A Unified and Discriminative Soft Syntactic Constraint Model for Hierarchical Phrase-based Translation

Lemao Liu Tiejun Zhao Chao Wang Hailong Cao
School of Computer Science and Technology
Harbin Institute of Technology, Harbin, China
{lmliu,tjzhao,hailong,wangchao}@mtlab.hit.edu.cn

Abstract

In the last decade, there have been a countless number of researches in soft syntactic features many of which have led to the improved performance for Hiero. However, it seems that all the syntactic constituent features cannot efficiently work together in the Hiero optimized by MERT. In this paper, we propose a more general soft syntactic constraint model based on discriminative classifiers for each constituent type and integrate all of them into the translation model with a unified form. The experimental results show that our method significantly improves the performance on the NIST05 Chinese-to-English translation task.

1 Introduction

Hierarchical phrase-based translation model (Chiang, 2005) is a compromise of two popular translation models: syntax based model and phrase based model. It is a formal syntax grammar(Chiang, 2005; Wu, 1997), which does not take linguistic analysis into account when compared with other pure syntax systems (Liu et al., 2006;Yamada and Knight, 2001;Galley et al., 2006). It promises to improve the performance by adding syntax information to phrase based as well as formal syntax based translation models.

Chiang (2005) first introduced syntax knowledge into the hierarchical phrase based model. To make the model sensitive to the syntax structure, a constituent feature was integrated into the translation model with the soft constraint method. It was defined as follows: it gains 1 for rules whose source side respect syntactic phrase boundary in the parse tree, and 0 otherwise. However, it did not achieve statistically significant improvement in the experiment. Marton and Resnik (2008) thought that different syntactic types may play different roles in the translation model. However, (Chiang, 2005)'s method did not treat them discriminatively. They then defined soft constraint features for each constituent type based on the observation of this phenomenon. Their experiments showed that some constituent features significantly improved the performance, but others didn't. It is an interesting question whether all these constituent type models can work together efficiently. Although (Marton and Resnik 2008) did not give the experiments to support the positive answer, as presented previously, Chiang (2005) provided the evidence that their constituent models could not work together. (Chiang et al.,2008) thought one of its reasons is the limitation of MERT(Och,2003) with many features. We think there are two other reasons in addition to their suggestion. On the one hand, their models are heuristic, and they are not sensitive for other features such as boundary word information. However, in the previous work (Xiong et al., 2006), these features were shown to be helpful for the translation model. On the other hand, uniform combination of all the constituent models may cause a model bias, since some constituent types happen more often than others.

In this paper, we will address the question above. First, a discriminative soft constraint model is proposed for each syntactic constituent type. The model can be integrated with much context information. We consider several classifiers with different accuracy to construct soft constraint models, and our aim is to study the effect of the accuracy of the classifiers on the translation performance. Then, we investigate an efficient method to combine all the models to give a unified soft constraint model. Instead of uniformly combining all the models, we introduce a prior distribution for them and combine them with the priority.

The rest of the paper is organized as follows. Section 2 presents our baseline model. Section 3 gives the discriminative classifier based soft syntactic models, followed by section 4 in which the presentation of a unified soft syntactic model is outlined. Section 5 describes training of these models. Experiments and results are reported in section 6. We review some related work in section 7 and give our conclusion in section 8.

2 Hierarchical Phrase-Based Translation

Hierarchical phrase translation model is based on a Probability Synchronous Context-Free Grammar (PSCFG). Formally, a PSCFG is a 5-tuple $<\mathrm{N}, T_s, T_t, R, w>$, where N is the nonterminal set, T_s and T_t denote the terminal sets in source and target side respectively, R is the production rule set, and w is a weight function over R. In Hiero, the rule set R can be extracted from the bilingual corpus automatically.

Given a source sentence f and a PSCFG G, translation is represented as to search a target sentence $\hat{e}$ in the decoding space $\Delta(f, G)$ such that

$$\hat{e} = \mathrm{argmax}_{e \in \Delta(f,G)} p(e|f) (1),$$

where $\Delta(f, G)$ consists of all possible translation space and it is determined by the grammar G, and $p(e|f)$ is a translation model. Basically, $p(e|f)$ is based on the rule probability distribution w and can be represented as follows:

$$p(e|f) = \sum_{d \in D(f,e)} p(d) = \sum_{d \in D(f,e)} \prod_{r \in d} w(r)\ (2).$$

$D(f, e)$ denotes the set of synchronous derivation trees with (f,e) as their leaves, and d is a derivation member in $D(f, e)$. In decoding step, it is intractable to find the extract solution as (1), since the number of elements in $D(f, e)$ is exponential. In fact, one can approximate the optimal solution via MAP:

$$\hat{e} = e\left(\mathrm{argmax}_{d \in D(f)} p(d)\right) \quad (3)$$

$D(f)$ denotes the derivation set which can induce f in the source side and $e(d)$ denotes the target translation corresponding to derivation d. In hierarchical phrase translation, the rule probability can be represented as the log-linear (Och and Ney, 2002) combination of some feature functions:

$$w(r) = \prod_i \phi_i^{\lambda_i}(r) \quad (4),$$

$\forall i, \phi_i$ is a feature function, and λ_i is its feature weight and it can be optimized via MERT (Och, 2003) on the development set. The features (Koehn et al.,2003; Chiang, 2005) can be taken as following:

- the phrase translation probability $p(\alpha|\gamma), p(\gamma|\alpha)$;
- the lexical weights $p_w(\alpha|\gamma), p_w(\gamma|\alpha)$;
- a penalty for hierarchical rules;
- a penalty for glue rules;
- a word penalty;
- language model.

The decoding process is similar as the monolingual CKY parse and it can be considered as the transduction of source language into target language.

3 Discriminative Soft Syntactic Constraint Models

3.1 Soft Syntactic Constraints

For different syntactic categories (e.g. NP), Marton and Resnik (2008) defined some kinds of soft-constraint constituency features (e.g. NP=, NP+, NP_,etc.) for Hiero rules. For instance, if a synchronous rule $X \rightarrow < \alpha, \gamma >$ is used in a derivation, and the span of α is a cross constituent “NP+” in the source language parse tree, this rule will get an additional value λ_{NP+} to the model score for the case of “NP+”. In fact, each of these features can also be viewed as a discrete model with value {0, 1}, i.e. for the case of “NP=” if the span of α is exactly “NP”, the rule $X \rightarrow < \alpha, \gamma >$ gets a score 1 and 0 otherwise. These constituency features don’t discriminate the rules with the same span in the source language. In the next subsection, we will present more general SSC models which are sensitive to different rules and their context. We call these models *Soft Syntactic Constraint* models (SSC).These SSC models proposed by Marton and Resnik are heuristic, while our SSC models are much more general and based on discriminative classifiers.

In this paper, we further decompose the crossing constituent into 3 types to contain more syntactic information. For example, similarly as (Zollmann and Venugopal, 2006), the crossing constituent "NP+"are divided into L\NP, NP/R, and L\NP/R, which means a partial syntactic category VP missing some category to the left, the right and the left and right together, respectively. We call them *general constituent labels*(GCL). Figure 1 shows some examples for the GCLs.

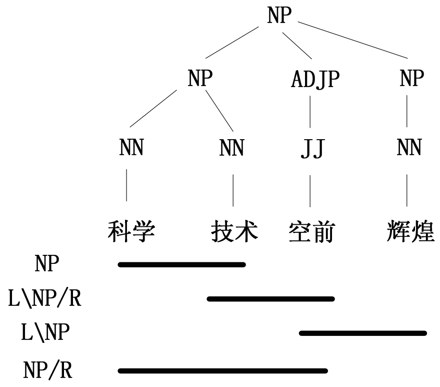


Figure 1: The general constituent labels for the spans.

3.2 Discriminative SSC models

Let r: $X \rightarrow < \alpha, \gamma >$ be a rule for the hierarchical phrase-based translation model, $\mathrm{GCL}(\alpha)$ the general constituent label for the span of α, and $\mathrm{C}(\alpha)$ the context of α in the source language f. For each $\mathrm{GCL}(\alpha)$, its SSC model is represented as a binary conditional probability$\mathrm{P}_{\mathrm{GCL}(\alpha)}(\mathrm{o}|\mathrm{C}(\alpha))$, whoseinherent meaning is the context-based influence of using the rule $X \rightarrow < \alpha, \gamma >$ to derive a derivation for f,under the $\mathrm{GCL}(\alpha)$.

It is a model selection problem to construct a proper model for $\mathrm{P}_{\mathrm{GCL}(\alpha)}(\mathrm{o}|\mathrm{C}(\alpha))$. In our case, the training scale is very huge (up to 10M examples). This means that the potential models should not be time-consuming during training. Previous work (Xiong et al., 2006; 2009) proved that MaxEnt achieves great performance in machine translation, so we will employ it in the paper. In order to evaluate the relationship between the accuracy of the SSC classifier and the performance of end to end translation, we need two other comparisons: one of them with lower accuracy and the other with higher accuracy than MaxEnt on the SSC classification task. Since with properly choosing features the logistic regression (LogReg) is less powerful than MaxEnt, we use LogReg for one classifier. Instead of trying to find a more powerful model on the classification task, we construct a model which is based on the combination of the MaxEnt and LogReg models. The first 2 SSC models are presented as following:

- MaxEnt based SSC model

$$\mathrm{P}_{\mathrm{GCL}(\alpha)}\big(\mathrm{o}\big|\mathrm{C}(\alpha)\big) = \frac{\exp\left(\sum_{\mathrm{i}} \lambda_{\mathrm{i}} \mathrm{f}_{\mathrm{i}}(\mathrm{o}, \mathrm{C}(\alpha))\right)}{\sum_{\mathrm{o}'} \exp\left(\sum_{\mathrm{i}} \lambda_{\mathrm{i}} \mathrm{f}_{\mathrm{i}}(\mathrm{o}', \mathrm{C}(\alpha))\right)} (5).$$

where$\mathrm{f}_{\mathrm{i}}\big(\mathrm{o}, \mathrm{C}(\alpha)\big)$ denotes a binary feature function, and λ_{i} its weight.

- LogReg based SSC model

$$\mathrm{P}_{\mathrm{GCL}(\alpha)}\big(\mathrm{o}\big|\mathrm{C}(\alpha)\big) = \begin{cases} \dfrac{\exp\left(\sum_{\mathrm{i}} \lambda_{\mathrm{i}} \mathrm{f}_{\mathrm{i}}(\mathrm{C}(\alpha))\right)}{1 + \exp\left(\sum_{\mathrm{i}} \lambda_{\mathrm{i}} \mathrm{f}_{\mathrm{i}}(\mathrm{C}(\alpha))\right)}, \text{if o} = 1, \\ \dfrac{1}{1 + \exp\left(\sum_{\mathrm{i}} \lambda_{\mathrm{i}} \mathrm{f}_{\mathrm{i}}(\mathrm{C}(\alpha))\right)}, \text{else.} \end{cases} (6).$$

where$\mathrm{f}_{\mathrm{i}}\big(\mathrm{C}(\alpha)\big)$ denotes a binary feature function[1], and λ_{i} its weight in above equations.

Among the many combination methods, the linear combination is simple and efficient. In order to reduce the number of parameters to be optimized, we combine the two component classifiers with interpolation weight as follows:

$$\mathrm{P}_{\mathrm{GCL}(\alpha)}\big(\mathrm{o}\big|\mathrm{C}(\alpha)\big) = \theta * \mathrm{P}^{\mathrm{ME}}{}_{\mathrm{GCL}(\alpha)}\big(\mathrm{o}\big|\mathrm{C}(\alpha)\big) + (1 - \theta) * \mathrm{P}^{\mathrm{LR}}{}_{\mathrm{GCL}(\alpha)}\big(\mathrm{o}\big|\mathrm{C}(\alpha)\big)(7).$$

[1] Please note that the features used in equation (5) are more than those in equation (6), so our MaxEnt based SSC models are more general than those based on LogReg.

where $\mathrm{P^{ME}}_{\mathrm{GCL}(\alpha)}$ and $\mathrm{P^{LR}}_{\mathrm{GCL}(\alpha)}$ are defined in equations (5) and (6).

We employ the toolkit implemented by Zhang (2004) to train each MaxEnt based SSC model and that implemented by Komarekand Moore (2005) to train each LogReg based model. The interpolation weight θ can be tuned on the development set.

4 A Unified Soft Syntactic Constraint Model

For different GCL, we have defined different SSC models hence giving us many models. Different models may have different contributions for translation: some of them have significant improvements such as VP+ and NP+, while others don't (Marton and Resnik, 2008). Instead of running experiments for each model, we define a unified SSC model to combine all the SSM models.

There are many methods to integrate all the SSC models into the translation model. For example, for each rule r, one can represent them as a unified feature with the following formula:

$$\phi_{\mathrm{SSC}}(\mathrm{r}) = \exp\left(\mathrm{P}_{\mathrm{GCL(r)}}(\mathrm{o} = 1|\mathrm{C}(\mathrm{source}(\mathrm{r}))\right) \quad (8).$$

where$\mathrm{C}(\mathrm{source}(\mathrm{r}))$ the context of the source side for r.Then one can easily add the feature into the equation (4). Suppose d denotes a translation derivation and $\log \mathrm{P}_{\mathrm{SSC}}(\mathrm{d})$ the log SSC score of$\mathrm{d}$,then $\log \mathrm{P}_{\mathrm{SSC}}(\mathrm{d})$ is described by the following equation:

$$\log \mathrm{P}_{\mathrm{SSC}}(\mathrm{d}) = \sum_{\mathrm{r}\in\mathrm{d}} \mathrm{P}_{\mathrm{GCL(r)}}(\mathrm{o} = 1|\mathrm{C}(\mathrm{source}(\mathrm{r})) \ (9)$$

We can see that it treats all GCL uniformly, and we call this representation uniform combination with SSC models. For instance, Turian and Melamed (2006) combine uniformly their models according to general syntactic labels and so do He et al. (2008) when integrating the rule selection models with respect to rules.

Observing that some GCLs are much more frequent than others, we consider a prior distribution of all GCLs. We define the following feature:

$$\phi_{\mathrm{SSC}}(\mathrm{r}) = \exp\left(\mathrm{P}\left(\mathrm{GCL}(\mathrm{r})\right) * \mathrm{P}_{\mathrm{GCL(r)}}(\mathrm{o} = 1|\mathrm{C}(\mathrm{source}(\mathrm{r})\right)(10),$$

where $\mathrm{P}\left(\mathrm{GCL}(\mathrm{r})\right)$ is a prior probability for GCLs and it isestimated by M.L.E. in the training examples. The $\log \mathrm{P}_{\mathrm{SSC}}(\mathrm{d})$ can be described as follows:

$$\log \mathrm{P}_{\mathrm{SSC}}(\mathrm{d}) = \sum_{\mathrm{r}\in\mathrm{d}} \mathrm{P}\left(\mathrm{GCL}(\mathrm{r})\right) * \mathrm{P}_{\mathrm{GCL(r)}}(\mathrm{o} = 1|\mathrm{C}\left(\mathrm{source}(\mathrm{r})\right)(11).$$

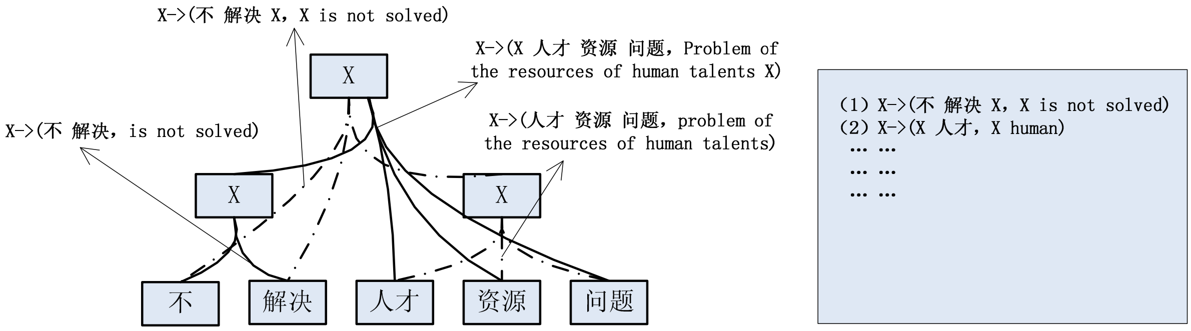
It is a Bayes-style combination.

5 Training SSCModels

5.1 Training instances

Unlike training models for ordinary classification tasks, our training instances are not available obviously because they are latent and by-product for machine translation. Cui et al. (2010) presented an efficient method to acquire training instances for rule selection model. Different from their method, ours is based on the derivations of the source sentence. Bilingual parsing (Wu, 1997) is a very efficient way to get the latent derivation for source language. In order to speed up the bilingual parsing, we limit the phrase table for each source sentence in the training data. When running bilingual parser for each source sentence, we just use the rules extracted from it and its reference during rule extraction step.

Although our method will prune some derivations which can derive the reference, it can still get derivations for more than 70% source sentences. Even if we can't get derivations for some source sentences, we can still extract the training examples for their partial derivations. Our method to extract instances is similar as that of (Turian et al., 2006), except that ours extracts from a derivation forest rather than a derivation tree. Due to the space limitation, the details are skipped. Figure 1 shows an example of some instances for training SSC models. In the rule table Figure 2(b), the rule (1) provides a positive instance because it is included in the derivations Figure 2(a), but the rule (2) provides a negative instance.


X->(不 解决 X，X is not solved)
X->(X 人才 资源 问题，Problem of the resources of human talents X)
X->(不 解决，is not solved)
X->(人才 资源 问题，problem of the resources of human talents)
X
X
X
不
解决
人才
资源
问题
(1) X->(不 解决 X，X is not solved)
(2) X->(X 人才，X human)
… …
… …
… …

(a)The derivation forest containing 2 trees

(b)The rule table

Figure 2.An example of training instances for SSC models.

5.2 Features

For a training instance corresponding to a rule $X \rightarrow < \alpha, \gamma >$, inspired by previous work (Zollmann and Venugopal, 2006; He et al., 2008; Cui et al., 2010), we design the following features to train our SSC models;

- **Syntactic features**, which are the general constituent labels defined in section 2.1 for the spans of r and the nonterminal symbols in the source side.
- **Parts-of-speech (POS) features**, which are the POS of the words immediately to the left and right of α and those of the boundary words covered by the nonterminal symbols in the source side.
- **Length features**, which are the length of sub-phrases covered by the nonterminal symbols in the source side.

In fact, our models can be extended to include other features, especially those in the target side. In order to compare our models with the work of Marton and Resnik (2008), we merely introduce several features.

6 Experiments and Results

We implement a hierarchical phrase-based system as our baseline, similar to Hiero (Chiang, 2005), and use XP+ (Marton and Resnik, 2008) as our comparison system. The features for the baseline are mentioned in section 1. We use the default setting as Hiero. When integrate our unified model into the translation model and then optimize its weight as the method in XP+, i.e. by MERT. We conduct our experiments on the Chinese-to-English translation task. The training data comes from FBIS corpus consisting of about 190k sentence pairs. The development set is NIST02 evaluation data and the test set is NIST05 evaluation data.

GCL	Size	GCL	Size
IP	0.09M	L\IP/R	2.98M
VP	0.59M	L\VP	0.36M
NP	0.95M	L\VP/R	2.31M
L\IP	1.15M	NP/R	0.62M
IP/R	0.94M	L\NP/R	1.06M

Table 1. The distribution of the training examples for partial general constituent labels.

We run GIZA++ (Och and Ney, 2000) on the training corpus in both directions (Koehn et al., 2003) to obtain the word alignment for each sentence pair. Then, we employ Stanford parser (Klein and Manning, 2003) to generate the parse tree for the source side of the data. We acquire about 15.85M training examples among which are 6.81M positive and 9.04M negative examples respectively. There are 88general constituent labels in all. Table 1 shows the distribution of the number of training examples for partial GCL labels. We employ the open toolkits of MaxEnt and LogReg to train SSC models for each GCL, and construct a linear combination model with them, where the interpolation weight is set to 0.86. We represent the translation systems based on LogReg, MaxEnt and their combination SSC models as the notations LogReg, MaxEnt and combination respectively. We train a 4-gram language model on the Xinhua portion of the English Gigaword corpus using the

SRILM Toolkits (Stolcke, 2002) with modified Kneser-Ney smoothing (Chen and Goodman,1998). In our experiments the translation performances are measured by case-sensitive BLEU4 metric (Papineni et al., 2002) and the statistical significance in BLEU score differences is tested by paired bootstrap re-sampling (Koehn, 2004).

Systems	BLEU-4
Baseline	27.45
XP+	27.85*
LogReg	27.67
MaxEnt	28.35*
Combination	28.48*

Table 2. The comparison results of our methods with p<0.005 on MT NIST05 test set.

As shown in Table 2, although LogReg does't improve significantly, both the systems MaxEnt and Combination significantly outperforms the baseline with more than 0.9 bleu scores and Combination achieves the best performance, which reflects that the discriminative SSC model can improve the performance over the baseline. Moreover, though LogReg is comparable to XP+, MaxEnt and Combination obtain improvements with 0.5 and 0.63 bleu scores, which indicate that the discriminative models are more superior to heuristic model when modeling the sub-task in translation.

Classifiers	Accuracy
LogReg	75%
MaxEnt	83%
Combination	88%

Table 3. The accuracy of classifiers on training examples.

Table 3 reports the accuracy of 3 discriminative classifiers in the SSC classification task. Combination is superior to both MaxEnt and LogReg, and followed by MaxEnt. We can see the similar rank of the translation results reported in Table 2. That empirically gives us the evidence that if one wants to achieve better translation performance, he/she needs try and construct a much superior sub-model.

In order to explain the effectiveness of our strategy to unify all SSC models into the decoder, we compare the two different methods mentioned in section 4.2 with MaxEnt SSC models. As reported in Table 4, there is a method to make the SSC models for all GCLs work together efficiently. Compared to the baseline, the method of uniformly combining SSC models doesn't obtain significant improvement, which is consistent with the result of the soft syntactic feature in (Chiang, 2005). However, the unified method with priority is much better than the uniform one. We believe that the most possible reason is that there is a model bias problem with the uniform combination of the SSC models. For example, some low frequent GCLs may have high SSC model score, which makes the translation model prone to choose more rules covered by those GCL. With the help of prior probability of GCLs, the model bias will be eliminated.

Methods	BLEU-4
Uniformly	27.70
With priority	28.35*

Table 4. The translation effect of unifying methods for MaxEnt based SSC models on MT NIST05 test set.

7 Related Work

There has been much effort to improve performance for hierarchical phrase-based machine translation by employing linguistic knowledge. Some of the work which is closely related with ours is reviewed in this section.

As presented previously, our work generalizes heavily from (Marton and Resnik, 2008). Besides exploring the soft syntactic constraints on hierarchical phrase model, ours investigates a way to make all the SSC models work together efficiently. (Stein et al., 2010) focuses on the syntactic constraint not only via the constituent parse but also via the dependency parse tree of source or target sentence. (Chiang et al., 2009; Chiang, 2010) similarly define many syntactic features including both source and target sides but integrated them into translation model by MIRA algorithm to optimize their weights. Their work proposes the heuristic syntactic features, while ours employ the discriminative syntactic models.

Zollmann and Venugopal (2006) use a constituent parse tree of target to provide constraints on the synchronous rules. They refine the translation grammar with the syntactic constituent types, while ours integrates syntactic knowledge as a sub-model. The idea to design the labels of our SSC models is bought from their work.

Huang et al. (2010) decorate the syntax structure into the non-terminal in hierarchical rules as a feature vector. During decoding time, they calculate the similarity between the syntax of the source side and the rules used to derive translations, and then they add the similarity measure to translation model as an additional feature. Their work differs from ours in that they don’t directly use the syntax knowledge to calculate the additional feature score, but use it to derive a latent syntactic distribution.

He et al.(2008) and Cui et al.(2010) employ the syntax knowledge as some of features to construct rule selection models. Our approach differs in two ways. First, their models are dominated by the rules, while ours are implemented by our syntactic labels. Secondly, when training discriminative models their training examples are derived from the rule extraction while ours are from the formal bilingual parsing derivation forest of the training data. Despite these differences, their strong results reinforce our claim that discriminative models are useful to build the sub-model in translation.

8 Conclusion and Future Work

In this paper we proposed a unified SSC model based on discriminative classifiers for hierarchical phrase-based translation. Experimental results prove the effectiveness of our method on the NIST05 Chinese-to-English translation task.

There are three contributions in this paper. Firstly, it shows that the discriminative soft syntactic constraint model achieves better result over the heuristic model as (Marton and Resnik, 2008). Secondly, it empirically proves that the more accurate classifier can gain better results when building a sub-model for the translation model. The third and final contribution is that our model proposes an efficient method which integrates all models with respect to general constituent labels into hierarchical phrase translation model and improves its performance.

In the future work we will investigate some strategies to selection examples for training classifiers, so as to prove our results on a much larger training data set.

Acknowledgments

This work is funded by National Natural Science Foundation of China (grant no. 60736014), National High Technology Research and Development Program of China (863 Program) (grant no. 2006AA010108), and Microsoft Research Asia IFP (grant no. FY09-RES-THEME-158)

References

Stanley F. Chen and Joshua Goodman. 1998. An empiricalstudy of smoothing techniques for language modeling.Harvard University *Technical Report TR-10-98*.

David Chiang. 2005. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In *Proc. ACL05.*

David Chiang, Kevin Knight, and Wei Wang. 2009. 11,001 new features for statistical machine translation. In *Proc. HLT- NAACL09.*

David Chiang. 2010. Learning to translate with source and target syntax. In *Proc. ACL10.*

Lei Cui, Dongdong Zhang, Mu Li, Ming Zhou, and Tiejun Zhao. 2010. A Joint Rule Selection Model for Hierarchical Phrase-Based Translation. In *Proc. of ACL10.*

Michel Galley, Jonathan Graehl, Kevin Knight, DanielMarcu, Steve DeNeefe, Wei Wang, and IgnacioThayer. 2006. Scalable Inference and Training ofContext-Rich Syntactic Models. In *Proc. of ACL-COLING06*.

Zhongjun He, Qun Liu and Shouxun Lin. 2008.Improving Statistical Machine Translation using Lexicalized Rule Selection. In *Proc. of COLING08.*

Zhongqiang Huang, Martin Cmejrek and Bowen Zhou. 2010. Soft Syntactic Constraint for Hierarchical Phrase-Based Translation Using Latent Syntactic Distributions. In *Proc. of EMNLP10.*

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003.Statistical Phrase-Based Translation. In *Proc. of HLT-NAACL03.*

Paul Komarek andAndrew Moore. 2005. Making Logistic Regression a Core Data Mining Tool With TR-IRLS.In *Proc. of ICDM05.*

Philipp Koehn. 2004. Statistical Significance Tests forMachine Translation Evaluation. In *Proc. of EMNLP04.*

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-String Alignment Template for Statistical Machine Translation.In *Proc. of ACL-COLING06.*

Yuval Marton and Philip Resnik. 2008. Soft Syntactic Constraints for Hierarchical Phrased-Based Translation. In *Proc. of ACL08.*

Franz Josef Och and Hermann Ney. 2000. Improved Statistical Alignment Models. In *Proc. of ACL00.*

Franz Josef Och and H. Ney. Discriminative training and maximum entropy models for statistical machine translation.In *Proc. of ACL*. 2002.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proc. of ACL03.*

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for AutomaticEvaluation of Machine Translation. In *Proc. of ACL02*.

D. Stein, S. Peitz, D. Vilar, and H. Ney. 2010. A Cocktail of Deep Syntactic Features for Hierarchical Machine Translation. In *Conf. of AMTA10.*

Andreas Stolcke. 2002. SRILM - An Extensible Language Modeling Toolkit.In *Proc. of ICSLP02.*

Joseph Turian and I. Dan Melamed.2006. Advances in Discriminative Parsing.In *Proc. of ACL-COLING06.*

Dekai Wu. 1997. Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora. *Computational Lingustics,23(3).*

Deyi Xiong, Qun Liu, and Shouxun Lin. 2006. Maximum Entropy Based Phrase Reordering Model forStatistical Machine Translation. In *Proc. of ACL-COLING06*.

Deyi Xiong, Min Zhang, Aiti Aw, and Haizhou Li.2009.A Syntax-Driven Bracketing Model for Phrase-Based Translation.In *Proc. of ACL09.*

Kenji Yamada and Kevin Knight. 2001. A Syntax-Based Statistical Translation Model. In *Proc. of ACL01*.

Andreas Zollmann and AshishVenugopal. 2006. Syntax Augmented Machine Translation via Chart Parsing. In *Proc. of HLT-NAACL06 Workshop.*

Le Zhang. 2006. Maximum Entropy Modeling Toolkit for Python and C++. Available at http://homepages.inf.ed.ac.uk/lzhang10/maxent_toolkit.html.