Multi-Grained Chinese Word Segmentation

Chen Gong, Zhenghua Li*, Min Zhang, Xinzhou Jiang
Soochow University, Suzhou, China
cgong@stu.suda.edu.cn, {zhli13,minzhang}@suda.edu.cn, xzjiang.hw@gmail.com

Abstract

Traditionally, word segmentation (WS) adopts the single-granularity formalism, where a sentence corresponds to a single word sequence. However, Sproat et al. (1996) show that the inter-native-speaker consistency ratio over Chinese word boundaries is only 76%, indicating single-grained WS (SWS) imposes unnecessary challenges on both manual annotation and statistical modeling. Moreover, WS results of different granularities can be complementary and beneficial for high-level applications.

This work proposes and addresses multi-grained WS (MWS). First, we build a large-scale pseudo MWS dataset for model training and tuning by leveraging the annotation heterogeneity of three SWS datasets. Then we manually annotate 1,500 test sentences with true MWS annotations. Finally, we propose three benchmark approaches by casting MWS as constituent parsing and sequence labeling. Experiments and analysis lead to many interesting findings.

1 Introduction

As the first processing step of Chinese language processing, word segmentation (WS) has been extensively studied and made great progress during the past decades, thanks to the annotation of large-scale benchmark datasets, among which the most widely-used are Microsoft Research Corpus (MSR) (Huang et al., 2006), Peking University People Daily Corpus (PPD) (Yu et al., 2003), and Penn Chinese Treebank (CTB) (Xue et al., 2005). Table 1 gives an example sentence segmented in different guidelines. Meanwhile, WS approaches gradually evolve from maximum matching based on lexicon dictionaries (Liu and Liang, 1986), to path searching from segmentation graphs based on language modeling scores and other statistics (Zhang and Liu, 2002), to character-based sequence labeling (Xue, 2003), to shift-reduce incremental parsing (Zhang and Clark, 2007). Recently, neural network models have also achieved success by effectively learning representation of characters and contexts (Zheng et al., 2013; Pei et al., 2014; Ma and Hinrichs, 2015; Chen et al., 2015; Zhang et al., 2016; Cai and Zhao, 2016; Liu et al., 2016).

MSR	全国各地				医学	界	专家	走出		人民大会堂				
PPD	全国		各地		医学界		专家	走	出	人民	大会堂			
CTB	全	国	各	地	医学界		专家	走出		人民	大会堂			

Table 1: An example of annotation heterogeneity: 全 (all) 国 (country) 各 (every) 地 (place) 医学 (medical science) 界 (field) 专家 (experts) 走 (walk) 出 (out) 人民 (people) 大会堂 (great hall).

To date, all the labeled datasets adopt the single-granularity formalization, and previous research mainly focuses on single-grained WS (SWS), where one sentence is segmented into a single word sequence. Although different WS guidelines share the same high-level criterion of word boundaries – a character string combined closely and used steadily forms a word, people greatly diverge due to individual differences on knowledge and living environments, etc. An anonymous reviewer kindly points

* Correspondence author

Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 701–712
Copenhagen, Denmark, September 7–11, 2017.

out that Vladímir Skalička of the Prague School claimed that unlike the "isolating" languages such as French and English, Chinese belongs to the "polysynthetic" type, in which compound words are normally produced from indigenous morphemes (Jernudd and Shapiro, 1989). The vague distinction between morphemes and compounds also contribute to the cognition divergence on the concept of words. Sproat et al. (1996) show that the consensus ratio over word boundaries is only 76% among Chinese native speakers without trained on a common guideline.To fill this gap, WS guidelines need to further group words into many types and provide illustration examples for each type. Nevertheless, it is very challenging even for well-trained annotators to fully grasp the guidelines and to be consistent on uncovered cases. For example, Xiu (2013) (in Tables 1-3) shows that about 3% characters are inconsistently segmented in the PPD training data used in SIGHAN Bakeoff 2005 (Emerson, 2005). We have also observed many inconsistency cases in all MSR/PPD/CTB during this work. In a word, SWS imposes great challenge on data annotation, and as a side effect, enforces statistical models to learn subtleness of annotation guidelines rather than the true WS ambiguities.

From another perspective, WS results of different granularities may be complementary in supporting applications such as information retrieval (IR) (Liu et al., 2008) and machine translation (MT) (Su et al., 2017). On the one hand, coarse-grained words enable statistical models to perform more exact matching and analyzing. On the other hand, fine-grained words are helpful in both reducing data sparseness and supporting deeper understanding of language.[1]

To solve the above two issues for SWS, this paper proposes and addresses multi-grained WS (MWS). Given an input sentence, the goal is to produce a hierarchy structure of all words of different granularities, as illustrated in Figure 1. To tackle the lack of labeled data, we build a large-scale pseudo MWS dataset for model training and tuning by automatically converting annotations of three heterogeneous

[1] Words in CTB are generally more fine-grained than those in PPD and MSR, probably due to the requirement of annotating syntactic structures.

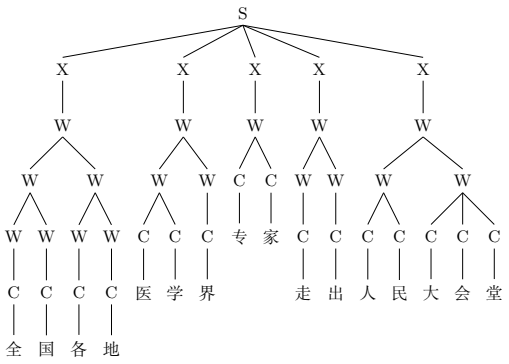

Figure 1: MWS as a constituent parse tree.

SWS datasets (i.e. MSR/PPD/CTB) based on the recently proposed coupled sequence labeling approach of Li et al. (2015). In order to fully investigate the problem, we manually annotate 1,500 test sentences with true MWS annotations. Finally, we propose three benchmark approaches by casting MWS as constituent parsing and sequence labeling problems. Experiments and data analysis lead to many interesting findings.

We will release the newly annotated data and the codes of the benchmark approaches at `http://hlt.suda.edu.cn/~zhli`. However, due to the license issue, we may not directly release all the pseudo MWS datasets. Instead, we will launch a web service for obtaining MWS annotations given a sentence with one of MSR/PPD/CTB annotations.

2 Pseudo MWS Data Conversion

This section introduces the process of gathering pseudo MWS data by making use of the annotation heterogeneity of the three existing datasets, i.e., MSR/PPD/CTB.

2.1 Annotation Heterogeneity

MSR is a manually labeled corpus with word boundaries and named entity tags, and is annotated by Microsoft Research Asia for supporting Chinese text processing (Huang et al., 2006). The key characteristic of MSR is treating named entities as single words. For example, "人民大会堂 (Great Hall of the People)" is a location and forms a word in Table 1. In general, MSR is more coarse-grained than PPD and CTB. PPD is a large-scale corpus with word boundaries, POS tagging, and phonetic notations to facilitate Chi-

nese information processing, and is annotated by Institute of Computational Linguistics at Peking University (Yu et al., 2003). Based on the Penn Chinese Treebank Project, CTB is built to create a Mandarin Chinese corpus with syntactic bracketing (Xue et al., 2005). We find that CTB is more fine-grained in word boundaries than MSR and PPD, since syntactic annotation tends to require deeper understanding of a sentence. For example, Table 5 reports the averaged number of characters per word in each corpus, and confirms our observations.

For better understanding of annotation heterogeneity, we summarize high-frequency differences among the three datasets observed and gathered during this study in Appendix A. However, it is difficult to obtain a complete list of annotation correspondences among the three datasets, since there are too many low-frequency and irregular cases. Moreover, we also observe a lot of inconsistency annotations of the same word or words with similar structures in all three datasets, as shown in Appendix B.

2.2 Coupled WS for Conversion: MSR/PPD as Example

This section introduces how to automatically produce high-quality PPD-side WS labels for a sentence with MSR-side gold-standard WS labels, by leveraging the two non-overlapping SWS data of MSR and PPD with the coupled sequence labeling approach of Li et al. (2015) and Li et al. (2016). Figure 2 shows the workflow.

Given a sentence $\mathbf{x} = [c_1, ..., c_i, ..., c_n]$, the coupled model aims to produce a sequence of bundled tags $\mathbf{t} = [t_1^a t_1^b, ..., t_i^a t_i^b, ..., t_n^a t_n^b]$, where t_i^a and t_i^b are two labels corresponding to two heterogeneous guidelines respectively. Table 2 gives an example of coupled WS on MSR/PPD. We employ the standard four-tag label set to mark word boundaries of one granularity, among which `B`, `I`, `E` respectively represent that the concerned character situates at the *beginning, inside, end* position of a word, and *S* represents a single-character word. The bottom row shows the gold-standard bundled tag sequence.

One key advantage of the coupled model is to directly learn from two *non-overlapping* heterogeneous training datasets, where each dataset only contains single-side gold-standard labels. To deal with this partial (or incomplete) labeling issue, they project each single-side label to a set of bundled labels by considering all labels at the missing side, as shown in the second row in Table 2. Such *ambiguous labelings* are used for model supervision.

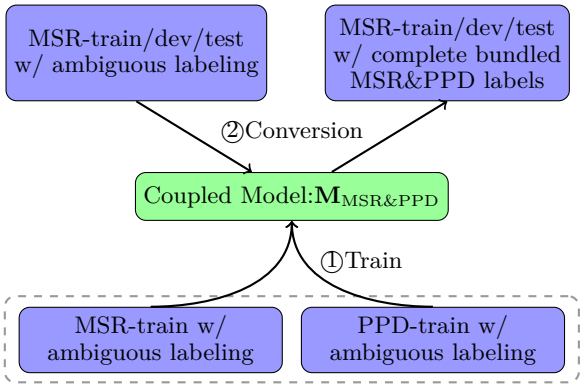

Figure 2: Conversion between MSR/PPD.

Input	全	国	各	地	医	学	界	专	家	...
Ambiguous	BB	IB	IB	EB	BB	EB	SB	BB	EB	...
Labeling for	BI	II	II	EI	BI	EI	SI	BI	EI	...
Training &	BE	IE	IE	EE	BE	EE	SE	BE	EE	...
Conversion	BS	IS	IS	ES	BS	ES	SS	BS	ES	...
Output	BB	IE	IB	EE	BB	EI	SE	BB	EE	...

Table 2: Coupled WS (MSR/PPD as example). Two WS labels are bundled to represent MSR/PPD annotations for a character. Ambiguous labeling is gained supposing this sentence has MSR-side gold-standard annotations.

Under a traditional CRF, the coupled model defines the score of a bundled tag sequence as

$$Score(\mathbf{x}, \mathbf{t}; \theta) = \theta \cdot \mathbf{f}(\mathbf{x}, \mathbf{t}) = \sum_{i=1}^{n+1} \theta \cdot \begin{bmatrix} \mathbf{f}_{joint}(\mathbf{x}, i, t_{i-1}^a t_{i-1}^b, t_i^a t_i^b) \\ \mathbf{f}_{sep\_a}(\mathbf{x}, i, t_{i-1}^a, t_i^a) \\ \mathbf{f}_{sep\_b}(\mathbf{x}, i, t_{i-1}^b, t_i^b) \end{bmatrix}$$

where $\mathbf{f}_{joint}(.)$ are the *joint features* whereas $\mathbf{f}_{sep\_a/sep\_b}(.)$ are the separate features. Li et al. (2015) demonstrate that the joint features capture the implicit mappings between heterogeneous annotations, while the back-off separate features work as a remedy for the sparseness of the joint features.

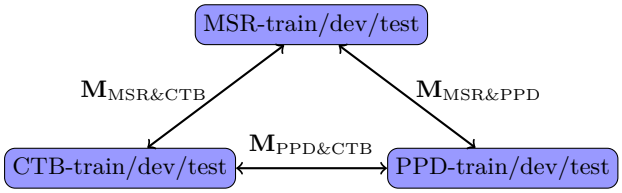

Figure 3: Producing pseudo MWS data.

In their case study of POS tagging, Li et al. (2015) show the coupled model improves tagging accuracy by $95.0 - 94.1 = 0.9\%$ on CTB5-test over the baseline non-coupled model trained on a single training data.

More importantly, they show that the coupled model can be naturally used for the task of annotation conversion, where second-side labels are automatically annotated, given one-side gold-standard labels. The given one-side tags are used to obtain ambiguous labelings, as shown in Table 2, and the coupled model finds the best bundled tag sequence in the *constrained search space*, instead of in the whole bundled tag space, hence greatly reducing the difficulty. Li et al. (2015) report that the coupled model can improve conversion accuracy on POS tagging by $93.9 - 90.6 = 3.3\%$ over the non-coupled model.[2]

2.3 Producing Pseudo MWS Data

Figure 3 shows the workflow of producing pseudo MWS data with three separately trained coupled models. Please note that one coupled model is able to perform conversion between one pair of annotation standards, and thus three coupled models are required for three kinds of annotation standards. Another alternative is that we could directly train one coupled model on MSR/PPD/CTB by extending the approach of Li et al. (2015) from two guidelines into three, which would lead to a much larger bundled tag space. For simplicity, we directly employ their released codes in this work, and leave that for future exploration.

After conversion, we obtain 9 pseudo MWS datasets (i.e., MSR/PPD/CTB-train/dev/test) and represent each sentence in a hierarchy structure as shown in Figure 1. Please kindly note that the guideline-specific information are thrown away, since we do not care which word belongs to which guideline.

In the resulting pseudo MWS data, we find about 0.08% of words overlap with other words, meaning a string "ABC" is segmented into "A/BC" and "AB/C" in two different annotations. We have manually checked these words, and find almost all those cases are caused by conversion errors. This confirms that our treatment of MWS as a hierarchy structure is reasonable.

3 Manual Annotation

In order to fully investigate the MWS problem, we have manually created a true MWS data of 1,500 sentences for final evaluation. From each test dataset in Table 5, we randomly sample 500 sentences with converted pseudo MWS annotations for manual correction. First, two coauthors of this work spent about two hours each day on manual correction of the pseudo MWS annotations for two weeks. During this period, we have summarized a list of high-frequency corresponding patterns among the three guidelines (see Appendix A), and have also written a simple program to automatically detect inconsistent annotations of given words in different training datasets, so that annotators can use the outputs of the program to decide ambiguous cases, which we find is extremely helpful for annotation.

Then, we employ 10 postgraduate students as our annotators who are at different familiarity in WS annotation. Before formal annotation, the annotators are trained for two hours on the basic concepts of MWS, high-frequency correspondences among the three guidelines, and the use of the outputs of the program. We also encourage the annotators to access the three training datasets directly for studying concrete cases under real contexts. Moreover, annotators are asked to recheck their annotations before final submission to improve quality.

To measure the inter-annotator consistency, 150 sentences (10%) are sampled for double annotation, and are grouped into four batches for four pairs of annotators. After annotation, two annotators on the same batch compare

[2]The accuracy seems quite low. The reason is only the 20% most ambiguous words of each sentence are manually labeled and evaluated in their experiments.

	#Words	Granularities Distrbution (%)		
		Single	Two	Three
Before	44,593	74.5	24.0	1.5
After	45,279	71.6	26.8	1.6

Table 3: Data statistics of the MWS test data before and after manual annotation.

their results and produce a consensus submission through discussion.

The annotation process lasts for four days, and each annotator spends about 8 hours in total on completing 160 sentences on average. Table 3 compares data statistics on the 1,500 sentences before and after manual annotation. The second column reports the number of words, and the last three columns report the distribution of words according to their granularity levels. To illustrate how to gain the distribution, we take Figure 1 as an example, which contains 1 single-grained words, 9 two-grained words, and 7 three-grained words.[3]

Table 3 shows that only 71.6% of all words are single-grained, which is somehow roughly consistent with the inter-native-speaker consistency ratio (76%) in Sproat et al. (1996). Among multi-grained words, $\frac{26.8}{26.8+1.6} = 94.4\%$ are two-grained. It is clear that manual annotation increases both the number of words by $\frac{45,279-44,593}{44,593} = 1.5\%$, and the number of multi-grained words by $74.5 - 71.6 = 2.9\%$. In fact, during annotation, we also feel that multi-granularity phenomena are under-represented in the pseudo MWS data. The reason may be two-fold. First, the conversion models incline to suppress granularity differences, since most words have the same granularity in different datasets. Second, the exist of many inconsistencies in the same dataset also makes the conversion models more reluctant to produce multi-grained words.

The inter-annotator consistency ratio is $\frac{3859}{3935} = 98.07\%$, where the denominator is the word number after merging the submission of all annotator pairs, and the numerator is the consensus word number. We argue that the consistency ratio is not high, considering most words do not need correction in the pseudo MWS annotations. In fact, we find that this annotation task is actually very difficult, since the annotators must consider three guidelines simultaneously. The main inconsistency source of all four annotator pairs are due to the situation where one annotator notices a mistake while another annotator overlooks it. To solve this issue, our long-term plan is to compile a unified MWS guideline by integrating existing SWS guidelines, and gradually improve it by more manual MWS annotation.[4]

4 Benchmark MWS Approaches

There has recently been a surge of interest in applying neural network models to both parsing and sequence labeling tasks. In this work, we propose three simple benchmark approaches for MWS, inspired by recently neural models for constituent parsing (Cross and Huang, 2016) and SWS (Pei et al., 2014).

4.1 MWS as Constituent Parsing

Due to its hierarchy structure shown in Figure 1, we naturally cast MWS as a constituent parsing problem, where characters are leaf nodes; "C" represent a character, "W" represent a word; "X" means that the spanning word cannot be further merged into a more coarse-grained word.

We employ the recently proposed transition-based constituent parser of Cross and Huang (2016) due to its simplicity and competitive performance on different parsing benchmark datasets. In the transition system, a stack S stores processed tokens and partial trees collected so far; a queue Q contains unprocessed tokens; structural[5] and labeling[6] decisions are alternatively made to advance the state until a complete tree forms. The network architecture is composed of two parts: 1) two cascaded

[3] Formally, we call a word s three-grained if there are two other words $s1$ and $s2$ satisfying any one conditions: 1) $s2 \in s1 \in s$ (like "全国各地" in Figure 1); 2) $s2 \in s \in s1$ (like "全国"); 3) $s \in s1 \in s2$ (like "全"), where $\in$ means substring. The definition of two-grained words is analogous; otherwise single-grained.

[4] Although this work has been confined to the three guidelines of MSR/PPD/CTB, we feel that the three guidelines can well capture most multi-granularity phenomena of words. During manual annotation, we have found very few cases where an obvious multi-granularity structure is not covered by the three guidelines.

[5] Shifting the first token in Q into S, or combining the top two items in S

[6] Assigning a non-terminal label or "NULL" to the top item in S

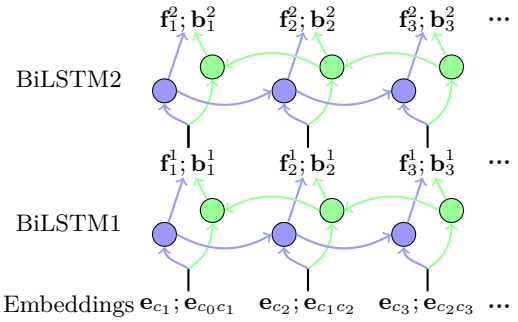

Figure 4: Two-layer BiLSTM architecture.

Chars	全	国	各	地	医	学	界	专	家	...
MWS labels	SBB	SEI	SBI	SEE	BB	EI	SE	B	E	...

Table 4: MWS as sequence labeling. SWS labels for the same character are organized fine-to-coarse.

bidirectional LSTM layers to encode the input token sequence, as shown in Figure 4; 2) two separate multilayer perceptrons (MLPs) to make structural/labeling decisions based on 4/3 simple LSTM span features. A span feature represents a sentence span (i, j) by concatenating the element-wise differences of BiLSTM outputs:

$$\mathbf{r}_{(i,j)} = [\mathbf{f}^1_j - \mathbf{f}^1_{i-1}; \mathbf{b}^1_i - \mathbf{b}^1_{j+1}; \mathbf{f}^2_j - \mathbf{f}^2_{i-1}; \mathbf{b}^2_i - \mathbf{b}^2_{j+1}]$$

To adapt the original parsing model to our MWS task, we concatenate bichar embeddings $\mathbf{e}_{c_{i-1}c_i}$ with single char embedding $\mathbf{e}_{c_i}$ as inputs to the first-layer BiLSTM, inspired by Pei et al. (2014), who show that bichar embeddings are very helpful for SWS.

4.2 MWS as Sequence Labeling

It is also straightforward to model MWS as a sequence labeling task by replacing SWS labels with MWS labels for each character. Table 4 encodes the MWS structure in Figure 1 with a sequence of MWS labels. The idea is to concatenate multiple SWS tags simultaneously for one character to denote the positions of the character under words of different granularities. Please note that each MWS label contains at most three SWS labels since we only consider three SWS datasets in this work. Here, we organize the SWS labels in the order of fine-to-coarse granularities.

For simplicity and fair comparison, we adopt a similar network architecture as the parsing model described in Section 4.1. To decide the MWS label of a character c_i in the input sentence, we feed the outputs of the two-layer BiLSTM outputs $[\mathbf{f}^1_i; \mathbf{b}^1_i, \mathbf{f}^2_i; \mathbf{b}^2_i]$ into a single-hidden-layer MLP.

	Train	Dev	Test	#Char per Word
MSR	78,232	8,692	3,985	1.71
PPD	46,815	2,000	5,000	1.67
CTB	16,091	803	1910	1.63

Table 5: Data statistics (in sentence number). The last column reports the averaged character number of each word.

4.3 MWS as SWS Aggregation

Instead of directly training a MWS model on the three pseudo MWS training datasets, we can also train three separate SWS models on the three SWS training datasets. Given an input sentence, we apply the three SWS models and then merge their outputs as MWS results.

The network architecture is the same with the sequence labeling model in Section 4.2, except the MLP outputs correspond to SWS labels instead of MWS labels.

5 Experiments

Data: for MSR, we adopt the training/test datasets of the SIGHAN Bakeoff 2005 (Emerson, 2005), and cut off 10% random training sentences as the dev data following Zhang et al. (2016); for PPD and CTB, we follow Li et al. (2015) and directly adopt their datasets and data split. Table 5 shows the data statistics.[7]

Evaluation Metrics: the goal of MWS is to precisely produce all words of different granularities given the input sentence. Therefore, to reach a balance of both precision ($P = \frac{\#Word_{gold \cap sys}}{\#Word_{sys}}$) and recall ($R = \frac{\#Word_{gold \cap sys}}{\#Word_{gold}}$), we use the F1 score ($= \frac{2PR}{P+R}$) as in SWS.

Hyper-parameter: we implement all our approaches based on the codes released by Cross and Huang (2016), by making extensions such as adding bichar embeddings and

[7] A DBC-to-SBC (double/single-byte characters) case preprocessing is performed on all datasets to avoid encoding inconsistency.

	Dev (Pseudo)			Test (Manual)							
	P	R	F	P	R	F	#Words	Single	Two	Three	Overlapping
Parsing	96.55	96.40	96.48	97.00	95.16	**96.07**	44,408	74.9%	23.5%	1.6%	–
w/o Bichar Emb	95.58	95.04	95.51	96.37	94.11	95.22	44,434	74.2%	24.1%	1.7%	–
Sequence Labeling	**96.86**	96.26	**96.59**	**97.01**	94.96	95.97	44,323	75.8%	22.7%	1.5%	–
w/o Bichar Emb	95.88	94.94	95.41	96.56	94.18	95.35	44,162	75.7%	22.8%	1.5%	–
SWS Aggregation	90.43	**97.44**	93.80	92.11	**96.59**	94.30	47,478	64.6%	31.4%	4.0%	1.0%

Table 6: Performance of different MWS approaches.

supporting sequence labeling.[8] For simplicity, char and bichar embeddings are randomly initialized following Cross and Huang (2016). The dimensions of char and bichar embeddings are both 50 and other hyper-parameters are the same with Cross and Huang (2016). In our preliminary experiments, we observe that under their neural network framework, the MWS performance is quite stable when reruning under random initialization or reasonably altering other hyper-parameters. Due to time limitation, we leave the use of pre-trained embeddings and more hyper-parameter tuning for future exploration.

Training/test settings: when training the parsing and sequence labeling based MWS models (not SWS aggregation) on MSR/PPD/CTB-train, we adopt the simple corpus weighting strategy used in Li et al. (2015) to balance the contributions of each training dataset. Before each iteration, we randomly sample 10,000 sentences from each training dataset, and merge and shuffle them for one-iteration training. We use merged MSR/PPD/CTB-dev as the MWS dev data for model selection.[9]

For the SWS aggregation model, three SWS models are separately trained on the three training/dev datasets. For evaluation, three SWS outputs produced independently are merged as one MWS result given a sentence.

In all experiments, training stops when F-score on the dev data does not improve in 20 consecutive iterations, and we choose the model that performs best on the dev data for final evaluation.

[8] https://github.com/jhcross/span-parser. We are very grateful for their helping us solve some code issues at the early stage of this work.

[9] For MSR-dev, only the first 3,000 sentences are used during training due to efficiency concern.

Main results: Table 6 reports the performance of different approaches on both the pseudo MWS dev data and the manually annotated MWS test data. The "#Word" column reports the total number of words returned by the corresponding model; the following three columns show the percentages of words of different granularities; the last "Overlapping" column gives the percent of words that overlap with other words, which only happens in the "SWS aggregation" approach, since no constraint can be applied to the three separate SWS models during testing. From the results, we can draw the following findings.

First, the results suggest that using pseudo training and dev datasets to build a MWS model is feasible, based on two evidences: 1) our simple benchmark model can reach a high F-score of 96.07% on the manually annotated test data, which is 1.77% higher than directly aggregating outputs of three SWS models; 2) the P/R/F scores on the pseudo dev data and on the manually labeled test data are quite consistent in general, indicating that it is reliable to use the pseudo dev data for model selection and tuning.

Second, the parsing approach and the sequence labeling approach (with or without bichar embeddings) achieve very similar performance (within 0.15% vibration), More importantly, the parsing approach produces more words and more multi-grained words than the sequence labeling approach, indicating that it is potentially more proper to model MWS as a parsing problem in order to better capture and represent multi-granularity structures. Another possible disadvantage of the sequence labeling approach is that the trained model cannot produce more granularity levels (e.g., four-grained) beyond

those in the training data. Nevertheless, compared against the manual annotations in Table 3, both the parsing and sequence labeling approaches retrieve much less multi-grained words, which is caused by the under-representation issue of the pseudo training data, as discussed in Section 3.

Third, the SWS aggregation approach achieves the best recall at the price of very low precision on both dev/test data. We believe the reason is that training three SWS models separately on one of the three training datasets has two disadvantages: 1) connections among different guidelines are totally ignored, leading to many overlapping words (1.0%); 2) smaller training data also degrades the performance of each SWS model.

Finally, using bichar embeddings turns out very helpful for MWS, and leads to 0.97 $\sim$ 1.18% F-score improvement on dev data and 0.62 $\sim$ 0.85% on test data, which is consistent with the SWS results in Pei et al. (2014).

6 Related Work

As far as we know, this is the first work that formally proposes and addresses the problem of Chinese MWS under the data-driven machine learning framework. It is true that the industrial community, driven by practical demand, has long been interested in retrieving words of different granularities from the engineering perspective, based on lexicon dictionaries and heuristic rules (Zhu and Li, 2008; Hou et al., 2010). We also discover two publicly released toolkits, i.e., IKAnalyzer[10] and PoolWord[11], which consider all substrings in a sentence and return those above a threshold probability as candidate words. In contrast, this paper defines MWS as a strict hierarchy structure, and propose a supervised learning framework for the problem.

To alleviate the high OOV-ratio issue of character-based sequence labeling, Zhang et al. (2006) and Zhao and Kit (2007) propose subword-based sequence labeling for word segmentation by extracting high-frequency subword and treating them as the basic labeling units. Li (2011) and Li and Zhou (2012) propose to jointly parse the internal structures of words and syntactic structure of a sentence. Their definition of internal structures mainly considers prefix or suffix information. They manually annotate the internal structures of words that have high-frequency prefixes or suffixes and left other words with flat structures in CTB. Zhang et al. (2013) further annotate internal structures of all words in CTB and then perform character-level parsing with WS labels. Cheng et al. (2015) propose to cope with the multiple WS standard problem based on internal word structures. After close study of the above works, we find that the MWS annotations automatically built in this work actually capture a lot of subwords and word internal structures in previous works. Most importantly, the main focus of previous works is to improve SWS or parsing performance, whereas this work aims to build a hierarchy structure of multi-grained words. We leave the integration of MWS and parsing for future work.

It has been a long debate whether there exists an optimal WS granularity for MT, which is further complicated by the inevitable mistakes contained in 1-best WS outputs. Dyer et al. (2008) propose an MT model based on source-language word lattices, obtained by merging the outputs of different segmenters. Xiao et al. (2010) propose joint SWS and MT based on word lattices. Recently, Su et al. (2017) propose a word lattice-based neural MT model. They train many segmenters on MSR/PPD/CTB, and merge the outputs to produce word lattices for source-language sentences, which is similar to our SWS aggregation approach. All above works show the usefulness of word lattices instead of a single SWS output. In help IR, Liu et al. (2008) propose a ranking based WS approach for producing words of different granularities. We believe this work can further help both IR and MT by supplying with more accurate MWS results.

7 Conclusion

This work proposes and addresses the problem of MWS, so that all words of different granularities can be captured in a hierarchy structure given a sentence. We can draw the

[10] https://github.com/medcl/elasticsearch-analysis-ik
[11] http://pullword.com/

following interesting findings.

(1) Our annotation conversion approach can gather high-quality pseudo MWS training/dev datasets, and hence it is feasible to use them for model training and tuning.

(2) Manual MWS data annotation tells us that about 28.4% words are multi-grained, and among them 94.4% are two-grained words.

(3) The parsing and sequence labeling approaches achieve very similar performance, and outperform the SWS aggregation approach by a large margin.

We believe there are many exploration directions for this new task, among which we are particularly interested in three in the near future: 1) improving our benchmark approaches by considering task-specific features and neural network architectures, 2) verifying the usefulness of MWS to high-level applications such as MT, 3) integrating MWS with syntactic parsing in some way by exploiting existing treebanks.

Acknowledgments

The authors would like to thank the anonymous reviewers for the helpful comments. We are greatly grateful to all students in LA group for their hard work as annotators. Especially, we thank Jiawei Sun for her extensive participant and support through this work. We also thank Wenliang Chen and Guodong Zhou for the helpful discussions. This work was supported by National Natural Science Foundation of China (Grant No. 61525205, 61373095, 61502325).

References

Deng Cai and Hai Zhao. 2016. Neural word segmentation learning for chinese. In *Proceedings of ACL*, pages 409–420.

Xinchi Chen, Xipeng Qiu, Chenxi Zhu, Pengfei Liu, and Xuanjing Huang. 2015. Long short-term memory neural networks for chinese word segmentation. In *Proceedings of EMNLP*, pages 1197–1206.

Fei Cheng, Kevin Duh, and Yuji Matsumoto. 2015. Synthetic word parsing improves chinese word segmentation. In *ACL*, pages 262–267.

James Cross and Liang Huang. 2016. Span-based constituency parsing with a structure-label system and provably optimal dynamic oracles. In *Proceedings of EMNLP*, pages 1–11.

Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008. Generalizing word lattice translation. In *Proceedings of ACL*, pages 1012–1020.

Thomas Emerson. 2005. The second international chinese word segmentation bakeoff. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing*, pages 123–133.

Lei Hou, Min Chu, Jingming Tang, Jian Sun, Xiaoling Liao, Rengang Peng, Yang Yang, and Bingjing Xu. 2010. Method and device for providing multi-granularity word segmentation result. Chinese Patent (CN102479191A).

Chang-Ning Huang, Yumei Li, and Xiaodan Zhu. 2006. Tokenization Guidelines of Chinese Text (V5.0, in Chinese). Microsoft Research Asia.

Björn H. Jernudd and Michael J. Shapiro, editors. 1989. *The Politics of Language Purism (page 214)*. Mouton de Gruyter.

Zhenghua Li, Jiayuan Chao, Min Zhang, and Wenliang Chen. 2015. Coupled sequence labeling on heterogeneous annotations: POS tagging as a case study. In *Proceedings of ACL*, pages 1783–1792.

Zhenghua Li, Jiayuan Chao, Min Zhang, and Jiwen Yang. 2016. Fast coupled sequence labeling on heterogeneous annotations via context-aware pruning. In *Proceedings of EMNLP*, pages 753–762.

Zhongguo Li. 2011. Parsing the internal structure of words: A new paradigm for chinese word segmentation. In *Proceedings of ACL*, pages 1405–1414.

Zhongguo Li and Guodong Zhou. 2012. Unified dependency parsing of chinese morphological and syntactic structures. In *Proceedings of EMNLP*, pages 1445–1454.

Yijia Liu, Wanxiang Che, Jiang Guo, Bing Qin, and Ting Liu. 2016. Exploring segment representations for neural segmentation models. In *Proceedings of AAAI*, pages 2880–2886.

Yixuan Liu, Bin Wang, Fan Ding, and Sheng Xu. 2008. Information retrieval oriented word segmentation based on character association strength ranking. In *Proceedings of EMNLP*, pages 1061–1069.

Yuan Liu and Nanyuan Liang. 1986. Fundation of Chinese processing: statistics of modern Chinese word frequencies. *Journal of Chinese Information Processing (in Chinese)*, 0(1):17–25.

Jianqiang Ma and Erhard Hinrichs. 2015. Accurate linear-time chinese word segmentation via embedding matching. In *Proceedings of ACL-IJCNLP*, pages 1733–1743.

Wenzhe Pei, Tao Ge, and Baobao Chang. 2014. Max-margin tensor neural network for chinese word segmentation. In *Proceedings of ACL*, pages 293–303.

Richard Sproat, William Gales, Chilin Shih, and Nancy Chang. 1996. A Stochastic Finite-State Word-Segmentation Algorithm for Chinese. *Computational Linguistics*, 22(3).

Jinsong Su, Zhixing Tan, Deyi Xiong, Rongrong Ji, Xiaodong Shi, and Yang Liu. 2017. Lattic-based recurrent neurla network encoders for neural machine translation. In *Proceedings of AAAI*.

Xinyan Xiao, Yang Liu, YoungSook Hwang, Qun Liu, and Shouxun Lin. 2010. Joint tokenization and translation. In *Proceedings of COLING*, pages 1200–1208.

Chi Xiu. 2013. *The research and implementation of method for domain Chinese word segmentation*. Ph.D. thesis, Beijing University of Technology.

Nianwen Xue. 2003. Chinese word segmentation as character tagging. *Computational Linguistics and Chinese Language Processing*, 8(1):29–48.

Nianwen Xue, Fei Xia, Fu-Dong Chiou, and Martha Palmer. 2005. The Penn Chinese Treebank: Phrase structure annotation of a large corpus. *Natural Language Engineering*, 11(2):207–238.

Shiwen Yu, Huiming Duan, Xuefeng Zhu, Bin Swen, and Baobao Chang. 2003. Specification for corpus processing at Peking University: Word segmentation, POS tagging and phonetic notation (In Chinese). *Journal of Chinese Language and Computing*, 13(2):121–158.

Hua-Ping Zhang and Qun Liu. 2002. Model of Chinese words rough segmentation based on n-shortest-paths method. *Journal of Chinese Information Processing (in Chinese)*, 16(5):1–7.

Meishan Zhang, Yue Zhang, Wanxiang Che, and Ting Liu. 2013. Chinese parsing exploiting characters. In *Proceedings of ACL*, pages 125–134.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2016. Transition-based neural word segmentation. In *Proceedings of ACL*, pages 421–431.

Ruiqiang Zhang, Genichiro Kikui, and Eiichiro Sumita. 2006. Subword-based tagging for confidence-dependent chinese word segmentation. In *Proceedings of COLING/ACL*, pages 961–968.

Yue Zhang and Stephen Clark. 2007. Chinese segmentation with a word-based perceptron algorithm. In *Proceedings of ACL*, pages 840–847.

Hai Zhao and Chunyu Kit. 2007. Effective subsequence-based tagging for Chinese word segmentation. *Journal of Chinese Information Processing (in Chinese)*, 21(5):8–13.

Xiaoqing Zheng, Hanyang Chen, and Tianyu Xu. 2013. Deep learning for Chinese word segmentation and POS tagging. In *Proceedings of EMNLP*, pages 647–657.

Jian Zhu and Shan Li. 2008. Method and device for large- and small-grained segmentation of Chinese text. Chinese Patent (CN101246472A).

Appendix A: Collected Annotation Inconsistencies

In MSR

(1) "一日 (one day)" is annotated as "一日 (one day)" or "一 (one)/日 (day)".

(2) "过多 (too much)" is annotated as "过多 (too much)" or "过 (too)/多 (much)".

(3) "这个 (this one)" and "这项 (this item)" have the same stucture of "这 (this)" + #, however they are annotated as "这个 (this one)" and "这 (this)/项 (item)" respectively.

(4) "无异于 (the same to)" and "归功于 (owe to)" have the same stucture of # + "于 (to)", however they are annotated as "无异 (the same)/于 (to)" and "归功于 (owe to)" respectively.

(5) "核武器 (nuclear weapon)" and "核技术 (nuclear technology)" have the same stucture of "核 (nuclear)" + #, however they are annotated as "核武器 (nuclear weapon)" and "核 (nuclear)/技术 (technology)" respectively.

(6) "下一步 (the next step)" and "下一场 (the next game)" have the same stucture of "下一 (the next)" + #, however they are annotated as "下一步 (the next step)" and "下 (next)/一 (one)/场 (game)" respectively.

(7) "副主任 (deputy director)" and "副总统 (vice-president)" have the same stucture of "副 (vice)" + #, however they are annotated as "副 (deputy)/主任 (director)" and "副总统 (vice-president)" respectively.

(8) "工作者 (worker)" and "创始者 (creator)" have the same stucture of # + "者 (-er/or)", however they are annotated as "工作者 (worker)" and "创始 (create)/者 (-or)" respectively.

(9) "跨世纪 (cross century)" and "跨国界 (cross border)" have the same stucture of "跨" + # (cross + #), however they are annotated as "跨世纪 (cross century)" and "跨 (cross)/国界 (border)"

In PPD

(1) "部长级 (ministerial level)" is annotated as "部长级 (ministerial level)" or "部长 (ministerial)/级 (level)".

(2) "一日 (one day)" is annotated as "一日 (one day)" or "一 (one)/日 (day)".

(3) "过多 (too much)" is annotated as "过多 (too much)" or "过 (too)/多 (much)".

(4) "最大 (biggest)" is annotated as "最大 (biggest)" or "最 (most)/大 (big)".

(5) "还有 (and also)" is annotated as "还有 (and also)" or "还 (also)/有 (have)".

(6) "重奖 (reward greatly)" is annotated as "重奖 (reward)" or "重 (reward)/奖 (greatly)".

(7) "借助于 (by means of)" and "归功于 (owe to)" have the same stucture of # + "于 (to)", however they are annotated as "借助于 (by means of)" and "归功 (owe)/于 (to)" respectively.

(8) "南斯拉夫联盟 (Yugoslavia Union)" and "南联盟 (Yugoslavia Union)" have the same stucture of # + "联盟 (Union)", however they are annotated as "南斯拉夫/联盟 (Yugoslavia Union)" and "南联盟 (Yugoslavia Union)" respectively.

category	Chinese example	CTB	PPD	MSR
时间词 (temporal word)	上午十一时 (11 a.m.)	上午/十一时	上午/十一时	上午十一时
	今年下半年 (the second half of this year)	今年/下半年	今年/下半年	今年下半年
	80 年代中期 (the mid-1980s)	80 年代/中期	80/年代/中期	80 年代中期
	2000 年 1 月 1 日 (January 1, 2000)	2000 年/1 月/1 日	2000 年/1 月/1 日	2000 年 1 月 1 日
数量词 (quantifier)	一个 (one)	一/个	一个	一个
	33 亿元 (3.3 billion yuan)	33 亿/元	33 亿/元	33 亿元
	八十二年 (eighty-two years)	八十二/年	八十二/年	八十二年
	十多个 (more than ten)	十多/个	十/多/个	十多个
团体、机构、组织 (orgnization)	欧洲联盟 (European Union)	欧洲/联盟	欧洲/联盟	欧洲联盟
	乒乓球队 (table tennis team)	乒乓球队	乒乓球队	乒乓球/队
	中共中央 (the Central Committee of the Communist Party of China)	中共/中央	中共中央	中共中央
	人事部门 (personnel department)	人事/部门	人事部门	人事/部门
地名 (placename)	森林公园 (forest park)	森林/公园	森林/公园	森林公园
	塞尔维亚共和国 (The Republic of Serbia)	塞尔维亚/共和国	塞尔维亚/共和国	塞尔维亚共和国
	中华人民共和国 (The People's Republic of China)	中华/人民/共和国	中华人民共和国	中华人民共和国
代词 + 名词 (pronoun + noun)	各国 (each country)	各/国	各国	各国
	每人 (everyone)	每/人	每人	每人
	各单位 (each unit)	各/单位	各/单位	各单位
专名 + 名词 (proper noun + noun)	东方人 (oriental)	东方人	东方/人	东方/人
	诺贝尔奖 (Nobel Prize)	诺贝尔奖 or 诺贝尔/奖	诺贝尔奖	诺贝尔/奖
令人 + # (make sb. + #)	令人满意 (satisfactory)	令人满意 or 令/人/满意	令人满意	令人/满意
	令人感动 (touching)	令/人/感动	令人感动	令人/感动
	令人瞩目 (eye-catching)	令/人/瞩目	令人瞩目	令人/瞩目
# + 于 (# + to/for)	有利于 (beneficial to)	有利/于 or 有利于	有利/于 or 有利于	有利于
	用于 (use for)	用于 or 用/于	用于	用于
	囿于 (confined to)	囿于	囿于 or 囿/于	囿/于
# + 率 (# + rate)	使用率 (utilization rate)	使用率	使用率	使用/率
	通胀率 (inflation rate)	通胀率	通胀率	通/胀/率
	通货膨胀率 (inflation rate)	通货膨胀率	通货膨胀率	通货膨胀/率
	市场占有率 (market share)	市场/占有率	市场占有率	市场占有/率
# + 出 (# + out)	看出 (find out)	看出	看/出	看/出
	走出 (go out)	走出	走/出	走出
	拨出 (dial out)	拨出	拨/出	拨/出
跨 + # (cross + #)	跨世纪 (cross-century)	跨世纪 or 跨/世纪	跨/世纪	跨世纪
	跨年度 (go beyond the year)	跨/年度	跨年度	跨年度
	跨国界 (cross border)	跨国界	跨/国界	跨/国界
# + 污染 (# + pollution)	水污染 (water pollution)	水污染 or 水/污染	水污染	水污染 or 水/污染
	环境污染 (environmental pollution)	环境/污染	环境/污染	环境污染
# + 工业 (# + industry)	轻工业 (light industry)	轻工业 or 轻/工业	轻工业	轻工业
	重工业 (heavy industry)	重工业 or 重/工业	重工业	重工业
	化学工业 (chemical industry)	化学/工业	化学工业	化学工业 or 化学/工业
全 + # (whole + #)	全市 (whole city)	全/市	全市	全/市 or 全市
	全天 (whole day)	全/天 or 全天	全天	全/天
	全省 (whole province)	全/省	全省	全省
# + 法 (# + law)	组织法 (constitutive law)	组织/法	组织/法	组织法
	刑事诉讼法 (criminal procedure law)	刑事/诉讼法	刑事诉讼法	刑事/诉讼法 or 刑事诉讼/法
	土地管理法 (land administration law)	土地/管理法	土地管理法	土地/管理/法
# + 后接成分 (# + subsequent component)	演唱者 (singer)	演唱者	演唱者	演唱/者
	金融家 (financier)	金融家	金融家	金融/家
	投资商 (investor)	投资商	投资商	投资/商
	丰富性 (richness)	丰富性	丰富性	丰富/性
	商业化 (commercialization)	商业化	商业化	商业/化
	知识型 (knowledge-based)	知识型	知识型	知识/型

Table 7: An incomplete collection of annotation heterogeneity.

(9) “中共中央 (The CPC Central Committee)” and “越共中央 (Vietnamese Communist Party)” have the same stucture of # + “共中央 (the central government)”, however they are annotated as “中共中央 (The CPC Central Committee)” and “越共 (Vietnamese Communist Party)/中央 (central)” respectively.

(10) “下一步 (the next step)” and “下一场 (the next game)” have the same stucture of “下一 (the next)” + #, however they are annotated as “下一步 (the next step)” and “下 (next)/一 (one)/场 (game)” respectively.

(11) “跨年度 (go beyond the year)” and “跨国界 (cross border)” have the same stucture of “跨” + # (cross + #), however they are annotated as “跨 (go beyond)/年度 (year)” and “跨国界 (cross border)” respectively.

In CTB

(1) “重量级 (heavyweight)” is annotated as “重量级 (heavyweight)” or “重量 (heavy)/级 (weight)”.

(2) “一日 (one day)” is annotated as “一日 (one day)” or “一 (one)/日 (day)”.

(3) “再就业 (re-employment)” is annotated as “再就业 (re-employment)” or “再 (once again)/就业 (employment)”.

(4) “野牛 (wild cow)” is annotated as “野牛 (wild cow)” or “野 (wild)/牛 (cow)”.

(5) “最大 (biggest)” is annotated as “最大 (biggest)” or “最 (most)/大 (big)”.

(6) “还有 (and also)” is annotated as “还有 (and also)” or “还 (also)/有 (have)”.

(7) “下一步 (the next step)” is annotated as “下一步 (the next step)” or “下 (next)/一 (one)/步 (step)”.

(8) “副总统 (vice-president)” is annotated as “副总统 (vice-president)” or “副 (vice)/总统 (president)”.

(9) “变得 (change into)” is annotated as “变得 (change into)” or “变 (change) 得 (into)”.

(10) “有利于 (beneficial to)” and “归功于 (owe to)” have the same stucture of # + “于 (to)”, however they are annotated as “有利于 (beneficial to)” and “归功 (owe)/于 (to)” respectively.

(11) “跨年度 (go beyond the year)” and “跨国界 (cross border)” have the same stucture of “跨” + # (cross + #), however they are annotated as “跨 (go beyond)/年度 (year)” and “跨国界 (cross border)” respectively.

Appendix B: See Table 7