Bilingually-Guided Monolingual Dependency Grammar Induction

Kai Liu[†§], **Yajuan Lü**[†], **Wenbin Jiang**[†], **Qun Liu**[‡†]

†Key Laboratory of Intelligent Information Processing
Institute of Computing Technology, Chinese Academy of Sciences
P.O. Box 2704, Beijing 100190, China
{liukai,lvyajuan,jiangwenbin,liuqun}@ict.ac.cn
‡Centre for Next Generation Localisation
Faculty of Engineering and Computing, Dublin City University
qliu@computing.dcu.ie
§University of Chinese Academy of Sciences

Abstract

This paper describes a novel strategy for automatic induction of a monolingual dependency grammar under the guidance of bilingually-projected dependency. By moderately leveraging the dependency information projected from the parsed counterpart language, and simultaneously mining the underlying syntactic structure of the language considered, it effectively integrates the advantages of bilingual projection and unsupervised induction, so as to induce a monolingual grammar much better than previous models only using bilingual projection or unsupervised induction. We induced dependency grammar for five different languages under the guidance of dependency information projected from the parsed English translation, experiments show that the bilingually-guided method achieves a significant improvement of 28.5% over the unsupervised baseline and 3.0% over the best projection baseline on average.

1 Introduction

In past decades supervised methods achieved the state-of-the-art in constituency parsing (Collins, 2003; Charniak and Johnson, 2005; Petrov et al., 2006) and dependency parsing (McDonald et al., 2005a; McDonald et al., 2006; Nivre et al., 2006; Nivre et al., 2007; Koo and Collins, 2010). For supervised models, the human-annotated corpora on which models are trained, however, are expensive and difficult to build. As alternative strategies, methods which utilize raw texts have been investigated recently, including unsupervised methods which use only raw texts (Klein and Manning, 2004; Smith and Eisner, 2005; William et al., 2009), and semi-supervised methods (Koo et al., 2008) which use both raw texts and annotated corpus. And there are a lot of efforts have also been devoted to bilingual projection (Chen et al., 2010), which resorts to bilingual text with one language parsed, and projects the syntactic information from the parsed language to the unparsed one (Hwa et al., 2005; Ganchev et al., 2009).

In dependency grammar induction, unsupervised methods achieve continuous improvements in recent years (Klein and Manning, 2004; Smith and Eisner, 2005; Bod, 2006; William et al., 2009; Spitkovsky et al., 2010). Relying on a predefined distributional assumption and iteratively maximizing an approximate indicator (entropy, likelihood, etc.), an unsupervised model usually suffers from two drawbacks, i.e., lower performance and higher computational cost. On the contrary, bilingual projection (Hwa et al., 2005; Smith and Eisner, 2009; Jiang and Liu, 2010) seems a promising substitute for languages with a large amount of bilingual sentences and an existing parser of the counterpart language. By projecting syntactic structures directly (Hwa et al., 2005; Smith and Eisner, 2009; Jiang and Liu, 2010) across bilingual texts or indirectly across multilingual texts (Snyder et al., 2009; McDonald et al., 2011; Naseem et al., 2012), a better dependency grammar can be easily induced, if syntactic isomorphism is largely maintained between target and source languages.

Unsupervised induction and bilingual projection run according to totally different principles, the former mines the underlying structure of the monolingual language, while the latter leverages the syntactic knowledge of the parsed counter-

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 1063–1072,
Sofia, Bulgaria, August 4-9 2013. ©2013 Association for Computational Linguistics

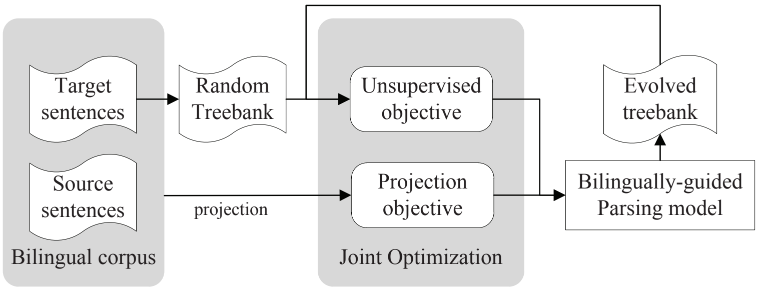

Figure 1: Training the bilingually-guided parsing model by iteration.

part language. Considering this, we propose a novel strategy for automatically inducing a monolingual dependency grammar under the guidance of bilingually-projected dependency information, which integrates the advantage of bilingual projection into the unsupervised framework. A randomly-initialized monolingual treebank evolves in a self-training iterative procedure, and the grammar parameters are tuned to simultaneously maximize both the monolingual likelihood and bilingually-projected likelihood of the evolving treebank. The monolingual likelihood is similar to the optimization objectives of conventional unsupervised models, while the bilingually-projected likelihood is the product of the projected probabilities of dependency trees. By moderately leveraging the dependency information projected from the parsed counterpart language, and simultaneously mining the underlying syntactic structure of the language considered, we can automatically induce a monolingual dependency grammar which is much better than previous models only using bilingual projection or unsupervised induction. In addition, since both likelihoods are fundamentally factorized into dependency edges (of the hypothesis tree), the computational complexity approaches to unsupervised models, while with much faster convergence. We evaluate the final automatically-induced dependency parsing model on 5 languages. Experimental results show that our method significantly outperforms previous work based on unsupervised method or indirect/direct dependency projection, where we see an average improvement of 28.5% over unsupervised baseline on all languages, and the improvements are 3.9%/3.0% over indirect/direct baselines. And our model achieves the most significant gains on Chinese, where the improvements are 12.0%, 4.5% over indirect and direct projection baselines respectively.

In the rest of the paper, we first describe the unsupervised dependency grammar induction framework in section 2 (where the unsupervised optimization objective is given), and introduce the bilingual projection method for dependency parsing in section 3 (where the projected optimization objective is given); Then in section 4 we present the bilingually-guided induction strategy for dependency grammar (where the two objectives above are jointly optimized, as shown in Figure 1). After giving a brief introduction of previous work in section 5, we finally give the experimental results in section 6 and conclude our work in section 7.

2 Unsupervised Dependency Grammar Induction

In this section, we introduce the unsupervised objective and the unsupervised training algorithm which is used as the framework of our bilingually-guided method. Unlike previous unsupervised work (Klein and Manning, 2004; Smith and Eisner, 2005; Bod, 2006), we select a self-training approach (similar to hard EM method) to train the unsupervised model. And the framework of our unsupervised model builds a random treebank on the monolingual corpus firstly for initialization and trains a discriminative parsing model on it. Then we use the parser to build an evolved treebank with the 1-best result for the next iteration run. In this way, the parser and treebank evolve in an iterative way until convergence. Let's introduce the parsing objective firstly:

Define e_i as the i^{th} word in monolingual sentence E; $d_{e_{ij}}$ denotes the word pair dependency relationship ($e_i \rightarrow e_j$). Based on the features around $d_{e_{ij}}$, we can calculate the probability $Pr(y|d_{e_{ij}})$ that the word pair $d_{e_{ij}}$ can form a dependency arc as:

$$Pr(y|d_{e_{ij}}) = \frac{1}{Z(d_{e_{ij}})} exp(\sum_n \lambda_n \cdot f_n(d_{e_{ij}}, y)) \quad (1)$$

where y is the category of the relationship of $d_{e_{ij}}$: $y = +$ means it is the probability that the word pair $d_{e_{ij}}$ can form a dependency arc and $y = -$ means the contrary. λ_n denotes the weight for feature function $f_n(d_{e_{ij}}, y)$, and the features we used are presented in Table 1 (Section 6). $Z(d_{e_{ij}})$ is a normalizing constant:

$$Z(d_{e_{ij}}) = \sum_y exp(\sum_n \lambda_n \cdot f_n(d_{e_{ij}}, y)) \quad (2)$$

Given a sentence E, parsing a dependency tree is to find a dependency tree D_E with maximum probability P_E:

$$P_E = \arg\max_{D_E} \prod_{d_{e_{ij}} \in D_E} Pr(+|d_{e_{ij}}) \quad (3)$$

2.1 Unsupervised Objective

We select a simple classifier objective function as the unsupervised objective function which is instinctively in accordance with the parsing objective:

$$\theta(\lambda) = \prod_{d_e \in D_{\overline{E}}} Pr(+|d_e) \prod_{d_e \in \widetilde{D}_{\overline{E}}} Pr(-|d_e) \quad (4)$$

where $\overline{E}$ is the monolingual corpus and $E \in \overline{E}$, $D_{\overline{E}}$ is the treebank that contains all D_E in the corpus, and $\widetilde{D}_{\overline{E}}$ denotes all other possible dependency arcs which do not exist in the treebank.

Maximizing the Formula (4) is equivalent to maximizing the following formula:

$$\begin{aligned} \theta_1(\lambda) = & \sum_{d_e \in D_{\overline{E}}} \log Pr(+|d_e) \\ & + \sum_{d_e \in \widetilde{D}_{\overline{E}}} \log Pr(-|d_e) \end{aligned} \quad (5)$$

Since the size of edges between $D_{\overline{E}}$ and $\widetilde{D}_{\overline{E}}$ is disproportionate, we use an empirical value to reduce the impact of the huge number of negative instances:

$$\begin{aligned} \theta_2(\lambda) = & \sum_{d_e \in D_{\overline{E}}} \log Pr(+|d_e) \\ & + \frac{|D_{\overline{E}}|}{|\widetilde{D}_{\overline{E}}|} \sum_{d_e \in \widetilde{D}_{\overline{E}}} \log Pr(-|d_e) \end{aligned} \quad (6)$$

where $|x|$ is the size of x.

Algorithm 1 Training unsupervised model

```
1: build random $D_{\overline{E}}$
2: $\lambda \leftarrow train(D_{\overline{E}}, \widetilde{D}_{\overline{E}})$
3: repeat
4:     for each $E \in \overline{E}$ do                    ▷ E step
5:         $D_{\overline{E}} \leftarrow parse(E, \lambda)$
6:     $\lambda \leftarrow train(D_{\overline{E}}, \widetilde{D}_{\overline{E}})$          ▷ M step
7: until convergence
```

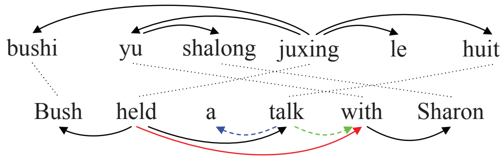

Figure 2: Projecting a Chinese dependency tree to English side according to DPA. Solid arrows are projected dependency arcs; dashed arrows are missing dependency arcs.

2.2 Unsupervised Training Algorithm

Algorithm 1 outlines the unsupervised training in its entirety, where the treebank $D_{\overline{E}}$ and unsupervised parsing model with λ are updated iteratively.

In line 1 we build a random treebank $D_{\overline{E}}$ on the monolingual corpus, and then train the parsing model with it (line 2) through a training procedure $train(\cdot, \cdot)$ which needs $D_{\overline{E}}$ and $\widetilde{D}_{\overline{E}}$ as classification instances. From line 3-7, we train the unsupervised model in self training iterative procedure, where line 4-5 are similar to the E-step in EM algorithm where calculates objective instead of expectation of 1-best tree (line 5) which is parsed according to the parsing objective (Formula 3) by parsing process $parse(\cdot, \cdot)$, and update the tree bank with the tree. Similar to M-step in EM, the algorithm maximizes the whole treebank's unsupervised objective (Formula 6) through the training procedure (line 6).

3 Bilingual Projection of Dependency Grammar

In this section, we introduce our projection objective and training algorithm which trains the model with arc instances.

Because of the heterogeneity between different languages and word alignment errors, projection methods may contain a lot of noises. Take Figure 2 as an example, following the Direct Projection Algorithm (DPA) (Hwa et al., 2005) (Section 5), the dependency relationships between words can be directly projected from the source

Algorithm 2 Training projection model

```
$D_P, D_N \leftarrow proj(\overline{F}, D_{\overline{F}}, A, \overline{E})$
repeat                                        $\triangleright\, train(D_P, D_N)$
    $\nabla\phi \leftarrow grad(D_P, D_N, \phi(\lambda))$
    $\lambda \leftarrow climb(\phi, \nabla\phi, \lambda)$
until maximization
```

language to the target language. Therefore, we can hardly obtain a treebank with complete trees through direct projection. So we extract projected discrete dependency arc instances instead of treebank as training set for the projected grammar induction model.

3.1 Projection Objective

Correspondingly, we select an objective which has the same form with the unsupervised one:

$$\begin{aligned}\phi(\lambda) = &\sum_{d_e \in D_P} \log Pr(+|d_e) \\ &+ \sum_{d_e \in D_N} \log Pr(-|d_e)\end{aligned} \tag{7}$$

where D_P is the positive dependency arc instance set, which is obtained by direct projection methods (Hwa et al., 2005; Jiang and Liu, 2010) and D_N is the negative one.

3.2 Projection Algorithm

Basically, the training procedure in line 2,7 of Algorithm 1 can be divided into smaller iterative steps, and Algorithm 2 outlines the training step of projection model with instances. $\overline{F}$ in Algorithm 2 is source sentences in bilingual corpus, and A is the alignments. Function $grad(\cdot,\cdot,\cdot)$ gives the gradient ($\nabla\phi$) and the objective is optimized with a generic optimization step (such as an LBFGS iteration (Zhu et al., 1997)) in the subroutine $climb(\cdot,\cdot,\cdot)$.

4 Bilingually-Guided Dependency Grammar Induction

This section presents our bilingually-guided grammar induction model, which incorporates unsupervised framework and bilingual projection model through a joint approach.

According to following observation: unsupervised induction model mines underlying syntactic structure of the monolingual language, however, it is hard to find good grammar induction in the exponential parsing space; bilingual projection obtains relatively reliable syntactic knowledge of the parsed counterpart, but it possibly contains a lot of noises (e.g. Figure 2). We believe that unsupervised model and projection model can complement each other and a joint model which takes better use of both unsupervised parse trees and projected dependency arcs can give us a better parser.

Based on the idea, we propose a novel strategy for training monolingual grammar induction model with the guidance of unsupervised and bilingually-projected dependency information. Figure 1 outlines our bilingual-guided grammar induction process in its entirety. In our method, we select compatible objectives for unsupervised and projection models, in order to they can share the same grammar parameters. Then we incorporate projection model into our iterative unsupervised framework, and jointly optimize unsupervised and projection objectives with evolving treebank and constant projection information respectively. In this way, our bilingually-guided model's parameters are tuned to simultaneously maximizing both monolingual likelihood and bilingually-projected likelihood by 4 steps:

1. Randomly build treebank on target sentences for initialization, and get the projected arc instances through projection from bitext.

2. Train the bilingually-guided grammar induction model by multi-objective optimization method with unsupervised objective and projection objective on treebank and projected arc instances respectively.

3. Use the parsing model to build new treebank on target language for next iteration.

4. Repeat steps 1, 2 and 3 until convergence.

The unsupervised objective is optimized by the loop—”tree bank→optimized model→new tree bank”. The treebank is evolved for runs. The unsupervised model gets projection constraint implicitly from those parse trees which contain information from projection part. The projection objective is optimized by the circulation—”projected instances→optimized model”, these projected instances will not change once we get them.

The iterative procedure proposed here is not a co-training algorithm (Sarkar, 2001; Hwa et al., 2003), because the input of the projection objective is static.

4.1 Joint Objective

For multi-objective optimization method, we employ the classical weighted-sum approach which just calculates the weighted linear sum of the objectives:

$$OBJ = \sum_{m} weight_m obj_m \tag{8}$$

We combine the unsupervised objective (Formula (6)) and projection objective (Formula (7)) together through the weighted-sum approach in Formula (8):

$$\ell(\lambda) = \alpha\theta_2(\lambda) + (1-\alpha)\phi(\lambda) \tag{9}$$

where $\ell(\lambda)$ is our weight-sum objective. And α is a mixing coefficient which reflects the relative confidence between the unsupervised and projection objectives. Equally, α and $(1-\alpha)$ can be seen as the weights in Formula (8). In that case, we can use a single parameter α to control both weights for different objective functions. When $\alpha = 1$ it is the unsupervised objective function in Formula (6). Contrary, if $\alpha = 0$, it is the projection objective function (Formula (7)) for projected instances.

With this approach, we can optimize the mixed parsing model by maximizing the objective in Formula (9). Though the function (Formula (9)) is an interpolation function, we use it for training instead of parsing. In the parsing procedure, our method calculates the probability of a dependency arc according to the Formula (2), while the interpolating method calculates it by:

$$\begin{aligned} Pr(y|d_{e_{ij}}) =& \alpha Pr_1(y|d_{e_{ij}}) \\ &+ (1-\alpha)Pr_2(y|d_{e_{ij}}) \end{aligned} \tag{10}$$

where $Pr_1(y|d_{e_{ij}})$ and $Pr_2(y|d_{e_{ij}})$ are the probabilities provided by different models.

4.2 Training Algorithm

We optimize the objective (Formula (9)) via a gradient-based search algorithm. And the gradient with respect to λ_k takes the form:

$$\nabla\ell(\lambda_k) = \alpha\frac{\partial\theta_2(\lambda)}{\partial\lambda_k} + (1-\alpha)\frac{\partial\phi(\lambda)}{\partial\lambda_k} \tag{11}$$

Algorithm 3 outlines our joint training procedure, which tunes the grammar parameter λ simultaneously maximize both unsupervised objective and projection objective. And it incorporates unsupervised framework and projection model algorithm together. It is grounded on the work which uses features in the unsupervised model (Berg-Kirkpatrick et al., 2010).

Algorithm 3 Training joint model

```
1: $D_P, D_N \leftarrow proj(\overline{F}, D_{\overline{F}}, A, \overline{E})$
2: $build\ random\ D_{\overline{E}}$
3: $\lambda \leftarrow train(D_P, D_N)$
4: repeat
5:     for each $E \in \overline{E}$ do                         ▷ E step
6:         $D_{\overline{E}} \leftarrow parse(E, \lambda)$
7:     $\nabla\ell(\lambda) \leftarrow grad(D_{\overline{E}}, \widetilde{D}_{\overline{E}}, D_P, D_N, \ell(\lambda))$
8:     $\lambda \leftarrow climb(\ell(\lambda), \nabla\ell(\lambda), \lambda)$                ▷ M step
9: until convergence
```

In line 1, 2 we get projected dependency instances from source side according to projection methods and build a random treebank (step 1). Then we train an initial model with projection instances in line 3. From line 4-9, the objective is optimized with a generic optimization step in the subroutine $climb(\cdot,\cdot,\cdot,\cdot,\cdot)$. For each sentence we parse its dependency tree, and update the tree into the treebank (step 3). Then we calculate the gradient and optimize the joint objective according to the evolved treebank and projected instances (step 2). Lines 5-6 are equivalent to the E-step of the EM algorithm, and lines 7-8 are equivalent to the M-step.

5 Related work

The DMV (Klein and Manning, 2004) is a single-state head automata model (Alshawi, 1996) which is based on POS tags. And DMV learns the grammar via inside-outside re-estimation (Baker, 1979) without any smoothing, while Spitkovsky et al. (2010) utilizes smoothing and learning strategy during grammar learning and William et al. (2009) improves DMV with richer context.

The dependency projection method DPA (H-wa et al., 2005) based on Direct Correspondence Assumption (Hwa et al., 2002) can be described as: if there is a pair of source words with a dependency relationship, the corresponding aligned words in target sentence can be considered as having the same dependency relationship equivalently (e.g. Figure 2). The Word Pair Classification (WPC) method (Jiang and Liu, 2010) modifies the DPA method and makes it more robust. Smith and Eisner (2009) propose an adaptation method founded on quasi-synchronous grammar features

Type	Feature Template		
Unigram	$word_i$	pos_i	$word_i \circ pos_i$
	$word_j$	pos_j	$word_j \circ pos_j$
Bigram	$word_i \circ pos_j$	$word_j \circ pos_i$	$pos_i \circ pos_j$
	$word_i \circ word_j$	$word_i \circ pos_i \circ word_j$	$word_i \circ word_j \circ pos_j$
	$word_i \circ pos_i \circ pos_j$	$pos_i \circ word_j \circ pos_j$	
	$word_i \circ pos_i \circ word_j \circ pos_j$		
Surrounding	$pos_{i-1} \circ pos_i \circ pos_j$	$pos_i \circ pos_{i+1} \circ pos_j$	$pos_i \circ pos_{j-1} \circ pos_j$
	$pos_i \circ pos_j \circ pos_{j+1}$	$pos_{i-1} \circ pos_i \circ pos_{j-1}$	$pos_i \circ pos_{i+1} \circ pos_{j+1}$
	$pos_{i-1} \circ pos_{j-1} \circ pos_j$	$pos_{i+1} \circ pos_j \circ pos_{j+1}$	$pos_{i-1} \circ pos_i \circ pos_{j+1}$
	$pos_i \circ pos_{i+1} \circ pos_{j-1}$	$pos_{i-1} \circ pos_j \circ pos_{j+1}$	$pos_{i+1} \circ pos_{j-1} \circ pos_j$
	$pos_{i-1} \circ pos_i \circ pos_{j-1} \circ pos_j$	$pos_i \circ pos_{i+1} \circ pos_j \circ pos_{j+1}$	
	$pos_i \circ pos_{i+1} \circ pos_{j-1} \circ pos_j$	$pos_{i-1} \circ pos_i \circ pos_j \circ pos_{j+1}$	

Table 1: Feature templates for dependency parsing. For edge $d_{e_{ij}}$: $word_i$ is the parent word and $word_j$ is the child word, similar to ”pos”. ”+1” denotes the preceding token of the sentence, similar to ”-1”.

for dependency projection and annotation, which requires a small set of dependency annotated corpus of target language.

Similarly, using indirect information from multilingual (Cohen et al., 2011; Täckström et al., 2012) is an effective way to improve unsupervised parsing. (Zeman and Resnik, 2008; McDonald et al., 2011; Søgaard, 2011) employ non-lexicalized parser trained on other languages to process a target language. McDonald et al. (2011) adapts their multi-source parser according to DCA, while Naseem et al. (2012) selects a selective sharing model to make better use of grammar information in multi-sources.

Due to similar reasons, many works are devoted to POS projection (Yarowsky et al., 2001; Shen et al., 2007; Naseem et al., 2009), and they also suffer from similar problems. Some seek for unsupervised methods, e.g. Naseem et al. (2009), and some further improve the projection by a graph-based projection (Das and Petrov, 2011).

Our model differs from the approaches above in its emphasis on utilizing information from both sides of bilingual corpus in an unsupervised training framework, while most of the work above only utilize the information from a single side.

6 Experiments

In this section, we evaluate the performance of the MST dependency parser (McDonald et al., 2005b) which is trained by our bilingually-guided model on 5 languages. And the features used in our experiments are summarized in Table 1.

6.1 Experiment Setup

Datasets and Evaluation Our experiments are run on five different languages: Chinese(ch), Danish(da), Dutch(nl), Portuguese(pt) and Swedish(sv) (da, nl, pt and sv are free data sets distributed for the 2006 CoNLL Shared Tasks (Buchholz and Marsi, 2006)). For all languages, we only use English-target parallel data: we take the FBIS English-Chinese bitext as bilingual corpus for English-Chinese dependency projection which contains 239K sentence pairs with about 8.9M/6.9M words in English/Chinese, and for other languages we use the readily available data in the Europarl corpus. Then we run tests on the Penn Chinese Treebank (CTB) and CoNLL-X test sets.

English sentences are tagged by the implementations of the POS tagger of Collins (2002), which is trained on WSJ. The source sentences are then parsed by an implementation of 2nd-ordered MST model of McDonald and Pereira (2006), which is trained on dependency trees extracted from Penn Treebank.

As the evaluation metric, we use parsing accuracy which is the percentage of the words which have found their correct parents. We evaluate on sentences with all length for our method.

Training Regime In experiments, we use the projection method proposed by Jiang and Liu (2010) to provide the projection instances. And we train the projection part $\alpha = 0$ first for initialization, on which the whole model will be trained. Availing of the initialization method, the model can converge very fast (about 3 iterations is sufficient) and the results are more stable than the ones trained on random initialization.

Baselines We compare our method against three kinds of different approaches: unsupervised method (Klein and Manning, 2004); single-source direct projection methods (Hwa et al., 2005; Jiang and Liu, 2010); multi-source indirect projection methods with multi-sources (M-

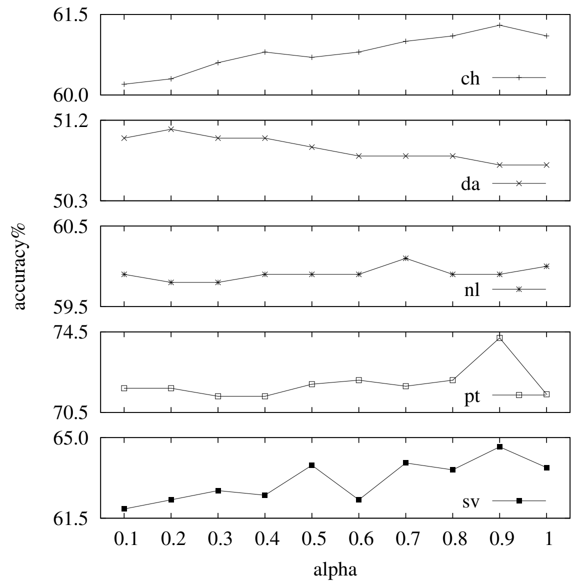

Figure 3: The performance of our model with respect to a series of ratio α

cDonald et al., 2011; Naseem et al., 2012).

6.2 Results

We test our method on CTB and CoNLL-X free test data sets respectively, and the performance is summarized in Table 2. Figure 3 presents the performance with different α on different languages.

Compare against Unsupervised Baseline Experimental results show that our unsupervised framework's performance approaches to the DMV method. And the bilingually-guided model can promote the unsupervised method consistency over all languages. On the best results' average of four comparable languages (da, nl, pt, sv), the promotion gained by our model is 28.5% over the baseline method (DMV) (Klein and Manning, 2004).

Compare against Projection Baselines For all languages, the model consistently outperforms on direct projection baseline. On the average of each language's best result, our model outperforms all kinds of baselines, yielding 3.0% gain over the single-source direct-projection method (Jiang and Liu, 2010) and 3.9% gain over the multi-source indirect-projection method (McDonald et al., 2011). On the average of all results with different parameters, our method also gains more than 2.0% improvements on all baselines. Particularly, our model achieves the most significant gains on Chinese, where the improvements are 4.5%/12.0% on direct/indirect projection baselines.

	Accuracy%					
Model	**ch**	**da**	**nl**	**pt**	**sv**	**avg**
DMV	42.5*	33.4	38.5	20.1	44.0	—.–
DPA	53.9	—.–	—.–	—.–	—.–	—.–
WPC	56.8	50.1	58.4	70.5	60.8	59.3
Transfer	49.3	49.5	53.9	75.8	63.6	58.4
Selective	51.2	—.–	55.9	73.5	61.5	—.–
$unsuper$	22.6	41.6	15.2	45.7	42.4	33.5
avg	61.0	50.7	59.9	72.0	63.1	61.3
max	61.3	51.1	60.1	74.2	64.6	62.3

Table 2: The directed dependency accuracy with different parameter of our model and the baselines. The first section of the table (row 3-7) shows the results of the baselines: a unsupervised method baseline (Klein and Manning, 2004)(DMV); a single-source projection method baseline (Hwa et al., 2005) (DPA) and its improvement (Jiang and Liu, 2010)(WPC); two multi-source baselines (McDonald et al., 2011)(Transfer) and (Naseem et al., 2012)(Selective). The second section of the table (row 8) presents the result of our unsupervised framework (unsuper). The third section gives the mean value (avg) and maximum value (max) of our model with different α in Figure 3.
*: The result is based on sentences with 10 words or less after the removal of punctuation, it is an incomparable result.

The results in Figure 3 prove that our unsupervised framework $\alpha = 1$ can promote the grammar induction if it has a good start (well initialization), and it will be better once we incorporate the information from the projection side ($\alpha = 0.9$). And the maximum points are not in $\alpha = 1$, which implies that projection information is still available for the unsupervised framework even if we employ the projection model as the initialization. So we suggest that a greater parameter is a better choice for our model. And there are some random factors in our model which make performance curves with more fluctuation. And there is just a little improvement shown in da, in which the same situation is observed by (McDonald et al., 2011).

6.3 Effects of the Size of Training Corpus

To investigate how the size of the training corpus influences the result, we train the model on extracted bilingual corpus with varying sizes: 10K, 50K, 100K, 150K and 200K sentences pairs.

As shown in Figure 4, our approach continu-

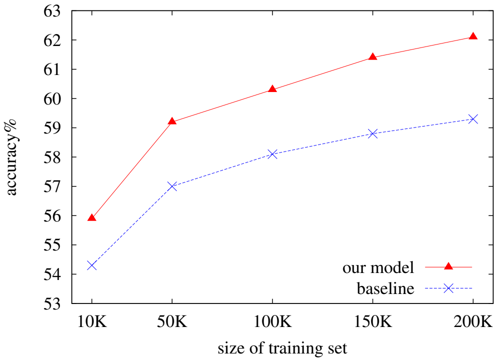

Figure 4: Performance on varying sizes (average of 5 languages, $\alpha = 0.9$)

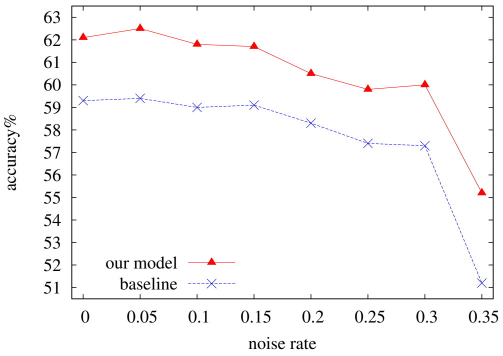

Figure 5: Performance on different projection quality (average of 5 languages, $\alpha = 0.9$). The noise rate is the percentage of the projected instances being messed up.

ously outperforms the baseline with the increasing size of training corpus. It is especially noteworthy that the more training data is utilized the more superiority our model enjoys. That is, because our method not only utilizes the projection information but also avails itself of the monolingual corpus.

6.4 Effect of Projection Quality

The projection quality can be influenced by the quality of the source parsing, alignments, projection methods, corpus quality and many other factors. In order to detect the effects of varying projection qualities on our approach, we simulate the complex projection procedure by messing up the projected instances randomly with different noise rates. The curves in Figure 5 show the performance of WPC baseline and our bilingual-guided method. For different noise rates, our model's results consistently outperform the baselines. When the noise rate is greater than 0.2, our improvement increases with the growth of the noise rate. The result suggests that our method can solve some problems which are caused by projection noise.

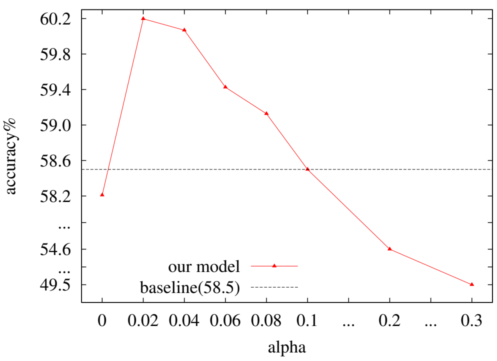

Figure 6: The performance curve of our model (random initialization) on Chinese, with respect to a series of ratio α. The baseline is the result of WPC model.

6.5 Performance on Random Initialization

We test our model with random initialization on different α. The curve in Figure 6 shows the performance of our model on Chinese.

The results seem supporting our unsupervised optimization method when α is in the range of $(0, 0.1)$. It implies that the unsupervised structure information is useful, but it seems creating a negative effect on the model when α is greater than 0.1. Because the unsupervised part can gain constraints from the projection part. But with the increase of α, the strength of constraint dwindles, and the unsupervised part will gradually lose control. And bad unsupervised part pulls the full model down.

7 Conclusion and Future Work

This paper presents a bilingually-guided strategy for automatic dependency grammar induction, which adopts an unsupervised skeleton and leverages the bilingually-projected dependency information during optimization. By simultaneously maximizing the monolingual likelihood and bilingually-projected likelihood in the EM procedure, it effectively integrates the advantages of bilingual projection and unsupervised induction. Experiments on 5 languages show that the novel strategy significantly outperforms previous unsupervised or bilingually-projected models. Since its computational complexity approaches to the skeleton unsupervised model (with much fewer iterations), and the bilingual text aligned to resource-rich languages is easy to obtain, such a hybrid method seems to be a better choice for automatic grammar induction. It also indicates that the combination of bilingual constraint and unsupervised methodology has a promising prospect for grammar induction. In the future work we will investigate such kind of strategies, such as bilingually unsupervised induction.

Acknowledgments

The authors were supported by National Natural Science Foundation of China, Contracts 61202216, 863 State Key Project (No. 2011AA01A207), and National Key Technology R&D Program (No. 2012BAH39B03), Key Project of Knowledge Innovation Program of Chinese Academy of Sciences (No. KGZD-EW-501). Qun Liu's work is partially supported by Science Foundation Ireland (Grant No.07/CE/I1142) as part of the CNGL at Dublin City University. We would like to thank the anonymous reviewers for their insightful comments and those who helped to modify the paper.

References

H. Alshawi. 1996. Head automata for speech translation. In *Proc. of ICSLP*.

James K Baker. 1979. Trainable grammars for speech recognition. *The Journal of the Acoustical Society of America*, 65:S132.

T. Berg-Kirkpatrick, A. Bouchard-Côté, J. DeNero, and D. Klein. 2010. Painless unsupervised learning with features. In *HLT: NAACL*, pages 582–590.

Rens Bod. 2006. An all-subtrees approach to unsupervised parsing. In *Proc. of the 21st ICCL and the 44th ACL*, pages 865–872.

S. Buchholz and E. Marsi. 2006. Conll-x shared task on multilingual dependency parsing. In *Proc. of the 2002 Conference on EMNLP*. Proc. CoNLL.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proc. of the 43rd ACL*, pages 173–180, Ann Arbor, Michigan, June.

W. Chen, J. Kazama, and K. Torisawa. 2010. Bitext dependency parsing with bilingual subtree constraints. In *Proc. of ACL*, pages 21–29.

S.B. Cohen, D. Das, and N.A. Smith. 2011. Unsupervised structure prediction with non-parallel multilingual guidance. In *Proc. of the Conference on EMNLP*, pages 50–61.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proc. of the 2002 Conference on EMNLP*, pages 1–8, July.

Michael Collins. 2003. Head-driven statistical models for natural language parsing. In *Computational Linguistics*.

D. Das and S. Petrov. 2011. Unsupervised part-of-speech tagging with bilingual graph-based projections. In *Proc. of ACL*.

K. Ganchev, J. Gillenwater, and B. Taskar. 2009. Dependency grammar induction via bitext projection constraints. In *Proc. of IJCNLP of the AFNLP: Volume 1-Volume 1*, pages 369–377.

R. Hwa, P. Resnik, A. Weinberg, and O. Kolak. 2002. Evaluating translational correspondence using annotation projection. In *Proc. of ACL*, pages 392–399.

R. Hwa, M. Osborne, A. Sarkar, and M. Steedman. 2003. Corrected co-training for statistical parsers. In *ICML-03 Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining, Washington DC*.

R. Hwa, P. Resnik, A. Weinberg, C. Cabezas, and O. Kolak. 2005. Bootstrapping parsers via syntactic projection across parallel texts. *Natural language engineering*, 11(3):311–325.

W. Jiang and Q. Liu. 2010. Dependency parsing and projection based on word-pair classification. In *Proc. of ACL*, pages 12–20.

D. Klein and C.D. Manning. 2004. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proc. of ACL*, page 478.

Terry Koo and Michael Collins. 2010. Efficient third-order dependency parsers. In *Proc. of the 48th ACL*, pages 1–11, July.

T. Koo, X. Carreras, and M. Collins. 2008. Simple semi-supervised dependency parsing. pages 595–603.

R. McDonald and F. Pereira. 2006. Online learning of approximate dependency parsing algorithms. In *Proc. of the 11th Conf. of EACL*.

R. McDonald, K. Crammer, and F. Pereira. 2005a. Online large-margin training of dependency parsers. In *Proc. of ACL*, pages 91–98.

R. McDonald, F. Pereira, K. Ribarov, and J. Hajič. 2005b. Non-projective dependency parsing using spanning tree algorithms. In *Proc. of EMNLP*, pages 523–530.

R. McDonald, K. Lerman, and F. Pereira. 2006. Multilingual dependency analysis with a two-stage discriminative parser. In *Proc. of CoNLL*, pages 216–220.

R. McDonald, S. Petrov, and K. Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In *Proc. of EMNLP*, pages 62–72. ACL.

T. Naseem, B. Snyder, J. Eisenstein, and R. Barzilay. 2009. Multilingual part-of-speech tagging: Two unsupervised approaches. *Journal of Artificial Intelligence Research*, 36(1):341–385.

Tahira Naseem, Regina Barzilay, and Amir Globerson. 2012. Selective sharing for multilingual dependency parsing. In *Proc. of the 50th ACL*, pages 629–637, July.

J. Nivre, J. Hall, J. Nilsson, G. Eryiğit, and S. Marinov. 2006. Labeled pseudo-projective dependency parsing with support vector machines. In *Proc. of CoNLL*, pages 221–225.

J. Nivre, J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kübler, S. Marinov, and E. Marsi. 2007. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(02):95–135.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proc. of the 21st ICCL & 44th ACL*, pages 433–440, July.

A. Sarkar. 2001. Applying co-training methods to statistical parsing. In *Proc. of NAACL*, pages 1–8.

L. Shen, G. Satta, and A. Joshi. 2007. Guided learning for bidirectional sequence classification. In *Annual Meeting-*, volume 45, page 760.

N.A. Smith and J. Eisner. 2005. Contrastive estimation: Training log-linear models on unlabeled data. In *Proc. of ACL*, pages 354–362.

D.A. Smith and J. Eisner. 2009. Parser adaptation and projection with quasi-synchronous grammar features. In *Proc. of EMNLP: Volume 2-Volume 2*, pages 822–831.

B. Snyder, T. Naseem, and R. Barzilay. 2009. Unsupervised multilingual grammar induction. In *Proc. of IJCNLP of the AFNLP: Volume 1-Volume 1*, pages 73–81.

Anders Søgaard. 2011. Data point selection for cross-language adaptation of dependency parsers. In *Proc. of the 49th ACL: HLT*, pages 682–686.

Valentin I. Spitkovsky, Hiyan Alshawi, and Daniel Jurafsky. 2010. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In *HLT: NAACL*, pages 751–759, June.

O. Täckström, R. McDonald, and J. Uszkoreit. 2012. Cross-lingual word clusters for direct transfer of linguistic structure.

William, M. Johnson, and D. McClosky. 2009. Improving unsupervised dependency parsing with richer contexts and smoothing. In *Proc. of NAACL*, pages 101–109.

D. Yarowsky, G. Ngai, and R. Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proc. of HLT*, pages 1–8.

Daniel Zeman and Philip Resnik. 2008. Cross-language parser adaptation between related languages. In *Proc. of the IJCNLP-08*. Proc. CoNLL.

Ciyou Zhu, Richard H Byrd, Peihuang Lu, and Jorge Nocedal. 1997. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software (TOMS)*, 23(4):550–560.