Fast Coupled Sequence Labeling on Heterogeneous Annotations via Context-aware Pruning

Zhenghua Li, Jiayuan Chao, Min Zhang*, Jiwen Yang
Soochow University, Suzhou, China
{zhli13,minzhang,jwyang}@suda.edu.cn, chaojiayuan.china@gmail.com

Abstract

The recently proposed coupled sequence labeling is shown to be able to effectively exploit multiple labeled data with heterogeneous annotations but suffer from severe inefficiency problem due to the large bundled tag space (Li et al., 2015). In their case study of part-of-speech (POS) tagging, Li et al. (2015) manually design context-free tag-to-tag mapping rules with a lot of effort to reduce the tag space.

This paper proposes a context-aware pruning approach that performs token-wise constraints on the tag space based on contextual evidences, making the coupled approach efficient enough to be applied to the more complex task of joint word segmentation (WS) and POS tagging for the first time. Experiments show that using the large-scale People Daily as auxiliary heterogeneous data, the coupled approach can improve F-score by $95.55 - 94.88 = 0.67\%$ on WS, and by $90.58 - 89.49 = 1.09\%$ on joint WS&POS on Penn Chinese Treebank. All codes are released at http://hlt.suda.edu.cn/~zhli.

1 Introduction

In statistical natural language processing, manually labeled data is inevitable for model supervision, but is also very expensive to build. However, due to the long-debated differences in underlying linguistic theories or emphasis of application, there often exist multiple labeled corpora for the same or similar tasks following different annotation guidelines (Jiang et al., 2009). For instance, in Chinese language processing, Penn Chinese Treebank version 5 (CTB5) is a widely used benchmark data and contains about 20 thousand sentences annotated with word boundaries, part-of-speech (POS) tags, and syntactic structures (Xue et al., 2005; Xia, 2000), whereas People's Daily corpus (PD)[1] is a large-scale corpus annotated with words and POS tags, containing about 300 thousand sentences from the first half of 1998 of People's Daily newspaper (Yu et al., 2003). Table 1 gives an example with both CTB and PD annotations. We can see that CTB and PD differ in both word boundary standards and POS tag sets.

	Especially		our	nation	economy	declines	.
CTB	特别是/AD		我/PN	国/NN	经济/NN	下滑/VV	。/PU
PD	特别/d	是/v	我国/n		经济/n	下滑/v	。/w

Table 1: An example of heterogeneous annotations.

Previous work on exploiting heterogeneous data mainly focuses on indirect guide-feature methods. The basic idea is to use one resource to generate extra guide features on another resource (Jiang et al., 2009; Sun and Wan, 2012), which is similar to stacked learning (Nivre and McDonald, 2008). Li et al. (2015) propose a coupled sequence labeling approach that can directly learn and predict two heterogeneous annotations simultaneously. The basic idea is to transform a single-side tag into a set of bundled tags for weak supervision based on the idea of ambiguous labeling. Due to the huge size of the bundled tag space, their coupled model is extremely inefficient. They then carefully design tag-to-tag

*Correspondence author

[1] http://icl.pku.edu.cn/icl_groups/corpustagging.asp

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 753–762,
Austin, Texas, November 1-5, 2016.

mapping rules to constrain the search space. Their case study on POS tagging shows that the coupled model outperforms the guide-feature method. However, the requirement of manually designed mapping rules makes their approach less attractive, since such mapping rules may be very difficult to construct for more complex tasks such as joint word segmentation (WS) and POS tagging.

This paper proposes a context-aware pruning approach that can effectively solve the inefficiency problem of the coupled model, making coupled sequence labeling more generally applicable. Specifically, this work makes the following contributions:

(1) We propose and systematically compare two ways for realizing context-aware pruning, i.e., online and offline pruning. Experiments on POS tagging show that both online and offline pruning can greatly improve the model efficiency with little accuracy loss.

(2) We for the first time apply coupled sequence labeling to the more complex task of joint WS&POS tagging. Experiments show that online pruning works badly due to the much larger tag set while offline pruning works well. Further analysis gives a clear explanation and leads to more insights in learning from ambiguous labeling.

(3) Experiments on joint WS&POS tagging show that our coupled approach with offline pruning improves F-score by $95.55 - 94.88 = 0.67\%$ on WS, and by $90.58 - 89.49 = 1.09\%$ on joint WS&POS on CTB5-test over the baseline, and is also consistently better than the guide-feature method.

2 Coupled Sequence Labeling

Given an input sequence of n tokens, denoted by $\mathbf{x} = w_1...w_n$, coupled sequence tagging aims to simultaneously predict two tag sequences $\mathbf{t}^a = t_1^a...t_n^a$ and $\mathbf{t}^b = t_1^b...t_n^b$, where $t_i^a \in \mathcal{T}^a$ and $t_i^b \in \mathcal{T}^b$ $(1 \leq i \leq n)$, and $\mathcal{T}^a$ and $\mathcal{T}^b$ are two different predefined tag sets. Alternatively, we can view the two tag sequences as one bundled tag sequence $\mathbf{t} = [\mathbf{t}^a, \mathbf{t}^b] = [t_1^a, t_1^b]...[t_n^a, t_n^b]$, where $[t_i^a, t_i^b] \in \mathcal{T}^a \times \mathcal{T}^b$ is called a *bundled tag*.

In this work, we treat CTB as the first-side annotation and PD as the second-side. For POS tagging, $\mathcal{T}^a$ is the set of POS tags in CTB, and $\mathcal{T}^b$ is the set of POS tags in PD, and we ignore the word boundary differences in the two datasets, following Li et al. (2015). We have $|\mathcal{T}^a| = 33$ and $|\mathcal{T}^b| = 38$.

For joint WS&POS tagging, we employ the standard four-tag label set to mark word boundaries, among which `B`, `I`, `E` respectively represent that the concerned character situates at the *begining, inside, end* position of a word, and *S* represents a single-character word. Then, we concatenate word boundary labels with POS tags. For instance, the first three characters in Table 1 correspond to "特/`B@AD` 别/`I@AD` 是/`E@AD`" in CTB, and to "特/`B@d` 别/`E@d` 是/`S@v`" in PD. We have $|\mathcal{T}^a| = 99$ and $|\mathcal{T}^b| = 128$.

2.1 Coupled Conditional Random Field (CRF)

Following Li et al. (2015), we build the coupled sequence labeling model based on a bigram linear-chain CRF (Lafferty et al., 2001). The conditional probability of a bundled tag sequence $\mathbf{t}$ is:

$$p(\mathbf{t}|\mathbf{x}, \tilde{\mathcal{S}}; \theta) = \frac{e^{Score(\mathbf{x},\mathbf{t};\theta)}}{Z(\mathbf{x}, \tilde{\mathcal{S}}; \theta)}$$
$$Z(\mathbf{x}, \tilde{\mathcal{S}}; \theta) = \sum\nolimits_{\mathbf{t} \in \tilde{\mathcal{S}}} e^{Score(\mathbf{x},\mathbf{t};\theta)} \quad (1)$$

where θ is the feature weights; $Z(\mathbf{x}, \tilde{\mathcal{S}}; \theta)$ is the normalization factor; $\tilde{\mathcal{S}}$ is the search space including all legal tag sequences for $\mathbf{x}$. We use $\tilde{\mathcal{T}}_i \subseteq \mathcal{T}^a \times \mathcal{T}^b$ to denote the set of all legal tags for token w_i, so $\tilde{\mathcal{S}} = \tilde{\mathcal{T}}_1 \times \cdots \times \tilde{\mathcal{T}}_n$.

According to the linear-chain Markovian assumption, the score of a bundled tag sequence is:

$$Score(\mathbf{x}, \mathbf{t}; \theta) = \theta \cdot \mathbf{f}(\mathbf{x}, [\mathbf{t}^a, \mathbf{t}^b])$$
$$\sum_{i=1}^{n+1} \theta \cdot \begin{bmatrix} \mathbf{f}_{joint}(\mathbf{x}, i, [t_{i-1}^a, t_{i-1}^b], [t_i^a, t_i^b]) \\ \mathbf{f}_{sep\_a}(\mathbf{x}, i, t_{i-1}^a, t_i^a) \\ \mathbf{f}_{sep\_b}(\mathbf{x}, i, t_{i-1}^b, t_i^b) \end{bmatrix} \quad (2)$$

where $\mathbf{f}(\mathbf{x}, [\mathbf{t}^a, \mathbf{t}^b])$ is the accumulated sparse feature vector; $\mathbf{f}_{joint/sep\_a/sep\_b}(\mathbf{x}, i, t', t)$ share the same list of feature templates, and return local feature vectors for tagging w_{i-1} as t' and w_i as t.

Traditional single-side tagging models can only exploit a single set of separate features $\mathbf{f}_{sep\_a}(.)$ or $\mathbf{f}_{sep\_b}(.)$. In contrast, the coupled model makes

use of all three sets of features. Li et al. (2015) demonstrate that the joint features $\mathbf{f}_{joint}(.)$ capture the implicit mappings between heterogeneous annotations, and the separate features function as back-off features for alleviating the data sparseness problem of the joint features.

For the feature templates, we follow Li et al. (2015) and adopt those described in Zhang and Clark (2008) for POS tagging, and use those described in Zhang et al. (2014b) for joint WS&POS tagging.

2.2 Learn from Incomplete Data

The key challenge for coupled sequence labeling is that both CTB and PD are non-overlapping and each contains only one-side annotations. Based on the idea of ambiguous labeling, Li et al. (2015) first concatenate a single-side tag with many possible second-side tags, and then use the set of bundled tags as possibly-correct references during training.

Suppose $\mathbf{x} = w_1...w_n$ is a training sentence from CTB, and $\mathbf{t}^a = \check{t}_1^a...\check{t}_n^a$ is the manually labeled tag sequence. Then we define $\mathcal{T}_i = \{\check{t}_i^a\} \times \mathcal{T}^b$ as the set of possibly-correct bundled tags, and $\mathcal{S} = \mathcal{T}_1 \times \cdots \times \mathcal{T}_n$ as a exponential-size set of possibly-correct bundled tag sequences used for model supervision.

Given $\mathbf{x}$ and the whole legal search space $\tilde{\mathcal{S}}$, the probability of the possibly-correct space $\mathcal{S} \subseteq \tilde{\mathcal{S}}$ is:

$$p(\mathcal{S}|\mathbf{x}, \tilde{\mathcal{S}}; \theta) = \sum_{\mathbf{t} \in \mathcal{V}} p(\mathbf{t}|\mathbf{x}, \tilde{\mathcal{S}}; \theta) = \frac{Z(\mathbf{x}, \mathcal{S}; \theta)}{Z(\mathbf{x}, \tilde{\mathcal{S}}; \theta)} \quad (3)$$

where $Z(\mathbf{x}, \mathcal{S}; \theta)$ is analogous to $Z(\mathbf{x}, \tilde{\mathcal{S}}; \theta)$ in Eq. (3) but only sums over $\mathcal{S}$.

Given $\mathcal{D} = \{(\mathbf{x}_j, \mathcal{S}_j, \tilde{\mathcal{S}}_j)\}_{j=1}^N$, the gradient of the log likelihood is:

$$\begin{aligned}
&\frac{\partial \mathcal{LL}(\mathcal{D}; \theta)}{\partial \theta} = \frac{\partial \log \sum_j p(\mathcal{S}_j|\mathbf{x}_j, \tilde{\mathcal{S}}_j; \theta)}{\partial \theta} \\
=&\sum\nolimits_j \left(\frac{\partial \log Z(\mathbf{x}_j, \mathcal{S}_j; \theta)}{\partial \theta} - \frac{\partial \log Z(\mathbf{x}_j, \tilde{\mathcal{S}}_j; \theta)}{\partial \theta} \right) \\
=&\sum\nolimits_j \left(E_{\mathbf{t}|\mathbf{x}_j, \mathcal{S}_j; \theta}[\mathbf{f}(\mathbf{x}_j, \mathbf{t})] - E_{\mathbf{t}|\mathbf{x}_j, \tilde{\mathcal{S}}_j; \theta}[\mathbf{f}(\mathbf{x}_j, \mathbf{t})] \right)
\end{aligned} \quad (4)$$

where the two terms are the feature expectations under $\mathcal{S}_j$ and $\tilde{\mathcal{S}}_j$ respectively. And the detailed derivations are as follows:

$$\begin{aligned}
&\frac{\partial \log Z(\mathbf{x}, \mathcal{S}; \theta)}{\partial \theta} \\
=&\frac{1}{Z(\mathbf{x}, \mathcal{S}; \theta)} \times \frac{\partial \sum_{\mathbf{t} \in \mathcal{S}} e^{Score(\mathbf{x}, \mathbf{t}; \theta)}}{\partial \theta} \\
=&\sum_{\mathbf{t} \in \mathcal{S}} \left(\frac{e^{Score(\mathbf{x}, \mathbf{t}; \theta)}}{Z(\mathbf{x}, \mathcal{S}; \theta)} \times \frac{\partial Score(\mathbf{x}, \mathbf{t}; \theta)}{\partial \theta} \right) \\
=&\sum_{\mathbf{t} \in \mathcal{S}} p(\mathbf{t}|\mathbf{x}, \mathcal{S}; \theta) \times \mathbf{f}(\mathbf{x}, \mathbf{t}) \\
=&E_{\mathbf{t}|\mathbf{x}, \mathcal{S}; \theta}[\mathbf{f}(\mathbf{x}, \mathbf{t})]
\end{aligned} \quad (5)$$

Please notice that $\mathbf{t} = [\mathbf{t}^a, \mathbf{t}^b]$ denotes a bundled tag sequence in this context of coupled sequence labeling.

2.3 Efficiency Issue

Under *complete mapping*, each one-side tag is mapped to all the-other-side tags for constructing bundled tags, producing a very huge set of legal bundled tags $\tilde{\mathcal{T}}_i = \mathcal{T}^a \times \mathcal{T}^b$. Using the classic Forward-Backward algorithm, we still need $O(n \times |\mathcal{T}^a|^2 \times |\mathcal{T}^b|^2)$ time complexity to compute $E_{\mathbf{t}|\mathbf{x}, \tilde{\mathcal{S}}; \theta}[\mathbf{f}(\mathbf{x}, \mathbf{t})]$, which is prohibitively expensive.[2]

In order to improve efficiency, Li et al. (2015) propose to use a set of context-free tag-to-tag mapping rules for reducing the search space. For example, we may specify that the CTB POS tag "NN" can only be concatenated with a set of PD tags like "{n, vn, ns}".[3] With much effort, they propose a set of *relaxed mapping* rules that greatly reduces the number of bundled tags from $|\mathcal{T}^a| \times |\mathcal{T}^b| = 33 \times 38 = 1,254$ to 179 for POS tagging.

3 Context-aware Pruning

Using manually designed context-free tag-to-tag mapping rules to constrain the search space has two major drawbacks. On the one hand, for more complex problems such as joint WS&POS tagging, it becomes very difficult to design proper mapping rules due to the much larger tag set. On the other hand, the experimental results in Li et al. (2015)

[2] In contrast, computing $E_{\mathbf{t}|\mathbf{x}, \mathcal{S}; \theta}[\mathbf{f}(\mathbf{x}, \mathbf{t})]$ is not the bottleneck, since $|\mathcal{T}_i| = |\mathcal{T}^b|$ for CTB or $|\mathcal{T}_i| = |\mathcal{T}^a|$ for PD.

[3] Please refer to http://hlt.suda.edu.cn/~zhli/resources/pos-mapping-CTB-PD.html for their detailed mapping rules.

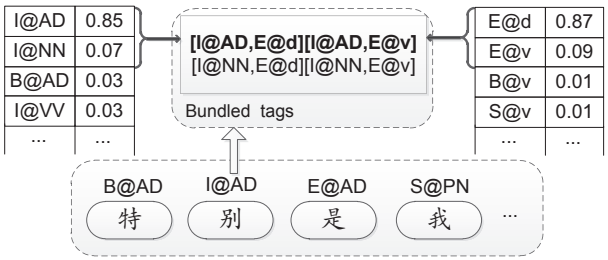

Figure 1: Illustration of context-aware pruning with $r = 2$ on a CTB training sentence.

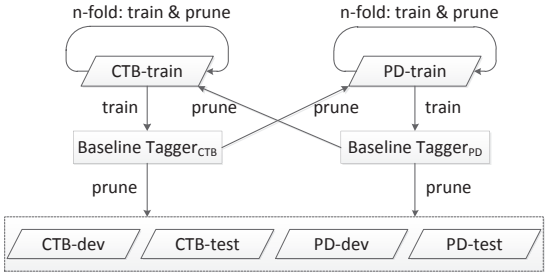

Figure 2: Workflow of offline pruning.

suggest that the coupled model can best learn the implicit context-sensitive mapping relationships between annotations under complete mapping, and imposing strict tag-to-tag mapping constraints usually hurts tagging accuracy.

In this work, our intuition is that the mapping relationships between heterogeneous annotations are highly context-sensitive. Therefore, we propose a context-aware pruning approach to more accurately capture such mappings, thus solving the efficiency issue. The basic idea is to consider only a small set of most likely bundled tags, instead of the whole bundled tag space $\mathcal{T}^a \times \mathcal{T}^b$, based on evidences of surrounding contexts. Specifically, for each token w_i, we only keep r one-side tags according to separate features $\mathbf{f}_{sep\_a/b}(.)$ for each side, and then use the remaining single-side tags to construct $\tilde{\mathcal{T}}_i$ and $\mathcal{T}_i$.

We use the second character "别/I@AD" in Fig. 1 as an example. We list the single-side tags in the descending order of their marginal probabilities according to $\mathbf{f}_{sep\_a/b}(.)$. Then we only keep $r = 2$ single-side tags, used as $\mathcal{T}_i^a$ and $\mathcal{T}_i^b$. Then $\tilde{\mathcal{T}}_i = \mathcal{T}^a \times \mathcal{T}^b$ contains the four bundled tags shown in the upper box, known as the whole possible tag set for searching. And $\mathcal{T}_i = \{\check{t}^a\} \times \mathcal{T}^b$ contains two bundled tags, as marked in bold, knowns as the possibly-correct tag set, since $\check{t}^a$ is the manually labeled tag. The case when the word has the second-side manually-labeled tag $\{\check{t}^b\}$ can be similarly handled.

Beside r, we use another hyper-parameter λ to further reduce the number of one-side tag candidates. The intuition is that in many cases, we may only need to use a smaller number $r' < r$ of possible candidates, since the remaining tags are very unlikely ones according to the marginal probabilities. Therefore, for each item w_i, we define r' as the smallest number of most likely candidate tags whose accumulative probability is larger than λ. Then, we only keep the $\min(r', r)$ most likely candidate tags.

We have $|\tilde{\mathcal{T}}_i| = r^2$ without considering the accumulated probability threshold λ. Thus, it requires $O(nr^4)$ time complexity to compute $E_{\mathbf{t}|\mathbf{x},\tilde{\mathcal{S}};\theta}[\mathbf{f}(\mathbf{x}, \mathbf{t})]$ using the Forward-Backward algorithm.

In the following, we propose two ways for realizing context-aware pruning, i.e., online and offline pruning. Their comparison and analysis are given in the experiment parts.

3.1 Online Pruning

The online pruning approach directly uses the coupled model to perform pruning. Given a sentence, we first use a subset of features $\mathbf{f}_{sep\_a}(.)$ and corresponding feature weights trained so far to compute marginal probabilities of first-side tags, and then analogously process the second-side tags based on $\mathbf{f}_{sep\_b}(.)$. This requires roughly the same time complexity as two baseline models. Then the marginal probabilities are used for pruning.

3.2 Offline Pruning

The offline pruning approach is a little bit more complex, and uses many additional single-side tagging models for pruning. Fig. 2 shows the workflow. Particularly, n-fold jack-knifing is adopted to perform pruning on the same-side training data. Finally, all training/dev/test datasets of CTB and PD are preprocessed in an offline way, so that each word in a sentence has a set of most likely CTB tags ($\mathcal{T}_i^a$) and another set of most likely PD tags ($\mathcal{T}_i^b$).

4 Experiment Settings

Data. Following Li et al. (2015), we use CTB5 and PD for the heterogeneous data. Under the standard

data split of CTB5, the training/dev/test datasets contain $16,091/803/1,910$ sentences respectively. For PD, we use the $46,815$ sentences in January 1998 as the training data, the first $2,000$ sentences in February as the development data, and the first $5,000$ sentences in June as the test data.

Evaluation Metrics. We use the standard token-wise tagging accuracy for POS tagging. For joint WS&POS tagging, besides character-wise tagging accuracy, we also use the standard precision (P), recall (R), and F-score of only words (WS) or POS-tagged words (WS&POS).

Parameter settings. Stochastic gradient descent (SGD) is adopted to train the baseline single-side tagging models, the guide-feature models, and the coupled models.[4]

For the coupled models, we directly follow the simple corpus-weighting strategy proposed in Li et al. (2015) to balance the contribution of the two datasets. We randomly sample $5,000$ CTB-train sentences and $5,000$ PD-train sentences, which are then merged and shuffled for one-iteration training. After each iteration, the coupled model is evaluated on both CTB-dev and PD-dev, providing us two single-side tag accuracies, one on CTB-side tags, and the other on PD-dev tags. Another advantage of using a subset of training data in one iteration is to monitor the training progress in smaller steps. For fair comparison, when building the baseline and guide-feature models, we also randomly sample $5,000$ training sentences from the whole training data for one-iteration training, and then report an tagging accuracy on development data. For all models, the training terminates if peak accuracies stop improving within 30 consecutive iterations, and we use the model that performs the best on development data for final evaluation on test data.

5 Experiments on POS Tagging

5.1 Parameter Tuning

For both online and offline pruning, we need to decide the maximum number of single-side tag candidates r and the accumulative probability threshold λ for further truncating the candidates. Table 2 shows the tagging accuracies and the averaged numbers of single-side tags for each token after pruning.

r	λ	Accuracy (%)		#Tags (pruned)	
		CTB5-dev	PD-dev	CTB-side	PD-side
Online Pruning					
2	0.98	94.25	95.03	2.0	2.0
4	0.98	95.06	95.66	3.9	4.0
8	**0.98**	**95.14**	**95.83**	6.3	7.4
16	0.98	95.12	95.81	7.8	14.1
8	0.90	**95.15**	95.79	3.7	6.3
8	0.95	95.13	**95.82**	5.1	7.1
8	0.99	**95.15**	95.74	7.4	7.9
8	1.00	**95.15**	95.76	8.0	8.0
Offline Pruning					
8	0.9999	94.95	96.05	4.1	5.1
16	**0.9999**	**95.15**	**96.09**	5.2	7.6
32	0.9999	95.13	96.09	5.5	9.3
16	0.99	94.42	95.77	1.6	2.2
16	0.999	95.02	**96.10**	2.6	4.0
16	0.99999	**95.10**	96.09	6.8	8.9

Table 2: POS tagging performance of online and offline pruning with different r and λ on CTB5 and PD.

The first major row tunes the two hyper-parameters for online pruning. We first fix $\lambda = 0.98$ and increase r from 2 to 8, leading to consistently improved accuracies on both CTB5-dev and PD-dev. No further improvement is gained with $r = 16$, indicating that tags below the top-8 are mostly very unlikely ones and thus insignificant for computing feature expectations. Then we fix $r = 8$ and try different λ. We find that λ has little effect on tagging accuracies but influences the numbers of remaining single-side tags. We choose $r = 8$ and $\lambda = 0.98$ for final evaluation.

The second major row tunes r and λ for offline pruning. Different from online pruning, λ has much greater effect on the number of remaining single-side tags. Under $\lambda = 0.9999$, increasing r from 8 to 16 leads to 0.20% accuracy improvement on CTB5-dev, but using $r = 32$ has no further gain. Then we fix $r = 16$ and vary λ from 0.99 to 0.99999. We choose $r = 16$ and $\lambda = 0.9999$ for offline pruning for final evaluation, which leaves each word with about 5.2 CTB-tags and 7.6 PD-tags on average.

[4] We use the implementation of SGD in CRFsuite (`http://www.chokkan.org/software/crfsuite/`), and set $b = 30$ as the batch-size and $C = 0.1$ as the regularization factor.

	Accuracy (%)		Speed
	CTB5-test	PD-test	Toks/Sec
Coupled (Offline)	**94.83**	95.90	246
Coupled (Online)	94.74	**95.95**	365
Coupled (No Prune)	94.58	95.79	3
Coupled (Relaxed)	94.63	95.87	127
Guide-feature	94.35	95.63	584
Baseline	94.07	95.82	1573
Li et al. (2012b)	94.60	—	—

Table 3: POS tagging performance of difference approaches on CTB5 and PD.

5.2 Main Results

Table 3 summarizes the accuracies on the test data and the tagging speed during the test phase. "Coupled (No Prune)" refers to the coupled model with complete mapping in Li et al. (2015), which maps each one-side tag to all the-other-side tags. "Coupled (Relaxed)" refers the coupled model with relaxed mapping in Li et al. (2015), which maps a one-side tag to a manually-designed small set of the-other-side tags. Li et al. (2012b) report the state-of-the-art accuracy on this CTB data, with a joint model of Chinese POS tagging and dependency parsing.

It is clear that both online and offline pruning greatly improve the efficiency of the coupled model by about two magnitudes, without the need of a carefully predefined set of tag-to-tag mapping rules.[5] Moreover, the coupled model with offline pruning achieves 0.76% accuracy improvement on CTB5-test over the baseline model, and 0.48% over our reimplemented guide-feature approach of Jiang et al. (2009). The gains on PD-test are marginal, possibly due to the large size of PD-train, similar to the results in Li et al. (2015).

6 Experiments on Joint WS&POS Tagging

6.1 Parameter Tuning

Table 4 shows results for tuning r and λ. From the results in the first major row, we can see that in the online pruning method, λ seems useless and r becomes the only threshold for pruning unlikely single-side tags. The accuracies are much inferior to those from the offline pruning approach. We believe that the accuracies can be further improved with larger r, which would nevertheless lead to severe inefficiency issue. Based on the results, we choose $r = 16$ and $\lambda = 1.00$ for final evaluation.

r	λ	Accuracy (%)		#Tags (pruned)	
		CTB5-dev	PD-dev	CTB-side	PD-side
Online Pruning					
8	1.00	90.41	89.91	8.0	8.0
16	0.95	90.65	90.22	15.9	16.0
16	0.99	90.77	**90.49**	16.0	16.0
16	**1.00**	**90.79**	**90.49**	16.0	16.0
Offline Pruning					
8	0.995	91.22	91.62	2.6	3.1
16	**0.995**	91.66	91.85	3.2	4.3
32	0.995	**91.67**	**91.87**	3.5	5.6
16	0.95	90.69	91.30	1.6	2.1
16	0.99	**91.64**	**91.92**	2.5	3.5
16	0.999	91.62	91.75	5.1	6.4

Table 4: WS&POS tagging performance of online and offline pruning with different r and λ on CTB5 and PD.

The second major row tries to decide r and λ for the offline pruning approach. Under $\lambda = 0.995$, increasing r from 8 to 16 improves accuracies both on CTB5-dev and PD-dev, but further using $r = 32$ leads to little gain. Then we fix $r = 16$ and vary λ from 0.95 to 0.999. Using $\lambda = 0.95$ leaves only 1.6 CTB tags and 2.1 PD tags for each character, but has a large accuracy drop. We choose $r = 16$ and $\lambda = 0.995$ for offline pruning for final evaluation, which leaves each character with 3.2 CTB-tags and 4.3 PD-tags on average.

6.2 Main Results

Table 5 summarizes the accuracies on the test data and the tagging speed (characters per second) during the test phase. "Coupled (No Prune)" is not tried due to the prohibitive tag set size in joint WS&POS tagging, and "Coupled (Relaxed)" is also skipped since it seems impossible to manually design reasonable tag-to-tag mapping rules in this case.

In terms of efficiency, the coupled model with offline pruning is on par with the baseline single-side tagging model.[6]

[5] Due to the model complexity of "Coupled (No Prune)", we discard all low-frequency (< 3) features in the training data to speed up training. This explains why "Coupled (No Prune)" has slightly lower accuracies than "Coupled (Relaxed)".

[6] The time estimation does not include the two separate processes of pruning single-side tags, which is approximately

	P/R/F (%) on CTB5-test		P/R/F (%) on PD-test		Speed
	Only WS	Joint WS&POS	Only WS	Joint WS&POS	Char/Sec
Coupled (Offline)	**95.65/95.46/95.55**	**90.68/90.49/90.58**	96.39/95.86/96.12	92.70/92.19/92.44	115
Coupled (Online)	95.17/94.71/94.94	89.80/89.37/89.58	95.76/95.45/95.60	91.71/91.41/91.56	26
Guide-feature	95.26/94.89/95.07	89.96/89.61/89.79	95.99/95.33/95.66	91.92/91.30/91.61	27
Baseline	95.00/94.77/94.88	89.60/89.38/89.49	**96.56/96.00/96.28**	**92.74/92.20/92.47**	119

Table 5: WS&POS tagging performance of difference approaches on CTB5 and PD.

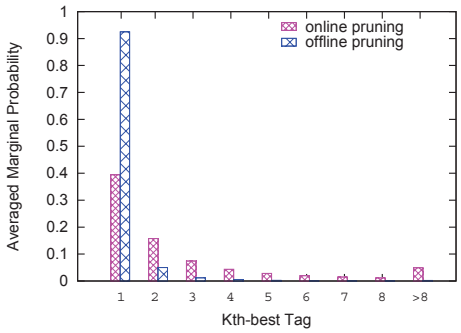

Figure 3: Probability distribution with online/offline pruning for the task of joint WS&POS.

In terms of F-score, the coupled model with offline pruning achieves 0.67% (WS) and 1.09% (WS&POS) gains on CTB5-test over the baseline model, and 0.48% (WS) and 0.79% (WS&POS) over our reimplemented guide-feature approach of Jiang et al. (2009). Similar to the case of POS tagging, the baseline model is very competitive on PD-test due to the large scale of PD-train.

6.3 Analysis

Online vs. offline pruning. The averaged numbers of single-side tags after pruning in Table 4 and 2), suggest that the online pruning approach works badly in assigning proper marginal probabilities to different tags. Our first guess is that in online pruning, the weights of separate features are optimized as a part of the coupled model, and thus producing somewhat flawed probabilities. However, our further analysis gives a more convincing explanation.

Fig. 3 compares the distribution of averaged probabilities of k^{th}-best CTB-side tags after online and offline pruning. The statistics are gathered on CTB5-test. Under online pruning, the averaged probability of the best tag is only about 0.4, which is surprisingly low and cannot be explained with the aforementioned improper optimization issue. Please note that both the online and offline models uses the best choices of r and λ based on Table 4, and are trained until convergence.

After a few trials of reducing the size of PD-train for training the coupled model, we realize that the underlying reason is that ambiguous labeling makes the probability mass more uniformly distributed, since for a PD-train sentence, the characters only have the gold-standard PD-side tags, and the model basically uses all CTB-side tags as gold-standard answers. Thanks to the CTB-train sentences, the model may be able to choose the correct tag, but inevitably becomes more indecisive at the same time due to the PD-train sentences.

In contrast, the offline pruning approach directly uses two baseline models for pruning, which is a job perfectly suitable for the baseline models. The entropy of the probability distribution for online pruning is about 1.524 while that for offline pruning is only 0.355.

Error distributions. To better understand the gains from the coupled approach, we show the F-score of specific POS tags for both the baseline and coupled models in Fig. 4, in the descending order of absolute F-score improvements. The largest improvement is from words tagged as "LB" (mostly for the word "被", marking a certain type of passive construction), and the F-score increases by $65.22 - 54.55 = 10.67\%$. Nearly all POS tags have more or less F-score improvement. Due to the space limit, we only show the tags with more than 2.0% improvement. The most noticeable exception is that F-score drops by $84.80 - 86.49 = -1.69\%$ for words tagged as "OD" (ordinal numbers, as opposed to cardinal numbers).

In terms of words, we find the largest gain is from "卢森博格/NR" (Luxemburgo, place name), which appears 11 times in CTB5-test, with an absolute

equal to the time of two baseline models.

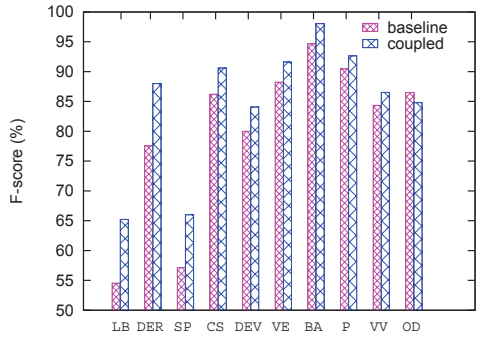

Figure 4: F-score comparison between the baseline and coupled WS&POS tagging models on different CTB POS tags.

	F (%) on CTB5X-test	
	Only WS	Joint WS&POS
Coupled (Offline)	**98.01**	**94.39**
Guide-feature	97.96	94.06
Baseline	97.37	93.23
Sun and Wan (2012)	—	94.36
Jiang et al. (2009)	98.23	94.03

Table 6: WS&POS tagging performance of difference approaches on CTB5X and PD.

improvement of $90.00 - 16.67 = 73.33\%$ in recall ratio. The reason is that PD-train contains a lot of related words such as "卢森堡" (Luxembourg, place name) and "克拉泽博格" (Krayzelburg, person name) while CTB5-train has none.

6.4 Comparison with Previous Work

In order to compare with previous work, we also run our models on CTB5X and PD, where CTB5X adopts a different data split of CTB5 and is widely used in previous research on joint WS&POS tagging (Jiang et al., 2009; Sun and Wan, 2012). CTB5X-dev/test only contain $352/348$ sentences respectively. Table 6 presents the F scores on CTB5X-test. We can see that the coupled model with offline pruning achieves 0.64% (WS) and 1.16% (WS&POS) F-score improvements over the baseline model, and 0.05% (WS) and 0.33% (WS&POS) over the guide-feature approach.

The original guide-feature method in Jiang et al. (2009) achieves 98.23% and 94.03% F-score, which is very close to the results of our reimplemented model. The sub-word stacking approach of Sun and Wan (2012) can be understood as a more complex variant of the basic guide-feature method.[7]

The results on both the larger CTB5-test (in Table 5) and CTB5X-test suggest that the coupled approach is more consistent and robust than the guide-feature method. The reason may be two-fold. First, in the coupled approach, the model is able to actively learn the implicit mappings between two sets of annotations, whereas the guide-feature model can only passively learn when to trust the automatically produced tags. Second, the coupled approach can directly learn from both heterogeneous training datasets, thus covering more phenomena of language usage.

7 Related Work

A lot of research has been devoted to design an effective way to exploit non-overlapping heterogeneous labeled data, especially in Chinese language processing, where such heterogeneous resources are ubiquitous due to historical reasons. Jiang et al. (2009) first propose the guide-feature approach, which is similar to stacked learning (Nivre and McDonald, 2008), for joint WS&POS tagging on CTB and PD. Sun and Wan (2012) further extend the guide-feature method and propose a more complex sub-word stacking approach. Qiu et al. (2013) propose a linear coupled model similar to that of Li et al. (2015). The key difference is that the model of Qiu et al. (2013) only uses separate features, while Li et al. (2015) and this work explore joint features as well.

Li et al. (2012a) apply the guide-feature idea to dependency parsing on CTB and PD. Zhang et al. (2014a) extend a shift-reduce dependency parsing model in order to simultaneously learn and produce two heterogeneous parse trees, which however assumes the existence of training data with both-side annotations.

Our context-aware pruning approach is similar to coarse-to-fine pruning in parsing community (Koo and Collins, 2010; Rush and Petrov, 2012), which is a useful technique that allows us to use very complex parsing models without too much efficiency cost. The idea is first to use a simple and basic off-shelf model to prune the search space and only keep highly likely dependency links, and then let the complex

[7] Sun and Wan (2012) achieve 94.68% F-score on CTB5X-test by further employing a re-training strategy.

model infer in the remaining search space. Weiss and Taskar (2010) propose structured prediction cascades: a sequence of increasingly complex models that progressively filter the space of possible outputs, and provide theoretical generalization bounds on a novel convex loss function that balances pruning error with pruning efficiency.

This work is also closely related with multi-task learning, which aims to jointly learn multiple related tasks with the benefit of using interactive features under a share representation (Ben-David and Schuller, 2003; Ando and Zhang, 2005; Parameswaran and Weinberger, 2010). However, as far as we know, multi-task learning usually assumes the existence of data with labels for multiple tasks at the same time, which is unavailable in our scenario, making our problem more particularly difficult.

Our coupled CRF model is similar to a factorial CRF (Sutton et al., 2004), in the sense that the bundled tags can be factorized into two connected latent variables. Initially, factorial CRFs are designed to jointly model two related (and typically hierarchical) sequential labeling tasks, such as POS tagging and chunking. In this work, our coupled CRF model jointly handles two same tasks with different annotation schemes. Moreover, this work provides a natural way to learn from incomplete annotations where one sentence only contains one-side labels.

Learning with ambiguous labeling is previously explored for classification (Jin and Ghahramani, 2002), sequence labeling (Dredze et al., 2009), parsing (Riezler et al., 2002; Täckström et al., 2013). Recently, researchers propose to derive natural annotations from web data to supervise Chinese word segmentation models in the form of ambiguous labeling (Jiang et al., 2013; Liu et al., 2014; Yang and Vozila, 2014).

8 Conclusion

This paper proposes a context-aware pruning approach for the coupled sequence labeling model of Li et al. (2015). The basic idea is to more accurately constrain the bundled tag space of a token according to its contexts in the sentence, instead of using heuristic context-free tag-to-tag mapping rules in the original work. We propose and compare two different ways of realizing pruning, i.e., online and offline pruning. In summary, extensive experiments leads to the following findings.

(1) Offline pruning works well on both POS tagging and joint WS&POS tagging, whereas online pruning only works well on POS tagging but fails on joint WS&POS tagging due to the much larger tag set. Further analysis shows that the reason is that under online pruning, ambiguous labeling during training makes the probabilities of single-side tags more evenly distributed.

(2) In terms of tagging accuracy and F-score, the coupled approach with offline pruning outperforms the baseline single-side tagging model by large margin, and is also consistently better than the mainstream guide-feature method on both POS tagging and joint WS&POS tagging.

Acknowledgments

The authors would like to thank the anonymous reviewers for the helpful comments. We are very grateful to Meishan Zhang for inspiring us to use online pruning to improve the efficiency of the coupled approach. We also thank Wenliang Chen for the helpful discussions. This work was supported by National Natural Science Foundation of China (Grant No. 61525205, 61502325, 61432013).

References

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learn Research*, 6:1817–1853.

Shai Ben-David and Reba Schuller. 2003. Exploiting task relatedness for multiple task learning. In *COLT*.

Mark Dredze, Partha Pratim Talukdar, and Koby Crammer. 2009. Sequence learning from data with multiple labels. In *ECML/PKDD Workshop on Learning from Multi-Label Data*.

Wenbin Jiang, Liang Huang, and Qun Liu. 2009. Automatic adaptation of annotation standards: Chinese word segmentation and POS tagging – a case study. In *Proceedings of ACL*, pages 522–530.

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.

Rong Jin and Zoubin Ghahramani. 2002. Learning with multiple labels. In *Proceedings of NIPS*.

Terry Koo and Michael Collins. 2010. Efficient third-order dependency parsers. In *ACL*, pages 1–11.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML 2001*, pages 282–289.

Zhenghua Li, Wanxiang Che, and Ting Liu. 2012a. Exploiting multiple treebanks for parsing with quasi-synchronous grammar. In *ACL*, pages 675–684.

Zhenghua Li, Min Zhang, Wanxiang Che, and Ting Liu. 2012b. A separately passive-aggressive training algorithm for joint POS tagging and dependency parsing. In *COLING*, pages 1681–1698.

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.

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.

Joakim Nivre and Ryan McDonald. 2008. Integrating graph-based and transition-based dependency parsers. In *Proceedings of ACL*, pages 950–958.

S. Parameswaran and K.Q. Weinberger. 2010. Large margin multi-task metric learning. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 1867–1875.

Xipeng Qiu, Jiayi Zhao, and Xuanjing Huang. 2013. Joint Chinese word segmentation and POS tagging on heterogeneous annotated corpora with multiple task learning. In *Proceedings of EMNLP*, pages 658–668.

Stefan Riezler, Tracy H. King, Ronald M. Kaplan, Richard Crouch, John T. III Maxwell, and Mark Johnson. 2002. Parsing the wall street journal using a lexical-functional grammar and discriminative estimation techniques. In *Proceedings of ACL*, pages 271–278.

Alexander Rush and Slav Petrov. 2012. Vine pruning for efficient multi-pass dependency parsing. In *Proceedings of NAACL-2012*, pages 498–507.

Weiwei Sun and Xiaojun Wan. 2012. Reducing approximation and estimation errors for Chinese lexical processing with heterogeneous annotations. In *Proceedings of ACL*, pages 232–241.

Charles Sutton, Khashayar Rohanimanesh, and Andrew McCallum. 2004. Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. In *International Conference on Machine Learning (ICML)*.

Oscar Täckström, Ryan McDonald, and Joakim Nivre. 2013. Target language adaptation of discriminative transfer parsers. In *Proceedings of NAACL*, pages 1061–1071.

David Weiss and Ben Taskar. 2010. Structured prediction cascades. In *Proceedings of International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Fei Xia. 2000. The part-of-speech tagging guidelines for the penn Chinese treebank 3.0. In *Technical Report, Linguistic Data Consortium*.

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

Fan Yang and Paul Vozila. 2014. Semi-supervised Chinese word segmentation using partial-label learning with conditional random fields. In *Proceedings of EMNLP*, pages 90–98.

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.

Yue Zhang and Stephen Clark. 2008. Joint word segmentation and POS tagging using a single perceptron. In *Proceedings of ACL-08: HLT*, pages 888–896.

Meishan Zhang, Wanxiang Che, Yanqiu Shao, and Ting Liu. 2014a. Jointly or separately: Which is better for parsing heterogeneous dependencies? In *Proceedings of COLING*, pages 530–540.

Meishan Zhang, Yue Zhang, Wanxiang Che, and Ting Liu. 2014b. Character-level Chinese dependency parsing. In *Proceedings of ACL*, pages 1326–1336.