# Multi-grained Chinese Word Segmentation with Weakly Labeled Data

**Chen Gong, Zhenghua Li*, Bowei Zou, Min Zhang**
Institute of Artificial Intelligence, School of Computer Science and Technology,
Soochow University, China
cgong@stu.suda.edu.cn, {zhli13, minzhang}@suda.edu.cn,
zou_bowei@i2r.a-star.edu.sg

## Abstract

In contrast with the traditional single-grained word segmentation (SWS), where a sentence corresponds to a single word sequence, multi-grained Chinese word segmentation (MWS) aims to segment a sentence into multiple word sequences to preserve all words of different granularities. Due to the lack of manually annotated MWS data, previous work train and tune MWS models only on automatically generated pseudo MWS data. In this work, we further take advantage of the rich word boundary information in existing SWS data and naturally annotated data from dictionary example (DictEx) sentences, to advance the state-of-the-art MWS model based on the idea of weak supervision. Particularly, we propose to accommodate two types of weakly labeled data for MWS, i.e., SWS data and DictEx data by employing a simple yet competitive graph-based parser with local loss. Besides, we manually annotate a high-quality MWS dataset according to our newly compiled annotation guideline, consisting of over 9,000 sentences from two types of texts, i.e., canonical newswire (NEWS) and non-canonical web (BAIKE) data for better evaluation. Detailed evaluation shows that our proposed model with weakly labeled data significantly outperforms the state-of-the-art MWS model by 1.12 and 5.97 on NEWS and BAIKE data in F1.

## 1 Introduction

As a preliminary but critical processing step for Chinese language processing, word segmentation (WS) has been extensively studied for decades and made great progress (Zheng et al., 2013; Pei et al., 2014; Zhang et al., 2016; Yang et al., 2019; He et al., 2020). However, most of previous works adopt the single-grained word segmentation (SWS) formulation, where a sentence corresponds to a single word sequence according to some pre-defined annotation guidelines. As shown in Figure 1 (left), the SWS annotations of the sentence are different according to the guidelines of Penn Chinese Treebank (CTB) (Xue et al., 2005), the People Daily Corpus of the Peking University (PPD) (Yu et al., 2003), and the Microsoft Research WS Corpus (MSR) (Huang et al., 2006). This is largely due to the fact that the boundary between words is usually subtle and vague (Jernudd and Shapiro, 1989) and there are various underlying linguistic theories. Sproat et al. (1987) show that the consensus ratio over word boundaries is only 76% even among Chinese native speakers without training on any guideline.

In order to guarantee the annotation consistency, people turn to detailed annotation guidelines according to specific tasks or applications. For example, CTB usually prefers fine-grained words over coarse-grained words to facilitate further annotation of syntax and semantics, while PPD accommodates more coarse-grained words with information extraction and retrieval tasks in mind. This poses a strong challenge in Chinese WS since words of different granularities are necessary for a variety of tasks and applications at the same time. Zhu and Li (2008) and Hou et al. (2010) adopt heuristic rules based on lexicon dictionaries to accommodate the necessity. Similar functions are provided by publicly available Chinese WS tools such as jieba and PullWord. However, the effectiveness of such tools are far below satisfactory without tackling segmentation ambiguity problem.

*Corresponding author.




*Proceedings of the 28th International Conference on Computational Linguistics*, pages 2026–2036
Barcelona, Spain (Online), December 8-13, 2020

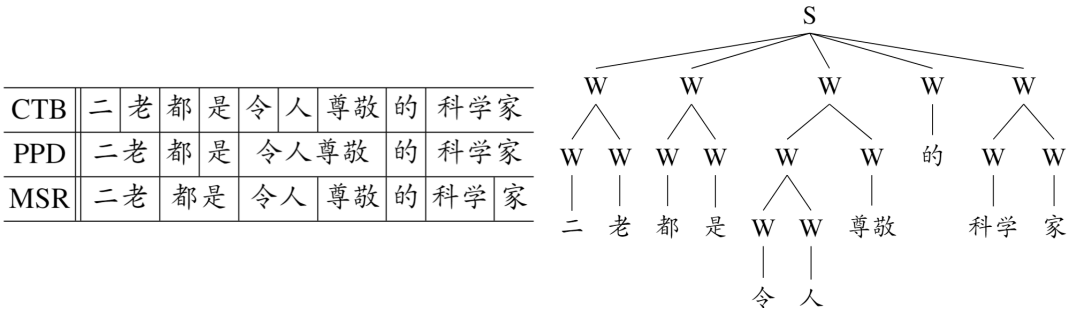


Figure 1: SWS annotation heterogeneity of an example sentence (left) with its MWS tree (right): "二 (two) 老 (elder) 都 (both) 是 (are) 令 (make) 人 (people) 尊敬 (respect) 的 (of) 科学 (science) 家 (expert)."

It is worth emphasizing that even for the same task or application, words of different granularities can be useful due to its potential complementarity: fine-grained words capture local features and help reduce data sparseness, whereas coarse-grained words reserve more semantics to perform exacter matching and analysis. This facilitates researchers to employ multiple SWS outputs at the same time in information retrieval (IR) (Liu et al., 2008) and machine translation (MT) (Su et al., 2017).

Motivated by above perspectives, multi-grained word segmentation (MWS) is formally proposed by Gong et al. (2017) as a useful and challenging direction for research on word segmentation. Given an input sentence, MWS aims to accommodate all words of different granularities with a hierarchical tree structure. Figure 1 (right) presents an example, where "W" means the spanning characters compose a word. In this example, "二 (two)", "老 (elder)", "二老 (two elders)", "都 (both)", "是 (are)", "都是 (both are)", "令 (make)", "人 (people)", "令人 (make people)", "尊敬 (respect)", "令人尊敬 (respectable)", "的 (of)", "科学 (science)", "家 (expert)", "科学家 (scientist)" are all the words of different granularities. To solve the issue of lacking labeled MWS data, they construct a large-scale pseudo MWS data for model training and tuning. They propose several MWS approaches and justify the superiority of treating MWS as constituent parsing. However, their approaches only learn from pseudo MWS data and do not fully exploit word boundary information from other available sources which are helpful and easy to obtain.

This paper advances the state-of-the-art MWS model with weakly labeled data. Particularly, we propose to accommodate two types of weakly labeled data, i.e., SWS and naturally annotated dictionary example (DictEx) sentences, as extra training data, by employing a simple but competitive graph-based parsing model with local span-wise loss. Besides, we develop a unified annotation guideline for MWS and manually annotate a large-scale high-quality MWS dataset containing over 9,000 sentences from both canonical newswire texts (NEWS) and non-canonical web texts (BAIKE) for better evaluation. Detailed evaluation shows that our proposed model with weakly labeled data significantly improves the state-of-the-art MWS model by 1.12 on NEWS and by 5.97 on BAIKE in F1. We release all the newly annotated data and the codes at `https://github.com/gloria0108/multi-grained-word-seg`.

## 2 Graph-based Model with Local Loss

Given an input sentence, the task of MWS is to retrieve all words of different granularities, which can be naturally organized as a hierarchical tree structure as shown in Figure 1 (right).

Gong et al. (2017) propose several MWS approaches and show that treating MWS as constituent parsing leads to the best performance. They adopt the transition-based parser of Cross and Huang (2016), which greedily searches an optimal shift-reduce action sequence to build a tree. In this work, instead of adopting the transition-based parser as Gong et al. (2017), we employ the graph-based parser of Stern et al. (2017) and replace the original global max-margin loss with local span-wise loss (Joshi et al., 2018; Teng and Zhang, 2018) as our basic MWS model due to two considerations: 1) the graph-based parser with local loss gains more efficiency without hurting the performance compared with the transition-based parser and the graph-based parser with global loss, which will be discussed in Section 5.3; 2)

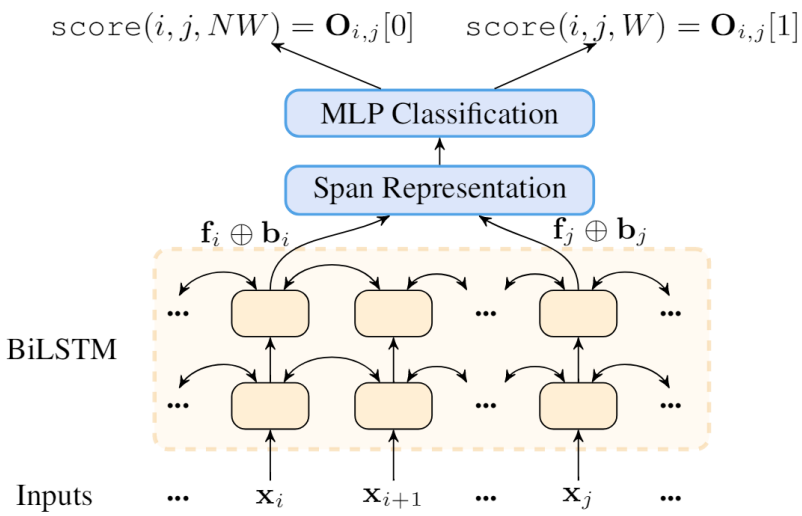


Figure 2: Architecture of our MWS model.

more importantly, this work aims to conduct in-depth study on a simple, efficient, and effective way to incorporate weakly labeled data for MWS. The graph-based parser with local loss trains the model directly on individual labeled spans, and thus can accommodate weakly labeled data naturally.

### 2.1 Model Architecture

As illustrated in Figure 2, our model architecture consists of following four components.

**The input layer** builds a dense vector representation for each character position $\mathbf{x}_0\mathbf{x}_1...\mathbf{x}_n$, given the input sentence $c_0c_1...c_n$, where $c_i$ denotes the $i$-th character and $c_0$ is a pseudo character for sentence start. Following previous work on Chinese word segmentation (Pei et al., 2014), we use the concatenation of single character embeddings $\mathbf{emb}^{c_i}$ and bigram character embeddings $\mathbf{emb}^{c_{i-1}c_i}$ as the input.

$$\mathbf{x}_i = \mathbf{emb}^{c_i} \oplus \mathbf{emb}^{c_{i-1}c_i} \tag{1}$$

**The encoding layer** uses two layers of BiLSTM to encode the sentence and produce contextualized representations. We use $\mathbf{f}_i$ and $\mathbf{b}_i$ to denote the hidden vector of the top-layer forward and backward LSTMs for the $i$-th position.

**The span representation layer** constructs a dense representation vector for each possible span $c_i...c_{j-1}$ denoted as $(i, j)$:

$$\mathbf{r}_{i,j} = (\mathbf{f}_j - \mathbf{f}_i) \oplus (\mathbf{b}_i - \mathbf{b}_j) \tag{2}$$

which is also known as the LSTM-minus features (Wang and Chang, 2016; Cross and Huang, 2016).

**The classification layer** uses an MLP to compute the labeling scores of each span.

$$\mathbf{o}_{i,j} = \mathbf{W}_2\texttt{ReLU}(\mathbf{W}_1\mathbf{r}_{i,j} + \mathbf{b}_1) + \mathbf{b}_2 \tag{3}$$

where $\mathbf{W}_1$, $\mathbf{W}_2$, $\mathbf{b}_1$, and $\mathbf{b}_2$ are parameters. In our task, the dimension of $\mathbf{o}_{i,j}$ is 2, w.r.t "W" and "NW" respectively ($\mathbf{o}_{i,j}[0]$ is the score of labeling span $(i, j)$ as a word, and $\mathbf{o}_{i,j}[1]$ as a non-word).

### 2.2 Training with Local Span-wise Loss

During training, we compute a local cross-entropy loss value for each span, and accumulate all loss values of all possible spans in the input sentence.

$$\mathcal{L} = -\sum_{0 \le i < j \le n} \log \frac{e^{\mathbf{o}_{i,j}[y^*_{i,j}]}}{e^{\mathbf{o}_{i,j}[0]} + e^{\mathbf{o}_{i,j}[1]}} \tag{4}$$

where $y^*_{i,j} \in \{0, 1\}$ is the gold-standard label for span $(i, j)$.

For the weakly labeled data introduced in Section 3, the loss function only accumulates the values of the spans with ground-truth labels and overlooks others.

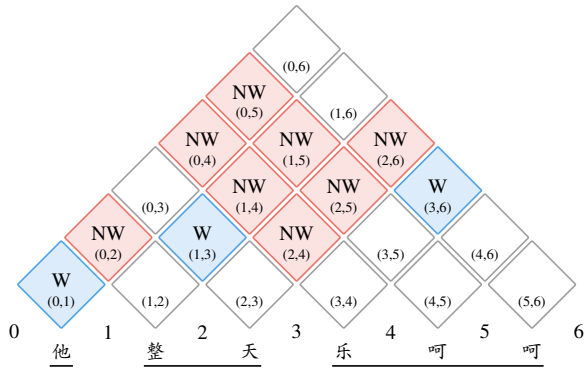


Figure 3: A weakly labeled sentence from SWS data: 他 (He) 整天 (whole day) 乐呵呵 (happy).

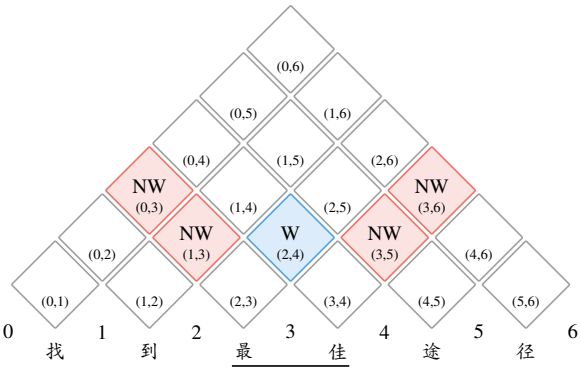


Figure 4: A weakly labeled sentence from DictEx: 找到 (find) 最佳 (the best) 途径 (way).

### 2.3 Inference with Chart Decoding

During test, after obtaining the scores of different labels for each span, we follow Stern et al. (2017) and adopt chart decoding to find a global optimal MWS tree $T^*$ with the highest score from all the possible trees. The score of an MWS tree $T$ is the sum of label scores of all spans.

$$\texttt{score}(T) = \sum_{(i,j,y) \in T} \mathbf{o}_{i,j}[y] \tag{5}$$

where $(i, j, y)$ means span $(i, j)$ is labeled as $y, y \in \{0, 1\}$.

## 3 MWS with Weakly Labeled Data

Due to the lack of manually labeled high-quality data, Gong et al. (2017) construct a large-scale pseudo labeled MWS data by automatically converting existing heterogeneous SWS data in a pairwise way. They train their model with the pseudo labeled data and obtain promising performance on a small scale manually labeled MWS data. However, they observe that the pseudo labeled data inevitably has a lot of noise due to the pairwise conversion model and severly suffer from the under-representation phenomena of multi-grained words. In order to alleviate above issues, we propose to accommodate two types of weakly labeled data (i.e., SWS data and DictEx data) as extra data for MWS training, inspired by previous work on utilizing naturally annotated data for SWS (Jiang et al., 2013; Zhao et al., 2018).

### 3.1 SWS Data as Weakly Labeled Data

Although Gong et al. (2017) already adopt three heterogeneous SWS data to construct pseudo MWS data, there exist many other high-quality SWS datasets to explore with, and new ones are constantly annotated by research institutes and commercial companies.

Instead of converting such new SWS data into noisy MWS data as Gong et al. (2017) did, we propose to treat SWS data as weakly labeled MWS data with a minor extension on the training loss. The key idea is that when accumulating training loss, we only consider spans whose gold-standard labels can be directly determined according to the SWS annotation and overlook others, as illustrated in Figure 3, where spans labeled with "W" correspond to the words in the SWS sentence, those labeled with "NW" are definitely non-words since they overlap with existing gold-standard words (i.e., the words annotated in SWS data), and all blank spans without labels are overlooked with no loss.

### 3.2 DictEx Data as Weakly Labeled Data

The use of naturally annotated data has been extensively studied for SWS (Jiang et al., 2013; Liu et al., 2014; Zhao et al., 2018). The basic idea is to derive word boundaries from implicit information encoded in web texts, such as anchor texts and punctuation marks, and use them as partially labeled training data in sequence labeling models.

In this work, we propose to obtain naturally annotated data with complete word information from the example sentences in dictionary (rather than only boundaries), which are manually constructed by linguistic experts, e.g.,

Entry: 最佳 (the best)
Example sentence:
1: 找到 最佳 途径　2: 这是 最佳 选择
(find the best way)　( this is the best choice)

where two DictEx sentences are carefully chosen to explain the usage of the word "最佳 (the best)". Obviously, we can safely assume the two characters "最佳 (the best)" compose an explicit word. In this way, we can obtain many naturally annotated sentences, each labeled with one explicit word from a dictionary entry. Similar to the SWS data case, we can adjust the training loss to utilize such naturally annotated DictEx sentences by only accumulating the loss of the spans whose gold-standard labels can be directly determined, as shown in Figure 4.

## 4 High-quality Evaluation Data Annotation

Gong et al. (2017) construct pseudo MWS data as the training and development datasets and manually annotate 1,500 sentences with the MWS tree structure as the test data. They present pseudo MWS results to annotators for correction without defining a strict MWS annotation guideline. Their inter-annotator consensus ratio is very high (98.1%). The main reason is that the annotators usually only detect very obvious segmentation errors and accept others as correct by default. Our early investigation shows that their data contains at least 5% annotation errors according to our newly compiled guideline.

In order to produce more high-quality data for better evaluation, we adopt a more scientific and robust annotation procedure and methodology. Our annotation work differs from Gong et al. (2017) from following aspects: 1) We compile a systematic and detailed MWS annotation guideline for annotators' reference; 2) We abandon the annotation-via-correction method and present raw texts to annotators without pseudo MWS results or any other word information; 3) We adopt double annotation for all sentences with inconsistency handling by experts, to guarantee high quality.

As a result, we achieve the first large-scale high-quality MWS evaluation data covering two sources, i.e., 3,000 NEWS and 6,320 BAIKE sentences.

### 4.1 Annotation Process

**Annotation guideline.** After a few months' in-depth study, we integrate the three well-known SWS guidelines (i.e., CTB, PPD and MSR) and compile a systematic and detailed MWS annotation guideline. We also gradually improve the guideline according to the feedback from the annotators.

**Quality control.** We employ 24 undergraduate students as annotators and train them for 4-8 hours before formal annotation. We apply strict *double annotation* to guarantee high quality and build a browser-based annotation platform to support the annotation workflow.

**Data selection.** To better understand the capability of the state-of-the-art MWS model in dealing with real-world data, we select data from two sources for annotation, i.e., canonical newswire (NEWS) and non-canonical web texts .

For the canonical newswire texts, we re-annotate the 1,500 sentences of Gong et al. (2017), which are randomly sampled from CTB, PPD and MSR, according to our guideline. We further annotate 1,500 sentences from the year 2000 PPD SWS data, leading to 3,000 sentences with MWS full annotations, which are used as the development and test data in this work.

For non-canonical web texts, we collect 12 million sentences with anchor texts from the Baidu Baike website[1] (similar to Wikipedia) after data cleaning. The anchor text is the visible, clickable text in a hyperlink, which usually indicates a word or a phrase and thus can provide word boundary information. For example, a hyperlink to the web page about "二氧化硫 (sulfur dioxide)" might take this form: "排放 (discharge) 大量 (a large amount of) <a href="http://...">二氧化硫 (sulfur dioxide) <\a>", where "二氧化硫 (sulfur dioxide)" is an anchor text. Inspired by the idea of naturally annotated WS data (Jiang et al., 2013), we use anchor texts as word boundaries to select sentences difficult for models. First, we

[1] https://baike.baidu.com/

| | #Sents | #Words | Grain Distribution (%) | | |
|---|---|---|---|---|---|
| | | | Single | Two | Three+ |
| NEWS$^\star$ | 1,500 | 45,279 | 71.6 | 26.8 | 1.6 |
| NEWS | 3,000 | 94,585 | 70.2 | 26.9 | 2.9 |
| BAIKE | 6,320 | 14,445 | 51.5 | 39.2 | 9.3 |

Table 1: Statistics of our manually annotated data. NEWS$^\star$ represents the NEWS data annotated by Gong et al. (2017) Please note that BAIKE data is partially annotated.

| | Type | #Sent | #Word |
|---|---|---|---|
| Train | Pseudo MWS | 138,628 | 4,127,461 |
| | SWS (additional) | 100,000 | 2,490,589 |
| | DictEx (additional) | 145,037 | 146,934 |
| Dev | Manual NEWS | 1,000 | 31,477 |
| Test | Manual NEWS | 2,000 | 63,108 |
| | Manual BAIKE | 6,320 | 14,445 |

Table 2: Data statistics in our experiments.

use our basic MWS model trained on pseudo MWS data to predict an MWS tree for each sentence. Then, we randomly select 6,320 sentences where the automatic MWS tree contains at least one word that violates with a boundary, indicating the model makes at least one mistake. To save cost, we adopt *partial annotation* for the Baike web texts, and annotate only words related with the anchor texts. We use the annotated Baike web texts as a cross-domain test data.

### 4.2 Statistics and Analysis

Table 1 shows the data statistics. Our re-annotated NEWS data contains 1.4% more multi-grained words compared with the NEWS data annotated by Gong et al. (2017), indicating that their annotation procedure may under-annotate multi-granularity structures. It is obvious that BAIKE contains much more multi-grained words than NEWS, because for BAIKE we only partially annotated words related with anchor texts, among which a large proportion are named entities and tend to be multi-grained.

**Inter-annotator consensus ratio.** The word-wise inter-annotator consensus ratio is defined as $\frac{\#\texttt{Word}_{\texttt{annoA}\cap\texttt{annoB}}}{\#\texttt{Word}_{\texttt{annoA}\cup\texttt{annoB}}}$, where the denominator is the number of words after merging the submission of all annotator pairs, and the numerator is the consensus word number. The overall word-wise consensus ratio is only 82.5%. This indicates the difficulty of the annotation task and the necessity of performing strict double annotation to guarantee data quality.

## 5 Experiments

We conduct various experiments to show the effectiveness of our proposed MWS approaches.

**Data.** Data statistics are shown in Table 2. For the training data, we directly adopt the pseudo MWS training data of Gong et al. (2017), consisting of about 140K sentences from CTB, PPD, and MSR via automatic pairwise conversion as the baseline. For the dev data, different from previous work which uses the pseudo MWS data, we randomly sample 1,000 sentences from the manually labeled NEWS data. We use two types of test data, the manually annotated 2,000 sentences of NEWS and 6,320 sentences of BAIKE. For the SWS data, we use 100K sentences from the year 2000 PPD, while we collect 140K naturally annotated DictEx sentences from the Dictionary of Modern Chinese Grammar Information of Peking University (Yu and Zhu, 2017) and the Xinhua dictionary[2].

**Evaluation metrics.** Given an input sentence, MWS aims to precisely segment the sentence into all words of different granularities. We adopt the same evaluation metircs as SWS. Precision ($\frac{\#\texttt{Word}_{\texttt{gold}\cap\texttt{pred}}}{\#\texttt{Word}_{\texttt{pred}}}$), recall ($\frac{\#\texttt{Word}_{\texttt{gold}\cap\texttt{pred}}}{\#\texttt{Word}_{\texttt{gold}}}$) and F1 ($\frac{2PR}{P+R}$) score are used to measure the MWS performance. We adopt Dan Bikel's randomized parsing evaluation comparator for significance test (Noreen, 1989).

**Model setting.** We preserve most of the hyperparameter settings in Stern et al. (2017), and our preliminary experiments show that the performance of our approach is quite stable. Each model is trained for at most 1000 iterations, and early stopping is triggered when the peak performance does not increase in 50 consecutive iterations.

### 5.1 Benchmark Methods

We adopt following four benchmark methods for comparison. Beside the transition-based parser and the graph-based parser with global max-margin loss, we also re-implement another two benchmark methods

[2]http://xh.5156edu.com/

| Models | Dev | | | Test (NEWS) | | | Test (BAIKE) | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | P | R | F1 | P | R | F1 |
| SWS aggregation | 87.75 | **91.65** | 89.66 | 87.78 | **91.45** | 89.58 | 38.21 | **43.30** | 40.59 |
| Sequence labeling | 92.39 | 89.15 | 90.74 | 92.49 | 88.89 | 90.65 | 42.85 | 32.27 | 36.82 |
| Transition-based parser | 94.18 | 91.64 | 92.89 | 94.55 | 90.88 | 92.68 | 48.98 | 39.10 | **43.08** |
| Graph-based parser (global) | **95.57** | 90.88 | **93.16** | **95.34** | 90.51 | **92.86** | **49.64** | 37.89 | 42.98 |
| Our graph-based basic parser (local) | **95.52** | 90.93 | 93.17 | **95.24** | 90.59 | 92.86 | 48.39 | 38.91 | 43.14 |
| +SWS | 94.70 | 93.31 | 94.00 | 94.63 | 93.09 | 93.85 | 50.19 | 47.63 | 48.88 |
| +DictEx | 95.15 | 91.60 | 93.34 | 94.94 | 91.27 | 93.07 | 47.61 | 39.87 | 43.40 |
| +SWS&DictEx | 94.94 | **93.67** | **94.30** | 94.68 | **93.29** | **93.98** | **50.21** | **47.94** | **49.05** |

Table 3: Main results of different approaches.

in Gong et al. (2017) and report their results on the new evaluation data for more insights. All the models adopt the similar architecture based on a multi-layer BiLSTM encoder.

**1. SWS aggregation.** We train three SWS models separately on PPD, MSR and CTB datasets, and merge their outputs as MWS results.

**2. Sequence labeling.** It considers MWS as sequence labeling problem. Each character corresponds to an MWS label to denote the positions of the character in all the multi-grained words containing it.

**3. Transition-based parser.** We adopt the same transition-based parser of Cross and Huang (2016).

**4. Graph-based parser (global).** We use the same graph-based constituent parser of Stern et al. (2017) with global max-margin loss.

### 5.2 Main results

Table 3 compares different approaches on the manually annotated development, NEWS-test and BAIKE-test. Please kindly note that the results look very low on BAIKE-test, because only the most difficult parts (anchor texts) are partially annotated for BAIKE sentences, as discussed in Section 4.1.

**Our graph-based parser with local loss** outperforms the SWS aggregation and sequence labeling methods with large margins ($p < 0.001$) on both dev and tests. We observe that although the SWS aggregation method achieves the best recall compared with other benchmark methods, the precision is very low due to the ignorance of the connections among different heterogeneous SWS data. The recall of the sequence labeling approach is poor, because the sequence labeling model produces less words and multi-grained words than other models and thus fails to produce words more than three granularity levels. These indicate that casting MWS as a constituent parsing problem is more proper.

Our graph-based model with local loss performs slightly better on NEWS-test in F1 and achieves nearly the same performance on BAIKE-test compared with the transition-based parser. Besides, the results of the graph-based parser with local loss and global loss are comparable.

Moreover, the precision values are much higher than the recall values for transition/graph-based parser on both types of test data. This large gap means that the MWS models tend to generate single-grained words rather than multi-grained ones. The main reason is that multi-granularity phenomena are under-represented in the noisy pseudo MWS data due to the mistakes and bias imposed by the automatic conversion models.

**Using additional SWS data** for training brings large F1 improvement over the basic parser trained on only pseudo MWS data. For NEWS, although precision decreases by 0.82/0.61 on dev/test, recall increases by much larger margin of 2.38/2.50, leading to overall F1 increase of 0.83 and 0.99 on dev and test ($p < 0.001$). We believe the reason is that the SWS data can alleviate the under-representation of multi-granularity phenomena in the noisy pseudo MWS training data. For further investigation, we calculate the word proportion (defined as $\frac{\#\texttt{Span}_{\texttt{W}}}{\#\texttt{Span}_{\texttt{W}}+\#\texttt{Span}_{\texttt{NW}}}$, where $\#\texttt{Span}_{\texttt{W}}$ and $\#\texttt{Span}_{\texttt{NW}}$ represent the number of spans labeled "W" and "NW" respectively), which accounts for 3.5% for SWS data and only 2.0% for pseudo MWS data. In other words, there are more positive examples for the "W" label in the

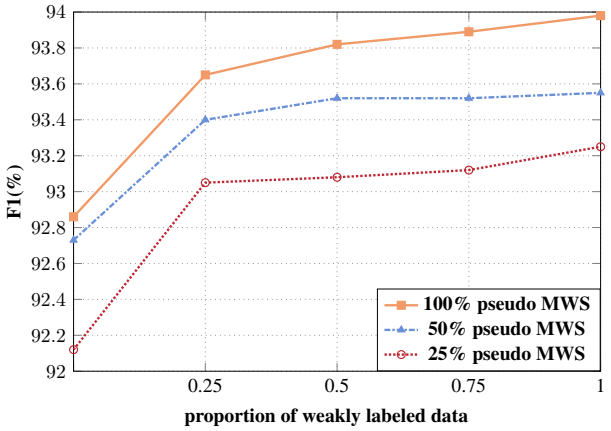


Figure 5: Influence of utilizing different amount of additional weakly labeled data on NEWS-test.

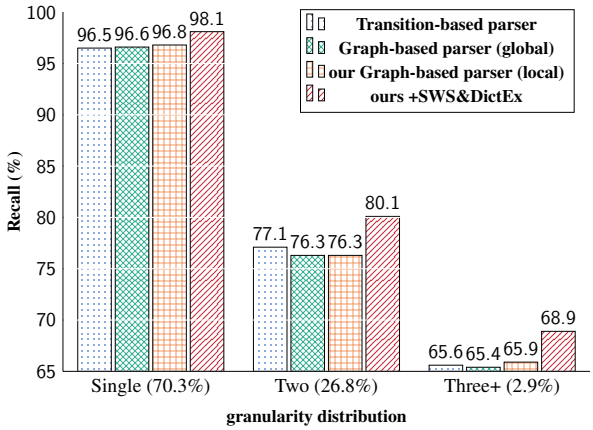


Figure 6: Recall against words of different granularities on NEWS-test.

SWS data than in the pseudo MWS data, thus encouraging the model to produce more multi-grained words and achieve higher recall to some extend. For BAIKE-test, the model with additional SWS data outperforms the basic parser by 5.74 (from 43.14 to 48.88) in F1 ($p < 0.001$). The improvement is mostly contributed by the large increase of recall, which is consistent with NEWS.

**Using the DictEx data** as extra training data consistently improves F1 score by 0.21/0.26 on NEWS/BAIKE test compared with the basic parser trained on only pseudo MWS data. The improvement seems small yet both significant ($p < 0.001$). Looking into the precision/recall changes, though precision decreases a little bit, recall increases a lot. This can be explained similarly by the proportion of positive "W" examples, which is 11.2% for DictEx and it is much higher than pseudo MWS data.

Above results show that the DictEx data is a very reliable knowledge source, and the small yet steady improvement is mainly due to its weak supervision, considering that only a very small portion of spans have loss in each sentence.

**Using both SWS and DictEx data** achieves the highest F1 scores on both dev and test, outperforming the basic parser by 1.13 on dev, 1.12/5.91 on NEWS/BAIKE test ($p < 0.001$). Compared with the "+ SWS" model, the extra DictEx data increases both precision and recall on dev and NEWS/BAIKE test.

Overall, above results show that the SWS data can effectively reach a balance with the pseudo MWS by alleviating under-representation problem of multi-granularity, while the DictEX data can provide complementary and consistent contribution.

### 5.3 Efficiency

We report the averaged one-epoch training time on the pseudo MWS data (consuming 15,000 training instances) as follows:

| | |
|---|---|
| Transition-based parser | 38min |
| Graph-based parser (global) | 35min |
| Our graph-based basic parser (local) | 23min |

We choose the transition-based parser and the graph-based parser with global loss for comparison, as they outperform other benchmark methods by large margins. We find that our graph-based parser with local loss is about 1.5 times faster than these two compared models.

### 5.4 Analysis

We conduct detailed analysis to gain more insights on our proposed approaches. For space limitation, we only present the analysis results on NEWS-test.

**Influence of the amount of data.** Figure 5 illustrates the influence of the amount of both the pseudo MWS data and the weakly labeled data. In each curve, we fix the size of the pseudo MWS data and incrementally add a random subset of the weakly labeled data. For the three curves, we use all, 50%, and 25% of the pseudo MWS data by random sampling. We observe that using more weakly labeled

data leads to consistently higher performance for all three curves. While the first 25% data produces largest improvement, the gain becomes less substantial afterwards. From another aspect, performance steadily goes up by large margin when raising the size of the pseudo MWS data, demonstrating that the pseudo MWS data is fundamentally important for training the MWS model, although containing inevitable noises.

**Performance of different granularities.** To understand the performance of our approaches on words of different granularities, we divide the gold-standard words into three subsets according to their guanularities and compute the recall of each approach on the three subsets. We compare our models with the transition-based parser and the graph-based parser with global loss, which have better performance than other benchmark methods. The results are shown in Figure 6. The percentages in parenthesis at the X-axis denote the word proportions of corresponding granularities on NEWS-test. Figure 6 shows that the words with more granularities have lower recall for all models. This indicates that multi-grained words are difficult to predict. Compared with the transition-based parser and the graph-based parser with global loss, our basic parser using only the pseudo data performs better in two of the three types of granularities. After utilizing weakly labeled data, our MWS model achieves consistent improvements on all the three types of granularities.

## 6 Related Work

**MWS approaches.** The industrial community has long been interested in retrieving words of different granularities with the help of lexicon dictionaries and heuristic rules (Zhu and Li, 2008; Hou et al., 2010). We also find that publicly available WS tools such as jieba[3] and PullWord[4] provide the interface for retrieving words of different granularities. However, all those tools judge the probability of each substring being a word independently, without resolving any segmentation ambiguity. Therefore, the output words may overlap with each other.

Gong et al. (2017) first formally address the MWS task and build a pseudo MWS dataset for model training. They also propose and compare three benchmark MWS approach, i.e., constituent parsing, sequence labeling, and SWS aggregation, showing that treating MWS as constituent parsing is most effective. We follow their work and advance the state-of-the-art MWS research progress from the perspectives of both data and approach.

**Utilizing MWS results.** Due to its critical importance, MWS results have been explored in various NLP applications. Liu et al. (2008) propose a ranking based WS approach to produce words of different granularities to help IR. Su et al. (2017) propose a lattice-based RNN encoder for neural MT by representing MWS outputs in word lattices, leading to improved translation performance. Due to the lack of MWS model, they obtain MWS outputs from several SWS models independently trained on heterogeneous SWS datasets.

**Utilizing weakly labeled data.** The use of weakly labeled data has been an interesting research direction in NLP for a long time. On the one hand, it is usually much easier and cheaper to perform partial annotation than complete annotation, especially for complex tasks such as parsing (Hwa, 1999; Sassano and Kurohashi, 2010; Li et al., 2016b; Joshi et al., 2018). On the other hand, it is sometimes feasible to automatically extract naturally annotated data. Several works utilize naturally annotated data with word boundaries for training SWS models, by making use of markup information such as anchor texts in web pages (Jiang et al., 2013; Liu et al., 2014; Zhao et al., 2018).

In this work, we propose two types of weakly labeled data for MWS, i.e., SWS data and naturally annotated data from DictEx sentences, which are shown to be complementary and able to alleviate the under-representation problem of multi-grained phenomena in the noisy pseudo MWS training data.

**SWS with heterogeneous data.** In recent years, there has been a surge of interest in improving SWS with heterogeneous SWS data. The basic idea is improving SWS by utilizing multiple manually labeled SWS data for training at the same time. Representative works include Li et al. (2016a), Chen et al. (2016), He et al. (2018), Chen et al. (2017) and Yang et al. (2017). Although MWS results can be

[3]https://github.com/fxsjy/jieba

[4]http://pullword.com

obtained by merging multiple SWS outputs, but many overlapped words may generated due to the lack of proper constraints, leading to low precision. In this work, we alleviate this issue by considering MWS as a constituent tree parsing problem.

## 7 Conclusions

This work advances the state-of-the-art MWS research from three perspectives. First, we manually annotate over 9,000 sentences for better evaluation, consisting of both canonical NEWS and non-canonical BAIKE texts. Second, we employ a simple graph-based parsing model with local loss to facilitate the use of weakly labeled data. Finally, we propose to accomodate two types of weakly labeled data as extra training data, i.e., the SWS data and the DictEx data. Detailed analysis show that 1) the simple graph-based parsing model with local loss achieves highly competitive performance; 2) both types of weakly labeled data can provide consistent and substantial gains; 3) our proposed approach outperforms the state-of-the-art MWS model by 1.12 on NEWS and by 5.97 on BAIKE in F1.

## Acknowledgments

We thank our anonymous reviewers for their helpful comments. This work was supported by the National Natural Science Foundation of China (Grant No. 61525205, 61703293), a Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

## References

Hongshen Chen, Yue Zhang, and Qun Liu. 2016. Neural network for heterogeneous annotations. In *Proceedings of EMNLP*, pages 731–741.

Xinchi Chen, Zhan Shi, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-criteria learning for Chinese word segmentation. In *Proceedings of ACL*, pages 1193–1203.

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, Austin, Texas.

Chen Gong, Zhenghua Li, Min Zhang, and Xinzhou Jiang. 2017. Multi-grained Chinese word segmentation. In *Proceedings of EMNLP*, pages 703–714, Copenhagen, Denmark.

Han He, Lei Wu, Hua Yan, Zhimin Gao, Yi Feng, and George Townsend. 2018. Effective neural solution for multi-criteria word segmentation. *arXiv preprint arXiv:1712.02856*.

Xuanli He, Gholamreza Haffari, and Mohammad Norouzi. 2020. Dynamic programming encoding for subword segmentation in neural machine translation. In *Proceedings of ACL*, pages 3042–3051.

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). *Journal of Logic and Computation*.

Rebecca Hwa. 1999. Supervised grammar induction using training data with limited constituent information. In *Proceedings of ACL*, pages 73–79.

Björn H. Jernudd and Michael J. Shapiro. 1989. *The Politics of Language Purism*.

Wenbin Jiang, Meng Sun, Yajuan Lü, Yating Yang, and Qun Liu. 2013. Discriminative learning with natural annotations: Word segmentation as a case study. In *Proceedings of ACL*, pages 761–769, Sofia, Bulgari.

Vidur Joshi, Matthew Peters, and Mark Hopkins. 2018. Extending a parser to distant domains using a few dozen partially annotated examples. In *Proceedings of ACL*, pages 1190–1199, Melbourne, Australia.

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

Zhenghua Li, Min Zhang, Yue Zhang, Zhanyi Liu, Wenliang Chen, Hua Wu, and Haifeng Wang. 2016b. Active learning for dependency parsing with partial annotation. In *Proceedings of ACL*, pages 344–357, Berlin, Germany.

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 495–504.

Yijia Liu, Yue Zhang, Wanxiang Che, Ting Liu, and Fan Wu. 2014. Domain adaptation for crf-based Chinese word segmentation using free annotations. In *Proceedings of EMNLP*, pages 864–874, Doha, Qatar.

Eric W. Noreen. 1989. *Computer-intensive methods for testing hypotheses: An introduction*. John Wiley & Sons, Inc., New York.

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

Manabu Sassano and Sadao Kurohashi. 2010. Using smaller constituents rather than sentences in active learning for japanese dependency parsing. In *Proceedings of ACL*, pages 356–365, Uppsala, Sweden.

Richard Sproat, William Gales, Chilin Shih, and Nancy Chang. 1987. A stochastic finite-state word-segmentation algorithm for Chinese. *Computational Linguistics*, 22:377–404.

Mitchell Stern, Jacob Andreas, and Dan Klein. 2017. A minimal span-based neural constituency parser. In *Proceedings of ACL*, pages 818–827, Vancouver, Canada.

Jinsong Su, Zhixing Tan, Deyi Xiong, Rongrong Ji, Xiaodong Shi, and Yang Liu. 2017. Lattice-based recurrent neurla network encoders for neural machine translation. In *Proceedings of AAAI*, pages 3302–3308, San Francisco, USA.

Zhiyang Teng and Yue Zhang. 2018. Two local models for neural constituent parsing. In *Proceedings of COLING*, pages 119–132.

Wenhui Wang and Baobao Chang. 2016. Graph-based dependency parsing with bidirectional lstm. In *Proceedings of ACL*, pages 2306–2315, Berlin, Germany.

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.

Jie Yang, Yue Zhang, and Fei Dong. 2017. Neural word segmentation with rich pretraining. In *Proceedings of ACL*, pages 839–849.

Jie Yang, Yue Zhang, and Shuailong Liang. 2019. Subword encoding in lattice LSTM for Chinese word segmentation. In *Proceedings of NAACL*, pages 2720–2725.

Shiwen Yu and Xuefeng Zhu. 2017. A dictionary of modern Chinese grammar information. `http://dx.doi.org/10.18170/DVN/EDQWIL`.

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.

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

Lujun Zhao, Qi Zhang, Peng Wang, and Xiaoyu Liu. 2018. Neural networks incorporating unlabeled and partially-labeled data for cross-domain Chinese word segmentation. In *Proceedings of IJCAI*, pages 4602–4608, Stockholm, Sweden.

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

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