\right)$$

where $Z(x)$ is the normalization factor, Φ is the feature vector function, and W is the feature weight vector.

Features used in the model mainly include unigram, bigram, and trigram character features, as well as character type features. Specific definitions can be found in Zhang Meishan et al. (2011), and will not be detailed here.

2.2 Integration of Lexicon Information

In statistical machine learning models, the most important component is feature selection, as the core of these methods is learning the weights of features in scoring. Therefore, integrating lexicon information into statistical models essentially becomes how to add lexicon-related features to the model. Here we directly introduce a series of methods proposed by Zhang Meishan et al. in 2011.

Before introducing specific features, three functions need to be defined. For a given sentence, position, and lexicon, considering the j-th character, define the following three functions:

$$f_B^{\max}(x,j,D) = \begin{cases} \max\{l | w = c_j \dots c_{j+l-1} \in D\} & \text{if } j+l-1 \le n \\ 0 & \text{otherwise} \end{cases}$$

$$f_M^{\max}(x,j,D) = \begin{cases} \max\{l | w = c_{j-s} \dots c_{j+l-1} \in D, s < l\} & \text{if } j-s \ge 1 \text{ and } j+l-1 \le n \\ 0 & \text{otherwise} \end{cases}$$

$$f_E^{\max}(x,j,D) = \begin{cases} \max\{l | w = c_{j-l+1} \dots c_j \in D\} & \text{if } j-l+1 \ge 1 \\ 0 & \text{otherwise} \end{cases}$$

where: w represents a word; $f_B^{\max}(x,j,D)$ represents the length of the word obtained by forward maximum matching at position j in sentence x according to lexicon D; $f_M^{\max}(x,j,D)$ represents the length of the longest word passing through position j that does not end at j, obtained by forward maximum matching at some position before j according to lexicon D; and $f_E^{\max}(x,j,D)$ represents the length of the word obtained by backward maximum matching at position j according to lexicon D.

After defining these three functions, lexicon features can be defined using these values. For the i-th position, the features used in this paper mainly include $f_B^{\max}(x,k,D)$, $f_M^{\max}(x,k,D)$, $f_E^{\max}(x,k,D)$ for $k = i, i \pm 1$, and their various combinations, primarily including binary and ternary combinations.

3 Word-Based Beam Search Model

3.1 Basic Model Introduction

The word-based beam search algorithm was first proposed by Zhang Yue and Clark [4] in 2007 and further refined in Zhang Yue and Clark (2011) [5]. It is also commonly called a transition-based model, where the core idea is to transform decoding into an action sequence. Each action execution changes the decoding state. Specifically, each state during decoding consists of a stack and a queue: the stack stores a partially decoded sequence of Chinese words, while the queue stores the sequence of characters yet to be processed, as shown in [Figure 1: see original paper]. Actions are divided into two types: SEPARATE, which moves the first character from the queue into the stack as the beginning of the next word; and APPEND, which appends the first character from the queue to the word at the top of the stack. Decoding starts with an empty stack and a queue containing all characters in the sentence, and ends when the queue is empty, with the result in the stack being the final segmentation.

[Figure 1: see original paper] shows the state diagram defined in the word-based beam search model. During decoding, each state can transform into two states, so when processing the i-th character, the number of possible states generated is 2^i, causing the algorithm' s time and space complexity to grow exponentially. To overcome this drawback, the method employs beam search, retaining only a fixed number of highest-scoring states at each step.

The score calculation formula is relatively simple: directly accumulate features generated by each action, then compute the dot product of the resulting sparse vector with model parameters:

$$\text{score}(a_1, \dots, a_n, x) = \sum_{i=1}^{n} W \cdot \Phi(x, a_i)$$

where W is the model parameter and Φ is the feature extraction function. Specific feature definitions can be found in Zhang Yue and Clark (2011) [5]. Model parameters are trained using the averaged perceptron algorithm, with details omitted here.

The main differences between character-based sequence labeling models and word-based beam search models lie in the models themselves. First, the specific features differ: because character-based models cannot directly access words, feature definitions indirectly reflect word information in the lexicon, while word-based methods can directly compare generated words with those in the lexicon. Second, the feature training methods differ: character-based methods use CRF, which is essentially a probabilistic graphical model, while word-based methods use averaged perceptron, belonging to the max-margin algorithm.

Regarding the combination of these two models, we directly employ a stack-based fusion approach, where the output of one model is passed to the second

model as features, as shown in [Figure 3: see original paper]. This approach has been widely applied in model fusion, as it is theoretically more elegant and can yield better performance. In implementation, we pass the results of the character-based model to the word-based model. While it is also possible to pass word-based model results to the character-based model, preliminary experiments found that this form of combination achieves better performance.

[Figure 2: see original paper] shows the training and decoding patterns of these two models during domain adaptation: users only need to train one model, and during cross-domain testing, simply replace the target domain lexicon during decoding.

[Figure 3: see original paper] shows the stack-based model fusion framework.

3.2 Integration of Lexicon Information

In word-based beam search models, integrating lexicon-related features is much more convenient than in character-based methods because the model can directly see information about generated words. Following Zhang et al. (2014), we check whether the word just generated at the top of the stack appears in lexicon D and determine its length when executing the SEPARATE operation.

5 Experiments

This paper uses the same corpus as Zhang Meishan et al. (2011) for model training and testing, utilizing the PKU training corpus from SIGHAN BAKEOFF 2005. Testing includes two domains: the PKU domain and a financial domain, primarily to evaluate model performance in both in-domain and cross-domain settings. The general lexicon comes from the publicly available lexicon of the Center for Chinese Linguistics at Peking University, containing approximately 100,000 words. The PKU domain lexicon and financial domain lexicon are constructed from the training corpus and extracted/annotated from Baidu Baike finance-related entries, respectively.

We use precision (P), recall (R), and F-measure (F) to evaluate segmentation models, with F-measure being the most important metric.

5.1 In-Domain Performance

First, we observe the performance of lexicon integration in the same domain, comparing it with models that do not use lexicon information. The final results are shown in .

shows the PKU domain segmentation performance comparison. The results demonstrate that character-based and word-based models achieve comparable performance without lexicon information. However, with lexicon information, the word-based model outperforms the character-based model by 0.4%. When the two models are combined, they achieve the best performance both with

and without lexicon integration. Additionally, all three models show significant performance improvement after using lexicon information, confirming the usefulness of lexicon resources.

5.2 Cross-Domain Performance

The previous experiments compared character-based and word-based models with and without lexicon information in the same domain, and presented the performance after combining the two models. Here, we further observe whether the final results in cross-domain scenarios follow the same trend as in-domain settings.

shows the final experimental results. The comparison reveals that cross-domain trends indeed completely match in-domain patterns. Notably, during this part of the experiment, the models were not retrained; the main change was replacing the original PKU domain lexicon with the financial domain lexicon. Additionally, the results show that by integrating external lexicon information, cross-domain performance can be improved by nearly 7%, from approximately 87% to over 94%.

5.3 Model Comparison Analysis

The previous results show that model fusion yields better performance. Here, we further observe the differences between these two models by comparing their error distributions. [Figure 4: see original paper] shows the error distribution of character-based and word-based models with lexicon integration in both in-domain and cross-domain settings. The error distribution is calculated based on individual sentence performance, where the horizontal axis represents the word-based model' s performance and the vertical axis represents the character-based model' s performance, with each scatter point representing a sentence. Observation of [Figure 4: see original paper] reveals that scatter points in both subfigures are widely dispersed and do not fall on a straight line, indicating that the error distributions of the two models are disorderly and demonstrating their differences.

6 Conclusion

This paper introduces, compares, and combines two mainstream word segmentation models in terms of lexicon information integration. We not only provide detailed analysis and comparison from a model construction perspective, pointing out the similarities and differences between the two models in utilizing lexicon information, but also analyze the error distributions of the two models experimentally. The analysis results show that although the two models are relatively similar in lexicon integration methods, certain differences exist, indicating that model fusion can bring further performance improvements. We combine the two models using a stack-based fusion approach, and final experimental results

demonstrate that the combination can more effectively improve overall model performance.

Our findings further validate that lexicon information is highly effective for domain adaptation. However, obtaining lexicons still presents certain difficulties despite past accumulation. Future work will focus on how to automatically acquire high-quality lexicons for specific domains.

References

[1] Xue Nianwen. Chinese word segmentation as character tagging [J]. International Journal of Computational Linguistics and Chinese Language Processing, 2003, 8(1): 29-48.

[2] Tseng H, Chang Pichuan, Andrew G, et al. A conditional random field word segmenter for SIGHAN bakeoff [C]// Proc of the 4th SIGHAN Workshop on Chinese Language Processing. 2005: 168–171.

[3] Lafferty J D, Mccallum A, Pereira F C N. Conditional random fields: Probabilistic models for segmenting and labeling sequence data [C]// Proc of the 18th ICML. San Francisco: Morgan Kaufmann Publishers, 2001: 282–289.

[4] Zhang Yue, Clark S. Chinese segmentation with a word-based perceptron algorithm [C]// Proc of the 45th ACL. 2007: 840–847.

[5] Zhang Yue, Clark S. Syntactic processing using the generalized perceptron and beam search [J]. Computational linguistics, 2011, 37(1): 105-151.

[6] Zhang Meishan, Deng Zhilong, Che Wanxiang, et al. Statistical and lexicon-based domain adaptation for Chinese word segmentation [J]. Journal of Chinese Information Processing, 2012, 26(2): 8–12.

[7] Zhang Meishan, Zhang Yue, Che Wanxiang, et al. Type-supervised domain adaptation for joint segmentation and POS-tagging [C]// Proc of EACL. 2014: 588-597.

[8] Chang Pichuan, Galley M, Manning C D. Optimizing Chinese word segmentation for machine translation performance [C]// Proc of the 3rd Workshop on Statistical Machine Translation. Stroudsburg: Association for Computational Linguistics, 2008: 224-232.

[9] Xu Huating, Zhang Yujie, Yang Xiaohui, et al. Chinese word segmentation domain adaptation based on active learning [J]. Journal of Chinese Information Processing, 2015, 29(5): 55-63.

[10] Liu Yijia, Che Wanxiang, Liu Ting, et al. Comparative analysis of Chinese word segmentation and POS tagging models based on sequence labeling [J]. Journal of Chinese Information Processing, 2013, 27(4): 30-37.

[11] Weiwei Sun. Word-based and character-based word segmentation models: Comparison and combination [C]// Proc of the 23rd International Conference

on Computational Linguistics. Stroudsburg: Association for Computational Linguistics, 2010: 1211-1219.

[12] Zhao Hai, Huang Changning, Li Mu. An improved Chinese word segmentation system with conditional random field [C]// Proc of the 5th SIGHAN Workshop on Chinese Language Processing. 2006: 162-165.

Note: Figure translations are in progress. See original paper for figures.