Statistics and Its Interface Volume 10 (2017) 165–173

Word segmentation in Chinese language processing

Xinxin Shu, Junhui Wang, Xiaotong Shen, and Annie Qu*

This paper proposes a new statistical learning method for word segmentation in Chinese language processing. Word segmentation is the crucial first step towards natural language processing. Segmentation, despite progress, remains under-studied; particularly for the Chinese language, the second most popular language among all internet users. One major difficulty is that the Chinese language is highly context-dependent and ambiguous in terms of word representations. To overcome this difficulty, we cast the problem of segmentation into a framework of sequence classification, where an instance (observation) is a sequence of characters, and a class label is a sequence determining how each character is segmented. Given the class label, each character sequence can be segmented into linguistically meaningful words. The proposed method is investigated through the Peking university corpus of Chinese documents. Our numerical study shows that the proposed method compares favorably with the state-of-the-art segmentation methods in the literature.

AMS 2000 subject classifications: Primary 62H30; secondary 68T50.

Keywords and phrases: Cutting-plane algorithm, Language processing, Support vector machines, Word segmentation.

1. INTRODUCTION

Digital information has become an essential part of modern life, from media news, entertainment, business, distance learning and communication, and research on product marketing, to potential threat detection and national security. With the enormous amount of information gathered nowadays, manual information processing is far from sufficient, and the development of fast automatic processes of information extraction is becoming extremely important.

In this paper, we focus on problems arising from Chinese natural language processing, and in particular we address problems of word segmentation. This is an important area and quite timely, since the Chinese language has become the second most popular language among all internet users. In 2000, there were about 22.5 million Chinese internet users. However, after rapid growth in the last decade, there were over 649 million internet users in 2013 writing text documents in Chinese, consisting of 23.2% of all internet users, compared to 28.6% in English. The online sales and entertainment businesses also promote the popularity of Chinese in the digital world. For example, Amazon.cn data (Zhang et al., 2009) consists of 5×10^5 Chinese reviews on various products.

However, Chinese language processing is still an area which has been severely under-studied. This is likely due to specific challenges caused by the characteristics of Chinese language. Word segmentation is considered a crucial step towards Chinese language processing tasks, due to the unique characteristics of Chinese language structure. Chinese words generally are composed of multiple characters without any delimiter appearing between words. For example, the word 博客 "blog" consists of two characters, 博 "plentiful" and 客 "guest". If characters in a word are treated individually rather than together, this could lead to a completely different meaning. Good word segmenters could correctly transform text documents into collections of linguistically meaningful words, and make it possible to extract information accurately from the documents. Therefore, accurate segmentation is a prerequisite step for Chinese document processing. Without effective word segmentation of Chinese documents, it is extremely difficult to extract correct information given the ambiguous nature of Chinese words.

Existing methods for Chinese segmentation are essentially based on either characters, words or their hybrids (Sun, 2010; Gao et al., 2005). Teahan et al. (2000) proposed a word-based method by applying forward or backward maximum matching strategies. Their method requires an existing corpus as a reference to identify exact character sequences and then segment character by character sequentially, through processing documents in either a forward or backward direction. This method is also developed in Chen and Goodman (1999). One obvious drawback of this approach is that the segmentation heavily relies on the coverage of the given corpus, and thus is not designed for identifying new words which are not in the corpus.

The character-based method considers segmentation as a sequence of labeling problems (Xue, 2003). That is, the location of characters in a word is labeled through statistical modeling such as conditional Gaussian random fields (CRF; Lafferty et al., 2001; Chang et al., 2008) or struc-

*Corresponding author.

tured support vector machines (SVM^{struct}; Tsochantaridis et al., 2005) based on hinge loss. Xue and Shen (2003) proposed a maximum entropy approach which combines both character-based and word-based methods. Specifically, their idea is to integrate forward/backward maximum matching with statistical or machine-learning models to improve segmentation performance. Sun and Xu (2011) proposed a unified approach for a learning segmentation model from both training and test datasets to improve segmentation accuracy.

However, these approaches suffer major drawbacks in that they do not utilize available linguistic information which can enhance the segmentation (Gao et al., 2005), and/or are incapable of identifying new words not appearing in training documents. Some current segmenters treat word segmentation and new word identification as two separate processes (Chen, 2003; Wu and Jiang, 2000), which may lead to inconsistent results in segmentation. Other methods of segmentation are embedded into other processing procedures such as translation, to serve a specific purpose, for example, Chinese-English translation (Xu et al., 2008; Zhang et al., 2008). These methods unified with other processing approaches have not been proposed for general use.

Although in some situations character-based methods tend to outperform word-based methods in terms of segmentation accuracy (Wang et al., 2010), the enormous variety of different permutations of Chinese characters makes the computation of segmentation intractable. In this paper, we propose a statistical method for Chinese word segmentation by incorporating linguistic rules to restrict the possible permutation of Chinese characters and thus alleviate the computation burden. Specifically, an instance (observation) is treated as a character sequence linked with a tag sequence which represents the position of each character as the beginning, middle, or end of a word. Given the tag sequence, each character sequence can then be segmented (broken or grouped) into linguistically meaningful words. The key challenge of segmentation is that it does not have explicit features and the number of choices for tag sequences increases exponentially with the length of the sequence. To circumvent this difficulty, a segmentation function is formulated as a rating function measuring the meaningfulness of the segmented words given each sequence labeling.

Specifically we utilize linguistically meaningful features through higher-order N-gram templates in the segmentation model. Linguistical features using different-order-gram templates are constructed to build the candidate set of features, and select significant features from the candidate set through minimizing a segmentation loss function. The proposed model can achieve higher segmentation accuracy incorporating linguistic rules while reducing the estimation complexity. This is because the proposed segmentation strategy does not completely rely on training samples which cannot identify new words. Instead, it is built on established linguistic rules which can segment new words more accurately, as the linguistic rules can be applied for new words as well.

The paper is organized as follows. Section 2 proposes the linguistically-embedded learning model. Section 3 provides a computational strategy to meet computational challenges in solving large-scale optimization for the proposed model. Section 4 illustrates the proposed method through application to the Peking university corpus in the SIGHAN Bakeoff. The final section provides concluding remarks and a brief discussion.

2. CHINESE LANGUAGE SEGMENTATION

One major challenge in Chinese word segmentation is that the Chinese language is a highly context-dependent or strongly analytic language. The major differences between Chinese and English are listed as follows. Chinese morphemes corresponding to words have very little inflection. English, on the other hand, is rich in equipped and therefore more context-independent. A large number of Chinese words have more than one meaning under different contexts. For example, the original meaning of the word 水分 means "water," but could also mean "inflated;" The word 算账 has double meanings of "balance budget" or "reckoning." Chinese has no tense on verbal inflections to distinguish past such as "-ed," present such as "-ing" and future activities, no number marking such as "-s" in English to distinguish singular versus plural, and no upper or lower case marking to indicate the beginning of a sentence. In addition, English morphemes can have more than one syllable, while Chinese morphemes are typically monosyllabic and written as one character (Wong et al., 2010).

Another challenge is that the number of Chinese characters is much greater than the number of letters in English. The Kangxi dictionary from the Qing dynasty in the 17th century records around 47,035 characters. Nowadays the number of characters has almost doubled to 87,019, according to the Zhonghua Zihai dictionary (Zhonghua Book Company, 1994). Moreover, new Chinese characters are constantly been created by internet users with the exponential speed of the internet in this information age.

In addition, the writing of Chinese characters is not unified because there are two versions of character writing. One is based on traditional characters and the other is simplified character writing. Simplified characters are officially used on the mainland of China, whereas traditional characters are maintained by Taiwan, Hong Kong and Macau. This leads to different coding systems for electronic Chinese documents and webpages. There are three main different coding systems, namely, GB, Big5, and Unicode. The GB encoding scheme is applied to simplified characters, while Big5 is for traditional characters. Unicode can be applied to both writing styles. One advantage of the Unicode system is that both GB and Big5 can be converted into Unicode.

Segmentation in Chinese language processing is a crucial step because there is no boundary delimiter among consec-

Table 1. A character can appear in different positions within different words

position	example
beginning	发生 "to happen"
middle	始发站 "starting station"
end	头发 "hair"

utive Chinese words. In fact, most Chinese characters can appear in any position for different words. Table 1 shows an example where the Chinese character 发 "happen" occurs at three different positions.

This unique feature of the Chinese language makes it quite challenging to determine word boundaries simply through detecting certain types of Chinese characters, even though the number of characters is finite. This is due to the fact that a character appearing in different positions leads to different meaning and interpretation of words and phrases. For instance, a segmenter could segment the sentence 网球拍卖完了 as 网球拍/卖完/了 "Tennis racquets are sold out," or segment the sentence as 网球/拍卖完/了 "Tennis ball(s) is/are auctioned." The ambiguity of the Chinese language is extremely challenging for Chinese language processing since mechanical methods such as tabulating frequencies of key words from the context are not effective for text mining. Therefore segmentation plays a very important role in Chinese language processing, since different segmentation may lead to different sentiment analysis.

2.1 Linguistically-embedded learning framework

In this section, we first introduce a character-based framework, and then illustrate how to incorporate the character-based framework into the proposed model. Let T be the number of characters in one sentence, and the corresponding sentence is denoted as $\mathbf{c} = c_1 \dots c_T$ and the set of its segmentation locators as $\mathbf{s} = s_1 \dots s_T$. Here each character c_t corresponds to a segmentation locator s_t, and $s_t \in \mathcal{S}$. Meng et al. (2010) suggest that a simple 4-tag set $\mathcal{S} = \{B, M, E, S\}$ is sufficient for unique determination and segmentation, where B, M, E, and S denote the beginning, middle, the end of a word, and a single-character word, respectively. For instance, consider the 11-character sentence $\mathbf{c}$ = 我们将创造美好的新世纪 "we will create a bright future," where $T = 11$, c_1 = 我, c_2 = 们,..., c_{11} = 纪, and $\mathbf{s} = BESBEBESBME$. So the linguistically meaningful segmentation is: 我们/将/创造/美好/的/新世纪. The 4-tag segmentation rule is effective in achieving segmentation accuracy and computation efficiency, which are two important and desirable properties in natural language processing.

To identify the segmentation locater for each Chinese sentence, we construct the segmentation model based on training data $(\mathbf{c}_i, \mathbf{s}_i)_{i=1}^n$, mapping from $\phi : \mathcal{C}^T \to \mathcal{S}^T$, where $\mathbf{c}_i$ and $\mathbf{s}_i$ are the character and locater vectors in the i-th sentence, and n is the number of sentences. For instance, $\phi(\{$我,们$\}) = \{B, E\}$ indicating 我们 is segmented as a word. Here ϕ can be a discontinuous and ultra-high-dimensional function when the size of a Chinese document is large. To reduce the dimensionality, we introduce a continuous segmentation function f, which quantifies the appropriateness of segmentation for each sentence. Specifically, $\phi(\mathbf{c}) = \text{argmax}_{\mathbf{s}} f(\mathbf{c}, \mathbf{s})$, and $f(\mathbf{c}, \mathbf{s}) = \sum_{k=1}^K \sum_{t=1}^T \lambda_k f_k(\mathbf{s}, \mathbf{c}, t)$, where $f_k(\mathbf{s}, \mathbf{c}, t)$ is a linguistically meaningful feature measuring the appropriateness of $\mathbf{c}$ at a specific location t of $\mathbf{s}$, K is the number of features and $\Lambda = (\lambda_1, \cdots, \lambda_K)^T$ are the relative importance measures of each feature $f_k(\mathbf{c}, \mathbf{s}, t)$. Usually, $f_k(\mathbf{s}, \mathbf{c}, t)$ takes value in $\{0, 1\}$, and thus the value of $f(\mathbf{c}, \mathbf{s})$ becomes a weighted measure of all features appearing in the segmentation of $\mathbf{s}$ by $\mathbf{c}$. Therefore, one key idea of the proposed method is to learn the relative importance Λ through the training documents, and then apply the estimated $f(\mathbf{c}, \mathbf{s})$ to segment future documents.

To estimate Λ, we minimize the following cost function:

$$\underset{\Lambda}{\text{argmin}} \sum_{i=1}^n L\Big(\sum_{t=1}^T \sum_{k=1}^K \lambda_k f_k(\mathbf{s}_i, \mathbf{c}_i, t)\Big) + \eta \sum_{k=1}^K J(\lambda_k) \quad (1)$$

where $L(u)$ is a large margin loss function that is non-increasing with u, $J(\lambda)$ is a regularizer, and η is a tuning parameter. The large margin loss function can take various forms and gives preference to a large value of $f(\mathbf{c}, \mathbf{s})$, which mimics the optimal rule $\phi(\mathbf{c})$ to achieve good estimation accuracy of Λ. To ensure model sparsity, the choices of $J(\lambda)$ include LASSO (Tibshirani, 1996), SCAD (Fan and Li, 2001), truncated L_1-penalty (Shen et al., 2012), among others. Note that the positive constraint $\lambda_k \geq 0$ is unnecessary in (1) as $f_k(\mathbf{s}, \mathbf{c}, t)$'s are all non-negative and $L(u)$ is non-increasing in u. We obtain $\hat{f}(\mathbf{c}, \mathbf{s})$ with selected important features through minimizing (1), and the sequence $\mathbf{c}$ can be segmented by $\hat{\mathbf{s}} = \hat{\phi}(\mathbf{c}) = \text{argmax}_{\mathbf{s}} \hat{f}(\mathbf{c}, \mathbf{s})$. That is, a document is segmented by maximizing the weighted combination of those important features constructed based on each candidate segmentation. Note that the segmentation formulation in (1) does not produce probabilistic outputs but only the most appropriate segmentation of $\mathbf{s}$. In the literature, a number of methods have been proposed (e.g., Wang et al., 2008; Wu et al., 2010) to construct probabilistic estimates for margin-based methods, which may be adapted for segmentation formulation.

The binary linguistic features $f_k(\mathbf{s}_i, \mathbf{c}_i, t)$ is constructed based on the N-gram templates. The unigram (or 1-gram) templates contain $I(s(0) = s_t, c(-1) = c_{t-1})$, $I(s(0) = s_t, c(0) = c_t)$ and $I(s(0) = s_t, c(+1) = c_{t+1})$, and bigram (or 2-gram) templates include $I(s(0) = s_t, c(-1) = c_{t-1}, c(0) = c_t)$ and $I(s(0) = s_t, c(0) = c_t, c(+1) = c_{t+1})$, where $c(-1)$, $c(0)$, $c(+1)$ and $s(0)$ denote the the previous, current and next characters, and the tag for the current character, re-

spectively. The unigram templates contain the single character's information on the previous, current or next characters given the current segmentation locator, while the bigram templates include two consecutive characters' information through combining the previous or next character with the current character.

For illustration, let $\mathbf{c} =$ 我们将创造美好的新世纪, and $\mathbf{s} = \{BESBEBESBME\}$. If each of any first and last character has 2 unigram and 1 bigram features, and each of any middle character has 3 unigram and 2 bigram features, then this generates 31 unigram and 20 bigram features in total. The higher-order gram templates can be defined in a similar way. However, the more higher-order gram templates are used, the more complex the model will be. Fortunately, mastering around 3000 characters is sufficient for understanding 99% of Chinese documents (Wong et al., 2000). Moreover, the proportions of words with one, two, three and four or more characters are 5%, 75%, 14% and 6% respectively. Therefore, unigram, bigram, trigram and quadrigram templates are sufficient to capture most Chinese words. For lexicon words, as illustrated in the above example, we can apply unigram templates and bigram templates to construct their binary features. For example, in the word 我们 "we" appearing at the beginning of the above sentence, the feature functions are

$$
\begin{aligned}
f_1 &= I(s(0) = B, c(0) = \text{我}),\\
f_2 &= I(s(0) = B, c(+1) = \text{们}),\\
f_3 &= I(s(0) = B, c(0) = \text{我}, c(+1) = \text{们}),\\
f_4 &= I(s(0) = E, c(-1) = \text{我}),\\
f_5 &= I(s(0) = E, c(0) = \text{们}),\\
f_6 &= I(s(0) = E, c(+) = \text{将}),\\
f_7 &= I(s(0) = E, c(-1) = \text{我}, c(0) = \text{们}),\\
f_8 &= I(s(0) = E, c(0) = , c(+1) = \text{将}).
\end{aligned}
$$

To further facilitate the computation, we consider a surrogate loss function $L(f, \mathbf{c}_i, \mathbf{s}_i) = L(\Lambda^T F_{\mathbf{c}_i,\mathbf{s}_i})$, where $\Lambda^T F_{\mathbf{c}_i,\mathbf{s}_i}$ is the generalized function margin for multiclass classification (Vapnik, 1998). We propose to use the hinge loss $L(u) = (1-u)_+$ for segmentation formulation, which is often used in large margin classification such as the support vector machine (SVM; Cortes and Vapnik, 1995). The hinge loss works effectively as a loss function for large margin classification, since the more the margin is violated, the greater the loss is. The proposed formulation with the hinge loss has a number of advantages. First, the model (2) contains only $2n+1$ constraints, which is on a much smaller scale compared to the exponential order of operations required by the conditional random fields (CRF) and structured support vector machine (SVM^{struct}). The CRF combines conditional models with the global normalization of random fields, and the SVM^{struct} solves classification problems involving multiple dependent output variables or structured outputs applicable for complex outputs problems such as natural language parsing. However, both methods involve exponential numbers of constraints. The proposed method makes use of the functional representation in CRF and integrates it in a regularization form as in SVM^{struct}. In a sense, it integrates the strengths of both CRF and SVM^{struct}, leading to a much simpler formulation. Second, the optimization process of (2) can be efficiently implemented through parallel computing and thus make the segmentation scalable. The details of the parallel algorithm to achieve scalable implementation is provided in Section 3. Third and most interestingly, the proposed method has the potential to incorporate various linguistic rules of Chinese in constructing the features, which may lead to an improvement in the segmentation performance. More detailed discussion is deferred to Section 5.

Specifically, the model in (1) with the hinge loss can be formulated as

$$
\begin{aligned}
(2)\quad \operatorname{argmin}_{\Lambda,\xi}\quad & \sum_{i=1}^{n}\xi_i + \eta\sum_{k=1}^{K}J(\lambda_k)\\
\text{s.t.}\quad & 1-\sum_{t=1}^{T}\sum_{k=1}^{K}\lambda_k f_k(\mathbf{s}_i,\mathbf{c}_i,t)\le\xi_i,\quad \xi_i\ge 0,
\end{aligned}
$$

where ξ_i is a slack variable for the hinge loss of each sentence.

In addition, we may also consider alternative surrogate loss functions such as the ψ-loss function $L(u) = \psi(u) = \min(1, (1-u)_+)$ (Shen et al., 2003). Although the ψ-loss function is non-convex, it is able to attain the optimal rate of convergence under certain conditions and outperforms the hinge loss in general. Intuitively, the advantage of the ψ-loss lies in the fact that it is much closer to the 0–1 loss $I(u > 0)$ in identifying segmentation error, especially when u is negative. Consequently, the ψ-loss is much less affected by, e.g., an outlying misclassified sentence with a negative functional margin $\Lambda^T F_{\mathbf{c}_i,\mathbf{s}_i}$. More detailed discussion can be found in Shen et al. (2003).

3. ALGORITHM AND COMPUTATION

It is important to develop an efficient optimization strategy together to improve computational efficiency. The key idea of the proposed computational scheme is based on "decomposition and combination" to meet computational challenges in solving large-scale optimization for the proposed model. The procedure is summarized in Algorithm 1.

We apply the truncated L_1-penalty (Shen et al., 2012) for $J(\Lambda)$, i.e. $J(\cdot) = \min(\|\cdot\|/\nu, 1)$, where ν is a tuning parameter. The truncated L_1-penalty has the following advantages. It selects linguistic features adaptively with Λ and also corrects the Lasso bias through tuning ν. It is capable of handling small Λ of linguistic features through tuning ν and therefore improves accuracy in segmentation. In addition, the truncated L_1-penalty is piecewise linear, and is computationally efficient in the optimization process.

Algorithm 1 Chinese word segmentation procedure based on penalized hinge loss

1: Build features $f_k(\mathbf{s}_i, \mathbf{c}_i, t)$ with N-gram templates for every $k = 1, \cdots, K, i = 1, \cdots, n$ and $t = 1, \cdots, T$.
2: Implement the ad hoc cutting-plane algorithm to get estimates $\hat{\lambda}_k$ for $k = 1, \cdots, K$ by minimizing (2).
3: Predict segmentation locators $\mathbf{s}$ by maximizing $\hat{f}(\mathbf{c}, \mathbf{s}) \equiv \sum_{k=1}^{K} \sum_{t=1}^{T} \hat{\lambda}_k f_k(\mathbf{c}, \mathbf{s}, t)$ for any $\mathbf{c} \in \mathcal{C}$.

The optimization is carried out using an ad-hoc cutting-plane algorithm. The idea of the cutting-plane method is to refine feasible sets iteratively through linear inequalities. Let $\mathcal{M}$ be the set of constraints in model (2) and $\mathcal{W} \subset \mathcal{M}$ be the current working set of constraints. In each iteration, the algorithm finds the solution over the current working set $\mathcal{W}$, searches for the most violated constraint in $\mathcal{M} \backslash \mathcal{W}$, and then adds it to the working set. The algorithm stops until all violations of the constraints are smaller than the tolerance ϵ. The ad-hoc cutting-plane algorithm is illustrated in Algorithm 2.

Algorithm 2 The cutting-plane algorithm for model (2)

1: Initial η, ϵ, set $\mathcal{W} = \emptyset$;
2: **Repeat**
3: Compute $\hat{\Lambda}^{(m)} = \text{argmin}_{\Lambda, \xi} \sum_{i=1}^{n} \xi_i + \eta \sum_{k=1}^{K} J(\lambda_k)$, s.t. $\mathcal{W}$;
4: Obtain the constraint $l^{(m)} \in \mathcal{M}$ which has the largest violation in $\mathcal{M}$ given $\hat{\Lambda}^{(m)}$;
5: Set $\mathcal{W} = \mathcal{W} \cup s_m$;
6: **Until** no violation is larger than ϵ
7: Obtain the estimator $\hat{\Lambda}$.

The computational efficiency of the cutting-plane algorithm for hinge loss has been extensively investigated by empirical studies. Indeed, it is much faster than the conventional training methods derived from decomposition methods (Joachims et al., 2009). Note that model (2) contains a large number of features which are computationally challenging. For example, assuming that $\mathcal{C}$ contains the 1000 most common Chinese characters and the character locator set $\mathcal{S}$ has 4 tags $\{B, M, E, S\}$, the unigram and bigram templates involve $3 \times |\mathcal{S}| \times |\mathcal{C}| + 2 \times |\mathcal{S}| \times |\mathcal{C}| \times |\mathcal{C}|$ or roughly 8×10^6 features. It might be necessary to implement the parallel computing strategy in Step 3 to accelerate the computational speed.

To handle large size documents or texts, MapReduce computation can be utilized to break large problems into many small subproblems in a recursive and parallel manner. In particular, we decompose our cost functions and regularizers for many observations by transforming complicated nonconvex optimization problems to many subproblems of convex minimization. In addition, we can alleviate high storage costs and increase the computational speed through parallel computing. To achieve this goal, we can implement OpenMP, the multi-platform shared-memory parallel programming platform (http://www.openmp.org), or Mahout, a library for scalable machine learning and data mining. These tools for solving large-scale problems allow us to analyze data containing several billions (10^9) of observations on a single machine with reasonable computational speed.

We can also consider other penalty functions in model (2). For example, if the regularizer L_2-norm penalty is applied, then solving model (2) is a convex-function optimization problem. This can be solved by sequential quadratic programming (QP) or linear programming (LP). Solving the LP problem using parallelization has been studied by Dongarra et al. (2002), and QP parallelization can be carried out by a parallel gradient projection-based decomposition method (Zanni et al., 2006). The key idea of QP parallelization is to split an original problem into a sequence of smaller QP subproblems, and parallelize the most demanding tasks of the gradient projection within each QP subproblem. Through QP or LP parallelization, we can process large-scale Chinese text data of size $O(10^7)$.

4. BENCHMARK: PEKING UNIVERSITY CORPUS

In this section, we analyze the corpora obtained from SIGHAN Bakeoff (http://www.sighan.org), which are popular corpora in Chinese language processing competitions. There are four datesets included in SIGHAN's International Chinese Word Segmentation Bakeoff: Academia Sinica (AS), City University of Hong Kong (HK), Peking University (PK) and Microsoft Research corpora (MSR). Each corpus is coded by Unicode and consists of training and test sets. The number of words in each corpus is shown in Table 2. In the Bakeoff corpora, Out-of-vocabulary (OOV) words are defined as words in the test set which are not present in the training set. The contents in the corpora are carefully selected, and domains in the corpora are broadly represented, including politics, economics, culture, law, science and technology, sports, military and literature. Therefore, the corpora are sufficiently representative to assess the performances of Chinese word segmentation.

We use the PK corpus in our experiments to investigate the proposed model. Table 2 shows that the PK corpus is well-balanced in terms of the OOV percentage and the size of the training and test sets, compared to the other three corpora. In the PK corpus, the training set has 161,212 sentences, i.e. n = 161,212, which is about 1.1 million words; while the test set has 14,922 sentences, equivalent to 17 thousand words. Figure 1 displays some randomly selected documents from the training and test sets. Furthermore, approximately 6.9% of the words in the test set are OOV, among which 30% are new words from time-sensitive events, such as 三通 "three links" and 非典 "SARS." In addition, more than 85% of the new words fall into categories of 2-character new word or 2-character word followed by another

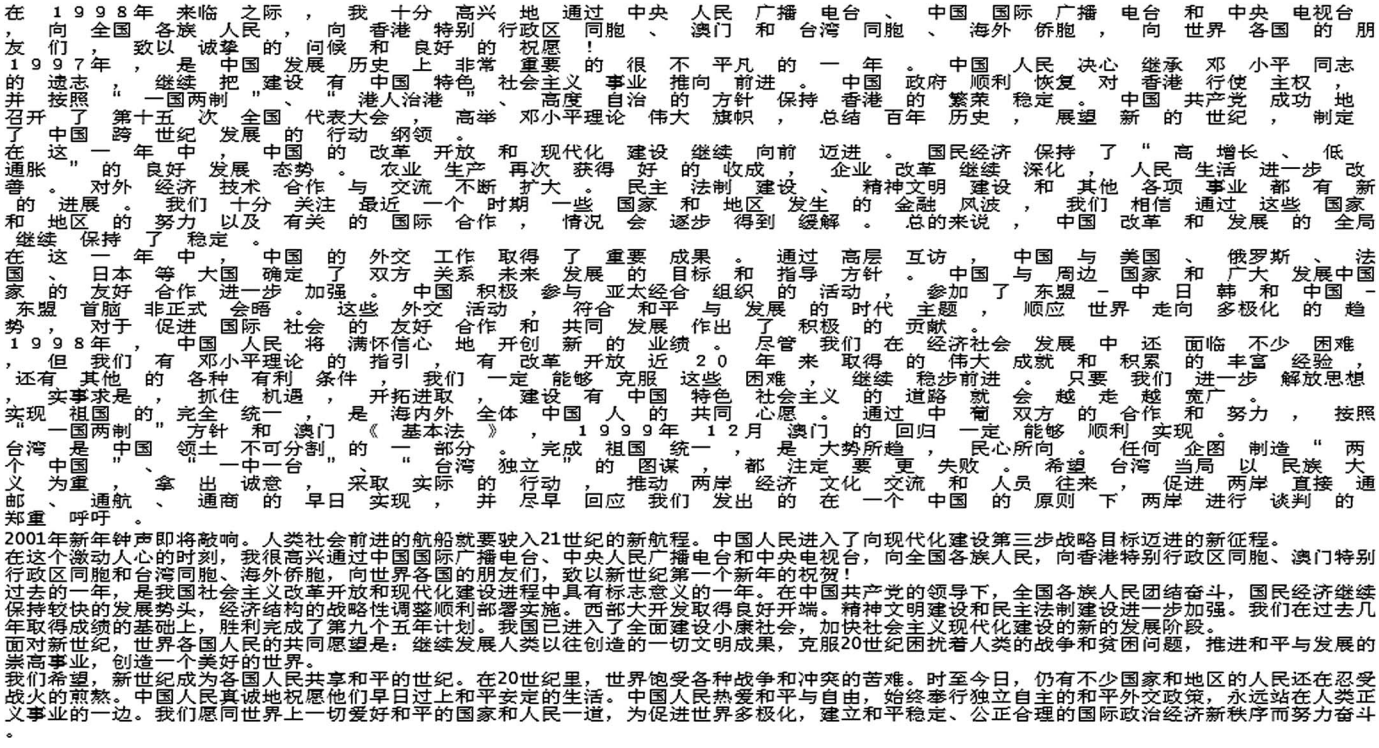

Figure 1. Fragments of the PK training set and test set.

Table 2. SIGHAN's corpora

corpora	# training words (in thousands)	# test words (in thousands)	OOV(%)
AS	5800	12	0.021
HK	240	35	0.071
PK	1100	17	0.069
MSR	20,000	226	0.002

Table 3. Comparison of two top performers and the proposed method

method	precision	recall	F-measure
$method^{12}$	0.929	0.965	0.947
$method^{1234}$	0.951	0.969	0.960
1st in bakeoff	0.952	0.951	0.952
2nd in bakeoff	0.955	0.939	0.947

character. Two fragments from the PK corpus training and test sets are provided in Figure 1.

We test the proposed method based on model (2) using two N-gram templates. The first $method^{12}$ only contains the unigram and bigram templates, while the second $method^{1234}$ utilizes additional trigram and quadrigram templates. We compare these two methods with the two top performers in the Second International Chinese Word Segmentation bakeoff. The performance of Chinese word segmenters is generally reported in terms of three performance metric criteria: precision (P), recall (R) and evenly-weighted F-measure (F). The precision is the fraction of segmented words that are correct, while the recall is the fraction of correct words that are segmented, and the evenly-weighted F-measure is the harmonic mean of the precision and recall defined as $F = (2 * P * R)/(P + R)$.

The segmentation results are shown in Table 3. The proposed method has higher recall values than the two top performers across all situations. In particular, the proposed $method^{12}$ using the unigram and bigram templates attains 92.9% precision, 96.5% recall and 94.7% in the F-measure, which delivers a comparable performance against the two top performers. Note that $method^{12}$ has relatively low precision and high recall values. This is because unigram and bigram templates only utilize the information of the consecutive two characters, and are unlikely to segment words with three or more characters. When the trigram and quadrigram templates are used for the proposed method, a significant improvement is achieved in performance. Specifically, the $method^{1234}$ achieves 95.1% in precision, and delivers the highest recall with 96.9%, and the highest F-measure with 96.0%. In conclusion, the $method^{1234}$ performs the best against the other three methods at an expense of computation time.

Our segmentation for the PK corpus data is computed using an Intel processor with 2 cores, quad CPU at 2.40GHz and 4G ram memory. The quadruple-grams based $method^{1234}$ requires about 1.75 times that of the bi-grams

Table 4. Taxonomy in Chinese words

category	subcategory	examples
LW	lexical word	学生, 照片, 约会
MDW	affixation	老师们
	reduplication	马马虎虎
	splitting	吃了饭
	merging	上下文
	head+morpheme	拿出来
FT	date & time	5月3日, 六月五日, 12点半, 三点二十分
	number & fraction	一千零二十四, 4897, 60%, 百分之一, 1/6
	email & website	johnson@email.com, www.google.com
NE	person name	张三, 约翰
	location name	北京, 上海
	organization name	长城, 大都会博物馆
NW	new word	吐槽, 非典

based *method*[12], because of the complexity of quadruple-grams over bi-grams. Note that run times for the two competitors are not available on the SIGHAN website. The proposed method outperforms the two top performers for the PK corpus in the literature when sufficient N-gram templates are incorporated. However, higher order N-grams require higher computational cost.

5. CONSTRUCTION OF LINGUISTICALLY-EMBEDDED FEATURES

The segmentation accuracy can be further improved by incorporating linguistic language rules into feature $f_k(\mathbf{s}, \mathbf{c}, t)$ construction through word categorization for Chinese words. This categorization method was first introduced by Gao et al. (2005) with five categories: lexical words (LW), morphologically derived words (MDW), factoids (FT), named entities (NE) and new words (NW). The taxonomy in Chinese words is summarized in Table 4.

For morphologically derived words, factoids and named entities, we use trigram and quadrigram templates such that the five main morphological rules are incorporated. The five rules include affixation (e.g., 老师们 "teachers" is teacher + plural), reduplication (e.g., 马马虎虎 "careless" reduplicates and emphasizes word 马虎), splitting (e.g., 吃了饭 "already ate" splits a lexical word 吃饭 "eat" by a particle 了), merging (e.g., 上下文 "context" merges 上文 "above text" and 下文 "following text"), and head morpheme (e.g., 拿出来 "take out" is the head 拿 "take" + the morphemes 出来 "out"). For instance, the head morpheme rule yields trigram template $I(s(0) = B, s(+1) = M, s(+2) = E, c(0) = e_0, (c(+1), c(+2)) = (e_1, e_2))$, where e_0 is a head character such as 拿 "take" and 放 "put," and (e_1, e_2) chooses a value from a set of selected morphemes such as 出来 "out," 进去 "in," or 下来 "down"; the reduplication rule leads to quadrigram templates $I(s(0) = B, s(+1) = M, s(+2) = M, s(+3) = E, c(0) = c(+1), c(+2) = c(+3))$.

Factoid words mainly consist of numeric and foreign characters, such as a number 一千零二十四 "1024" or a foreign organization "FBI." Given a set of numeric and foreign characters $\mathcal{F}$, the factoid words lead to trigram templates $I(s(0) = B, c(-1) \notin \mathcal{F}, (c(0), c(+1)) \in \mathcal{F})$, $I(s(0) = M, (c(-1), c(0), c(+1)) \in \mathcal{F})$, and so on. Named entities include frequently-used Chinese names for persons, locations and organizations. A person's name requires extensive enumeration to identify since it does not follow any language rules. In contrast, names for locations and organizations can be identified by using built-in feature templates. For example, an organization template $I(s(-2) = B, s(-1) = M, s(0) = E, c(0) \in \mathcal{L})$, where $\mathcal{L}$ is a collection of keywords such as 部, 局 and 委员会, "ministry, bureau and committee".

It is much more challenging to identify new words in Chinese segmentation. There is little literature on new word identification, though this has substantial impact on the performance of word segmentation. Therefore, it is important to develop good strategies to detect new words utilizing linguistic rules and more updated language features. For example, enumeration can be used to detect new factoid words and named entities, as discussed above. Linguistic features constructed for the lexicon and morphologically derived words can also be employed to detect new words. Specifically, certain characters are always located at the beginning or at the end of a Chinese word, so new words containing those characters can be easily detected by using the unigram template. For instance, 反 "anti-" typically appears at the beginning of a Chinese word, so the unigram template $I(s(0) = B, c(0) = 反)$ can be used to detect new words such as 反对 "disagree" and 反抗 "resist." In addition, if a new word satisfies the splitting rule as discussed above, trigram templates can be utilized such as $I(s(-1) = B, s(0) = M, s(+1) = E, c(0) = 了)$ for detecting new words like 吐了槽 "already complained".

More importantly, the linguistically-embedded constraints can be integrated into (1), which is powerful for

reducing the effective size of the parameter space through ranking the importance of features. As discussed above, some Chinese characters appear much more frequently at the beginning of a word. For instance, 读 "read," has a chance of 74% to occur in the beginning position (Li et al., 2004). Therefore we can formulate some simple constraints to obtain the relative order of importance measures λ_k's. E.g., for this example, we can assign the importance measure λ_k for $I(s(0) = B, c(0) = 读)$ larger than that associated with $I(s(0) = M, c(0) = 读)$, $I(s(0) = E, c(0) = 读)$ and $I(s(0) = S, c(0) = 读)$. In addition, the existing linguistic rules presented in Section 2.2 should be considered and incorporated into the constraints as well. For example, the merging rule implies that λ_k for 上下文 "context" with $I(s(0) = B, s(+1) = M, s(+2) = E, (c(0), c(+1), c(+2)) =$ 上下文) should be relatively large compared to those associated with other choices of trigram features.

Specifically, the model in (1) with the hinge loss can be formulated as

$$(3)\quad \operatorname{argmin}_{\Lambda,\xi} \quad \sum_{i=1}^{n} \xi_i + \eta \sum_{k=1}^{K} J(\lambda_k)$$

$$\text{s.t.} \qquad 1 - \sum_{t=1}^{T} \sum_{k=1}^{K} \lambda_k f_k(\mathbf{s}_i, \mathbf{c}_i, t) \le \xi_i, \quad \xi_i \ge 0,$$

$$\lambda_k \ge \lambda_j \text{ for all } (k, j) \in \mathcal{I},$$

where ξ_i is a slack variable for the hinge loss of each sentence, and $\mathcal{I}$ is comprised of all available linguistically-embedded constraints. However, the construction of $\mathcal{I}$ requires enumerating all possible language rules manually, and thus can be labor intensive and time consuming. In the current project, we only make use of the standard N-gram features, but leave the linguistically-embedded features as a future development.

6. DISCUSSION

In this paper, we propose a machine-learning framework to utilize linguistically-embedded features for Chinese word segmentation. The proposed model is a character-based method constructing feature functions mapping from characters to segmentation locators in words. The key idea is to build feature functions through N-gram templates, which contain the information of the character itself in conjunction with its consecutive characters. We apply the hinge loss to make the model more scalable, which is effective in reducing the number of constraints and also simplifies the constraint forms.

In addition, computational tractability is one crucial component in segmentation because it requires one to process a large amount of text information in a short time period for real life applications. One important property of the proposed model is that the optimization process can be efficiently implemented through transforming complicated nonconvex optimization problems to many subproblems of convex minimization. This allows one to compute many subproblems of convex optimization in a parallel fashion, and therefore helps to achieve scalable computing for high-volume text data.

Furthermore, there is a trade-off in between the accuracy in segmentation performance and computational complexity. Applying higher-order-gram templates leads to higher accuracy in segmentation, but could result in an increased amount of computational cost. For different corpora, the order of grams needs to be selected carefully to meet the demand of time constraints for real-life applications. Finally, the segmentation for new words is still a challenging problem, and further research is needed on developing segmentation strategies to incorporate more complex linguistic rules.

ACKNOWLEDGEMENTS

Research supported in part by National Science Foundation Grants DMS-1207771, DMS-1415500, DMS-1415308, DMS-1308227, DMS-1415482, and HK GRF-11302615. The authors thank the editors, the associate editor and the reviewers for helpful comments and suggestions.

Received 2 December 2015

REFERENCES

Chang, P., Galley, M., and Manning, C. (2008). Optimizing Chinese word segmentation for machine translation performance. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pp. 224–232.

Chen, A. (2003). Chinese word segmentation using minimal linguistic knowledge. *SIGHAN*, 148–151.

Chen, S. and Goodman, J. (1999). An empirical study of smoothing techniques for language modeling. *Computer Speech and Language* **13**, 359–394.

Cortes, C. and Vapnik, V. (1995). Support vector networks. *Machine Learning* **20**, 273–297.

Dongarra, J., Foster, I., Fox, G., Gropp, W., Kennedy, K., Torczon, L., and White, A. (2002). *The Sourcebook of Parallel Computing.* Morgan Kaufmann, San Francisco.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of American Statistical Association* **96**, 1348–1360. MR1946581

Gao, J., Li, M., Wu, A., and Huang, C. (2005). Chinese word segmentation and named entity recognition: A pragmatic approach. *Computational Linguistics Journal* **31**, 531–574.

Joachims, T., Finley, T., and Yu, C. (2009). Cutting-plane training of structural SVMs. *Machine Learning* **27**, 27–59.

Lafferty, J., McCallum, A., and Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning*, Morgan Kaufmann, pp. 282–289.

Li, H., Huang, C., Gao, J., and Fan, X. (2004). The use of SVM for Chinese new word identification. In *Proceedings of the 1st International Joint Conference on Natural Language*, Springer, pp. 497–504.

Meng, W., Liu, L., and Chen, A. (2010). A comparative study on Chinese word segmentation using statistical models. In *IEEE International Conference on Software Engineering and Service Sciences*, pp. 482–486.

Shen, X., Pan, W., and Zhu, Y. (2012). Likelihood-based selection and sharp parameter estimation. *Journal of the American Statistical Association* **107**, 223–232. MR2949354

Shen, X., Tseng, G., Zhang, X., and Wong, W. (2003). On ψ-learning. *Journal of the American Statistical Association* **98**, 724–734. MR2011686

Sun, W. (2010). Word-based and character-based word segmentation models: Comparison and combination. In *Proceedings of the 23rd International Conference on Computational Linguistics*, 1211–1219.

Sun, W. and Xu, J. (2011). Enhancing Chinese word segmentation using unlabeled data. *Empirical Methods in Natural Language Processing*.

Teahan, W., Wen, Y., and Witten, I. (2000). A compression-based algorithm for Chinese word segmentation. *Computational Linguistics* **26**, 375–393.

Tibshirani, R. (1996). Regression shrinkage and selection via the LASSO. *Journal of Royal Statistical Society Series B* **58**, 267–288. MR1379242

Tsochantaridis, I., Joachims, T., Hofmann, T., and Altun, Y. (2005). Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research* **6**, 1453–1484. MR2249862

Vapnik, V. (1998). *Statistical learning theory*, Chichester, UK, Wiley. MR1641250

Wang, J., Shen, X., and Liu, Y. (2008). Probability estimation for large margin classifiers. *Biometrika* **95**, 149–167. MR2409720

Wang, K., Zong, C., and Su, K. (2010). A character-based joint model for Chinese word segmentation. In *Proceedings of the 23rd International Conference on Computational Linguistics*, 1173–1181.

Wong, K. F., Li, W., Xu, R., and Zhang, Z.-S. (2010). *Introduction to Chinese Natural Language Processing*. Morgan & Claypool Publishers.

Wu, A. and Jiang, Z. (2000). Statistically-enhanced new word identification in a rule-based Chinese system. In *Proceeding of the 2nd ACL Chinese Processing Workshop*, 41–66.

Wu, Y., Zhang, H. H., and Liu, Y. (2010). Robust model-free multiclass probability estimation. *Journal of the American Statistical Association* **105**, 424–436. MR2656060

Xu, J., Gao, J., Toutanova, K., and Ney, H. (2008). Bayesian semi-supervised Chinese word segmentation for statistical machine translation. In *Proceedings of the 22nd International Conference on Computational Linguistics* **1**, pp. 1017–1024.

Xue, N. (2003). Chinese word segmentation as character tagging. *Computational Linguistics and Chinese Language Processing* **8**, 29–48.

Xue, N. and Shen, L. (2003). Chinese word segmentation as LMR tagging. In *Proceedings of 2nd SIGHAN Workshop on Chinese Language Processing*, pp. 176–179.

Zanni, L., Serafini, T., and Zanghirati, G. (2006). Parallel software for training large scale support vector machines on multiprocessor systems. *Journal of Machine Learning Research* **7**, 1467–1492. MR2274413

Zhang, C., Zeng, D., Li, J., Wang, F., and Zuo, W. (2009). Sentiment analysis of Chinese documents: From sentence to document level. *Journal of the American Society for Information Science and Technology* **60**, 2474–2487.

Zhang, R., Yasuda, K., and Sumita, E. (2008). Chinese word segmentation and statistical machine translation. *ACM Transactions on Speech and Language Processing* **5**, 4:1–4:19.

Zhonghua Zihai Dictionary. Zhonghua Book Company, 1994.

Xinxin Shu
Department of Biostatistics and Research Decision Sciences
Merck Research Laboratories
USA
E-mail address: xinxin.shu@merck.com

Junhui Wang
Department of Mathematics
City University of Hong Kong
People's Republic of China
E-mail address: j.h.wang@cityu.edu.hk

Xiaotong Shen
School of Statistics
University of Minnesota
USA
E-mail address: xshen@umn.edu

Annie Qu
Department of Statistics
University of Illinois at Urbana-Champaign
USA
E-mail address: anniequ@illinois.edu