Utilizing Target-Side Semantic Role Labels to Assist Hierarchical Phrase-based Machine Translation

Qin Gao and **Stephan Vogel**
Language Technologies Institute, Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
{qing, stephan.vogel}@cs.cmu.edu

Abstract

In this paper we present a novel approach of utilizing Semantic Role Labeling (SRL) information to improve Hierarchical Phrase-based Machine Translation. We propose an algorithm to extract SRL-aware Synchronous Context-Free Grammar (SCFG) rules. Conventional Hiero-style SCFG rules will also be extracted in the same framework. Special conversion rules are applied to ensure that when SRL-aware SCFG rules are used in derivation, the decoder only generates hypotheses with complete semantic structures. We perform machine translation experiments using 9 different Chinese-English test-sets. Our approach achieved an average BLEU score improvement of 0.49 as well as 1.21 point reduction in TER.

1 Introduction

Syntax-based Machine Translation methods have achieved comparable performance to Phrase-based systems. Hierarchical Phrase-based Machine Translation, proposed by Chiang (Chiang, 2007), uses a general non-terminal label X but does not use linguistic information from the source or the target language. There have been efforts to include linguistic information into machine translation. Liu et al (2006) experimented with tree-to-string translation models that utilize source side parse trees, and later improved the method by using the Packed Forest data structure to reduce the impact of parsing errors (Liu and Huang, 2010). The string-to-tree (Galley et al, 2006) and tree-to-tree (Chiang, 2010) methods have also been the subject of experimentation, as well as other formalisms such as Dependency Trees (Shen et al., 2008).

One problem that arises by using full syntactic labels is that they require an exact match of the constituents in extracted phrases, so it faces the risk of losing coverage of the rules. SAMT (Zollmann and Venugopal, 2006) and Tree Sequence Alignment (Zhang et al., 2008) are proposed to amend this problem by allowing non-constituent phrases to be extracted. The reported results show that while utilizing linguistic information helps, the *coverage* is more important (Chiang, 2010). When dealing with formalisms such as semantic role labeling, the coverage problem is also critical. In this paper we follow Chiang's observation and use SRL labels to augment the extraction of SCFG rules. I.e., the formalism provides additional information and more rules instead of restrictions that remove existing rules. This preserves the coverage of rules.

Recently there has been increased attention to use semantic information in machine translation. Liu and Gildea (2008; 2010) proposed using Semantic Role Labels (SRL) in their tree-to-string machine translation system and demonstrated improvement over conventional tree-to-string methods. Wu and Fung (2009) developed a framework to reorder the output using information from both the source and the target SRL labels. In this paper, we explore an approach of using the target side SRL information in addition to a Hierarchical Phrase-based Machine Translation framework. The proposed method extracts initial phrases with two different heuristics: The first heuristic is used to extract rules that have a general left-hand-side (LHS) non-terminal tag X,

Proceedings of SSST-5, Fifth Workshop on Syntax, Semantics and Structure in Statistical Translation, pages 107–115,
ACL HLT 2011, Portland, Oregon, USA, June 2011.

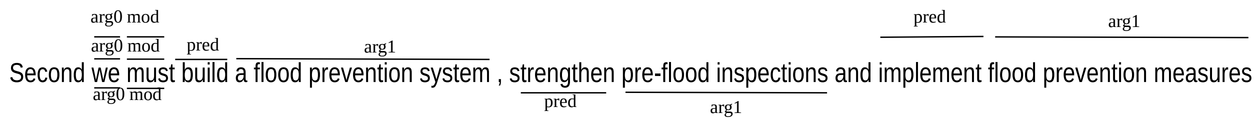


Figure 1: Example of predicate-argument structure in a sentence

i.e., Hiero rules. The second will extract phrases that contain information of SRL structures. The predicate and arguments that the phrase covers will be represented in the LHS non-terminal tags. After that, we obtain rules from the initial phrases in the same way as the Hiero extraction algorithm, which replaces nesting phrases with their corresponding non-terminals.

By applying this scheme, we will obtain rules that contain SRL information, without sacrificing the coverage of rules. In this paper, we call such rules SRL-aware SCFG rules. During decoding, both the conventional Hiero-style SCFG rules with general tag X and SRL-aware SCFG rules are used in a synchronous Chart Parsing algorithm. Special conversion rules are introduced to ensure that whenever SRL-aware SCFG rules are used in the derivation, a complete predicate-argument structure is built.

The main contributions are:

1. an algorithm to extract SRL-aware SCFG rules using target side SRL information.

2. an approach to use Hiero rules side-by-side with information-rich SRL-aware SCFG rules, which improves the quality of translation results.

In section 2 we briefly review SCFG-based machine translation and SRL. In section 3, we describe the SRL-aware SCFG rules. Section 4 provides the detail of the rule extraction algorithm. Section 5 presents two alternative methods how to utilize the SRL information. The experimental results are given in Section 6, followed by analysis and conclusion in Section 7.

2 Background

2.1 Hierarchical Phrase-based Machine Translation

Proposed by Chiang (2005), the Hierarchical Phrase-based Machine Translation model (commonly known as the Hiero model) has achieved results comparable, if not superior, to conventional Phrase-based approaches. The basic idea is to treat the translation as a synchronous parsing problem. Using the source side terminals as input, the decoder tries to build a parse tree and synchronously generate target side terminals. The rules that generates such synchronous parse trees are in the following form:

$$X \rightarrow (f_1\ X_1\ f_2\ X_2\ f_3, e_1\ X_2\ e_2\ X_1\ e_3)$$

where X_1 and X_2 are non-terminals, and the subscripts represents the correspondence between the non-terminals. In Chiang's Hiero model all non-terminals will have the same tag, i.e. X. The formalism, known as Synchronous Context-Free Grammar (SCFG) does not require the non-terminals to have a unique tag name. Instead, they may have tags with syntactic or semantic meanings, such as NP or VP.

2.2 Semantic Role Labeling and Machine Translation

The task of semantic role labeling is to label the semantic relationships between predicates and arguments. This relationship can be treated as a dependency structure called "Predicate-Argument Structure" (PA structure for short). Figure 1 depicts examples of multiple PA structures in a sentence. The lines indicate the span of the predicates and arguments of each PA structure, and the tags attached to these lines show their role labels.

Despite the similarity between PA structure and dependency trees, SRL offers a structure that posses better granularity. Instead of trying to analyze all links between words in the sentences, PA structure only deals with the relationships between verbs and constituents that are arguments of the predicates. This information is useful in preserving the meaning of the sentence during the translation process.

However, using semantic role representation in machine translation has its own set of problems.

First, we face the coverage problem. Some sentences might not have semantic structure at all, if, for instance they consist of single noun phrases or contain only rare predicates that are not covered by the semantic role labeler. Moreover, the PA structures are not guaranteed to cover the whole sentence. This is especially true when two or more predicates are presented in a coordinated structure. In this case, the arguments of other predicates will not be covered in the PA structure of the predicate.

The second problem is that the SRL labels are only on the constituents of predicate and arguments. There is no analysis conducted inside the augments. That is different from syntactic parsing or dependency parsing, which both provide a complete tree from the sentence to every individual word. As we can see in Figure 1, words such as “Second” and “and” are not covered. Inside the NPs such as “a flood prevention system”, SRL will not provide more information. Therefore it is hard to build a self-contained formalization based only on SRL labels. Most work on SRL labels is built upon or assisted by other formalisms. For instance, Liu and Gildea (2010) integrated SRL label into a tree-to-string translation system. Wu and Fung (2009) used SRL labels for reordering the n-best output of phrase-based translation systems. Similarly, in our work we also adopt the methodology of using SRL information to assist existing formalism. The difference of our method from Wu and Fung is that we embed the SRL information directly into the decode, instead of doing two-pass decoding. Also, our method is different from Liu and Gildea (2010) that we utilize target side SRL information instead of the source side.

As we will see in section 3, we define a mapping function from the SRL structures that a phrase covers to a non-terminal tag before extracting the SCFG rules. The tags will restrict the derivation of the target side parse tree to accept only SRL structures we have seen in the training corpus. The mapping from SRL structures to non-terminal tags can be defined according to the SRL annotation set.

In this paper we adopt the PropBank (Palmer et al., 2005) annotation set of semantic labels, because the annotation set is relatively simple and easy to parse. The small set of argument tags also makes the number of LHS non-terminal tags small, which alleviates the problem of data scarcity. However the methodology of this paper is not limited to PropBank tags. By defining appropriate mapping, it is also possible to use other annotation sets, such as FrameNet (Baker et al., 2002).

3 SRL-aware SCFG Rules

The SRL-aware SCFG rules are SCFG rules. They contain at least one non-terminal label with information about the PA structure that is covered by the non-terminal. The labels are called SRL-aware labels, and the non-terminal itself is called SRL-aware non-terminal. The non-terminal can be on the left hand side or right hand side or the rule, and we do not require all the non-terminals in the rules be SRL-aware, thus, the general tag X can also be used. In this paper, we assign SRL-aware labels based on the SRL structure they cover. The label contains the following components:

1. The predicate frame; that is the predicate whose predicate argument structure belongs to the SRL-aware non-terminal.

2. The set of complete arguments the SRL-aware non-terminal covers.

In practice, the predicates are stemmed. For example, if we have a target side phrase: *She beats eggs today*, where *She* will be labeled as *ARG0* of the predicate *beat*, and *eggs* will be labeled as *ARG1*, *today* will be labeled as *ARG-TMP*, respectively. The SRL-aware label that covers this phrase is:

$$\#beat/0\_1\_TMP$$

There are two notes for the definition. Firstly, the order of arguments is not important in the label. $\#beat/0\_1\_TMP$ is treated identically to $\#beat/0\_TMP\_1$. Secondly, as we always require the predicate to be represented, an SRL-aware non-terminal should always cover the predicate. This property will be re-emphasized when we discuss the rule extraction algorithm in Section 3. Figure 2 shows some examples of the SRL-aware SCFG rules.

When the RHS non-terminal is an SRL-aware non-terminal, we define the rule as a conversion rule. A conversion rule is only generated when the right

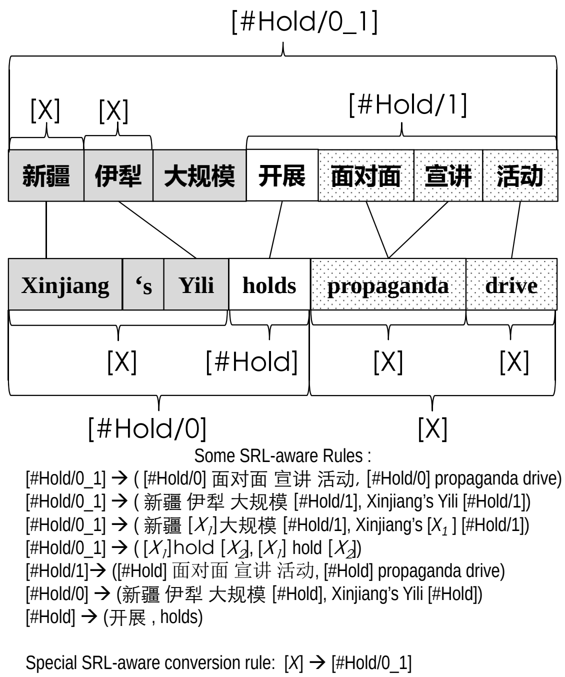
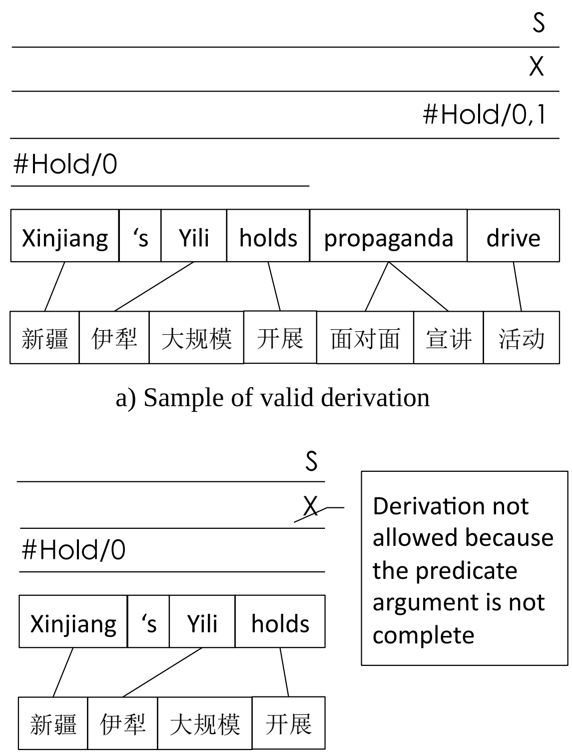


Figure 2: Example SRL structure with word alignment



a) Sample of valid derivation

b) Sample of invalid derivation

Figure 3: Example of a derivations of sentence

hand side is a complete SRL structure. For example, $\#hold/0$ is not a complete SRL structure in Figure 2, because it lacks of a required argument, while $\#hold/0\_1$ is a complete SRL structure. In this case, the conversion rule $X \rightarrow \#hold/0\_1$ will be extracted from the example shown in Figure 2, but not the other. Together with the *glue rules* that commonly used in Hiero decoder, i.e. $S \rightarrow (S\ X_1, S\ X_1)$ and $S \rightarrow (X_1, X_1)$, the conversion rules ensures that whenever SRL-aware SCFG rules are used in parsing, the output parse tree contains only complete SRL structures. This is because only complete SRL structures that we have observed in the training data can be converted back to the general tag X.

After we have extracted the SRL-aware SCFG rules, derivation can be done on the input of source sentence. For example, the sentence 新疆 大规模 开展 面对面 宣讲 活动 [1] can generate the parse tree and translation in Figure 3a) using the rules shown in Figure 2. Also, we can see in Figure 3b) that incomplete SRL structures cannot be generated due to the absence of a proper conversion rule.

[1]The translation is *Xinjiang's Yili holds propaganda drive* and the Pinyin transliteration is *Xinjiang daguimo kaizhan mianduimian xuanjiang huodong*

We can see from the example in Figure 3a), that the SRL-aware SCFG rules fit perfectly in the SCFG framework. Therefore no modification need to be made on a decoder, such as MosesChart decoder,for instance (Hoang and Koehn, 2008). The main problem is how to extract the SRL-aware SCFG rules from the corpus and estimate the feature values so that it works together with the conventional Hiero rules. In the next two sections we will present the rule extraction algorithm and two alternative methods for comparison.

4 Rule Extraction Algorithm

The Hiero rule extraction algorithm uses the following steps:

1. Extract the initial phrases with the commonly used alignment template heuristics. To reduce the number of phrases extracted, an additional restriction is applied that the boundary words must be aligned on both sides. Also, the maximum length of initial phrases is fixed, and usually set to 10.

2. If an initial phrase pair contains another phrase pair, then we can replace the embedded phrase pairs with non-terminal X. Restrictions also apply in this stage. Firstly the source side phrase can only contain two or less non-terminals. Secondly, two source side non-terminals must not be next to each other. And finally, after the substitution, at least one remaining terminal in the source side should have alignment links to the target side terminals.

It is easy to see this scheme is not able to handle the extraction of SRL-aware SCFG rules. The length of initial phrases is limited and it may not be able to cover a complete predicate-argument structure. In the meantime, the restrictions on unaligned words on the boundaries will cause a large number of SRL-aware SCFG rules to be excluded. Therefore, a modified algorithm is proposed to handle extraction of SRL-aware SCFG rules.

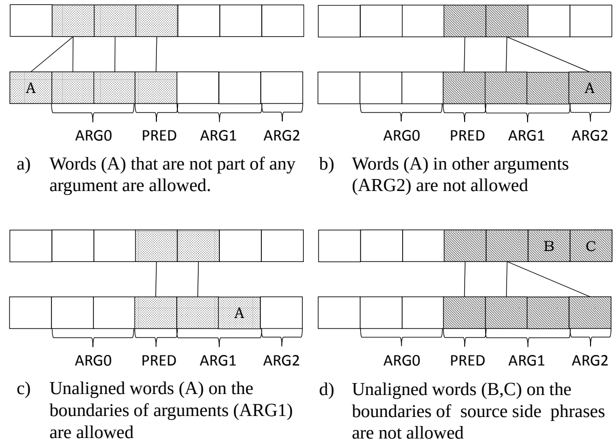
One sentence may have multiple verbs and, therefore, multiple PA structures. Different PA structures may be nested within each other. However we do not want to complicate the representation by attempting to build a tree structure from multiple structures. Instead, we treat them independently.

For each word-aligned sentence pair, if there is no PA structure given, we run the general Hiero extraction algorithm. Otherwise, for each PA structure, we apply the algorithm for SRL-aware rule extraction, which takes two steps, extracting the initial SRL-aware phrases and extracting the SRL-aware SCFG rules.

4.1 Extraction of Initial SRL-aware Phrases

First, a different heuristics is used to extract initial SRL-aware phrases. These phrases have the following properties:

1. On the target side, the phrase covers at least one complete constituent in the PA structure, which must include the predicate. The phrase pair can include words that are not part of any argument; however it cannot include partial arguments. In Figure 4b), the phrase pair is not included in the initial SRL-aware phrases because it includes a word A from argument $ARG2$. However, in Figure 4a), inclusion of the first target word A, which is not part of any argument, is allowed.



a) Words (A) that are not part of any argument are allowed.
b) Words (A) in other arguments (ARG2) are not allowed
c) Unaligned words (A) on the boundaries of arguments (ARG1) are allowed
d) Unaligned words (B,C) on the boundaries of source side phrases are not allowed

Figure 4: Demonstration of restrictions of whether or not a rule is included in initial SRL-aware phrases. The sub-figures a) and c) show two cases that unaligned words or words not in any arguments are allowed in extracted phrases and sub-figures b) and d) show two cases that the phrases are excluded from the phrase table. The shaded blocks indicate the range of candidate phrases.

2. At least one word pair between the source and the target side phrase is aligned, and no words in the source or the target side phrase align to words outside the phrase pair. These are the standard heuristics used in the hierarchical phrase extraction algorithm.

3. For the target side, unaligned words on the boundaries are allowed only if the word is found inside one of the arguments. On the source side, however, unaligned words are not allowed on the boundaries. The idea is demonstrated in Figure 4c) and 4d). In Figure 4c), the unaligned boundary word A is included in the target side phrase because it is part of an argument. In Figure 4d), unaligned words B and C are not allowed to be included in the proposed phrase.

Given a PA structure of the sentence, we applied following algorithm:

1. Extract all possible target side phrases that contain **the predicate** and **any number of arguments**.

2. For each of the extracted target side phrases T, find the minimum span of the source side phrase S that contains all the words aligned to

the target side phrase. This can be done by simply calculating the minimum and maximum index of the source side words aligned to the target side phrase.

3. Find the minimum span of target side phrase T_1 that are aligned to the source side phrase S. If the minimum span is already covered by the target side phrase extracted in the previous step, i.e. $T_1 = T$, we add the phrase pair (S, T) to the pool of initial phrases. If the newly obtained target side phrase is larger than the original one, then we need to decide whether the new span contains a word in another arguments. If so, then we do not add the phrase pair, return to step 2 and continue with the next target side phrase. Otherwise, we update $T := T_1$ and go back to step 2.

The readers may notice that although in several steps we need to determine whether there are links outside the phrase pairs, the information is easy to compute. We only need to keep track of the maximum and minimum indices of words that each source and target word aligns to. With the indices pre-computed, in the worst case scenario we only need to calculate M times for the maximum and minimum indices, where M is the total number of words in the source and the target side, before we can determine the validity of the largest target side SRL-aware phrase. The worst case complexity of the algorithm is $O(N * M)$, where N is the number of arguments in the segmentation. This is only a rough upper bound for the time complexity; the average case will be much better.

4.2 Extracting SRL-aware SCFG Rules

Before we generate rules from the extracted initial phrases, we first need to assign non-terminal labels to the initial SRL-aware phrases. We define a map from the SRL structures to non-terminal tags of SCFG rules. An SRL-aware non-terminal label is a combination of the predicate label and the argument labels. The predicate label is the stemmed predicate. We can eliminate the morphology to alleviate the problem of the data scarcity. In addition, the argument labels represent all the arguments that the current SRL-aware rule covers. The mapping is trivial given the initial SRL-aware phrase extraction algorithm, and it can be determined directly in the first step.

The initial phrases already are SCFG rules. To extract rules with non-terminals we will replace the sub-phrases with non-terminals if the sub-phrase is embedded in another phrase pair. The algorithm is similar to that described by Chiang (2005). However we apply new restrictions because we now have two sets of different initial phrases. If the outer rule is SRL-aware, we allow both sets of the initial phrases to be candidates of embedded phrases. However if the outer rule is X, we do not allow a replacement of SRL-aware SCFG rules within it. Therefore we will have rules where LHS non-terminals are SRL-aware, and some RHS non-terminals are X, but not vice versa. The reason for the restriction is to prevent the conversion of incomplete predicate-argument structures back to X. As we mentioned before, one of the design goals of our algorithm is to ensure that once SRL-aware SCFG rules are used in the derivation, a complete PA structure must be generated before it can be converted back. The only way of converting SRL-aware tags back to X is through special conversion rules, whose LHS is the X and the RHS is a complete SRL-aware tag. Extracting such conversion rules is trivial given the SRL labels.

The extracted rules are subject to filtering by the same restrictions as conventional Hiero rules. The filtering criteria include:

1. Two non-terminals on the source side should not be adjacent.

2. We allow up to two non-terminals on the RHS.

3. The source side rule contains no more than five tokens including terminals and non-terminals.

5 Decoder Integration

The extracted SCFG rules, both SRL-aware and X, will go through the feature estimation process to produce the rule table. Integrated with the conversion rules, most chart-based decoders such as MosesChart (Hoang and Koehn, 2008), cdec (Dyer et al, 2010) and Joshua (Li et al, 2009) can use these rules in decoding. We applied MosesChart for tuning and decoding.

While the SRL-aware SCFG rules are used to constrain the search space and derivation, we do not in-

		mt02	mt03	mt04	mt05	mt08	bl-nw	bl-wb	dv-nw	dv-wb	avg
Baseline	BLEU	29.56	27.02	30.28	26.80	21.16	21.96	20.10	24.26	20.13	n/a
	TER	68.87	70.19	67.18	70.60	69.93	64.44	64.74	63.21	66.61	n/a
	(T-B)/2	19.66	21.59	18.45	21.90	24.39	21.24	22.32	19.48	23.24	n/a
SRL	BLEU	+0.33	−0.50	+0.20	+0.47	−0.16	+1.24	+1.13	+0.39	+1.35	+0.49
	TER	−1.58	−1.77	−1.93	−1.68	−0.71	−0.29	−0.22	−1.36	−1.34	−1.21
	(T-B)/2	−0.95	−0.63	−1.07	−1.08	−0.28	−0.76	−0.68	−0.88	−1.35	−0.85

Table 1: Experiment results on Chinese-English translation tasks, bl-nw and bl-wb are newswire and weblog parts for DEV07-blind, dv-nw and dv-wb are newswire and weblog parts for DEV07-dev. We present the BLEU scores, TER scores and (TER-BLEU)/2.

troduce new features into the system. The features we used in the decoder are commonly used, including source and target rule translation probabilities, the lexical translation probabilities, and the language model probability. The feature values are calculated by MLE estimation.

Besides the expanded rule table and conversion rules, the decoder does not need to be modified. We incorporate MERT to tune the feature weights. The minimum modifications for the decoder make the proposed method an easy replacement for Hiero rule extractors.

6 Experiments and discussion

We performed experiments on Chinese to English translation tasks. The data set we used in the experiments is a subset of the FBIS corpus. We filter the corpus with maximum sentence length be 30. The corpus has 2.5 million words in Chinese side and 3.1 million on English side.

We adopted the ASSERT semantic role labeler (Pradhan et al., 2004) to label the English side sentences. The parallel sentences are aligned using MGIZA++ (Gao and Vogel, 2008) and then the proposed rule extraction algorithm was used in extracting the SRL-aware SCFG rules. We used the MosesChart decoder (Hoang and Koehn, 2008) and the Moses toolkit (Koehn et al, 2007) for tuning and decoding. The language model is a trigram language model trained on English GIGAWord corpus (V1-V3) using the SRILM toolkit.

We used the NIST MT06 test set for tuning, and experimented with an additional 9 test sets, including MT02, 03, 04, 05, 08, and GALE test sets DEV07-dev and DEV07-blind. DEV07-dev and DEV07-blind are further divided into newswire and weblog parts.

We experimented with the proposed method and the alternative methods presented in section 4, and the results of nine test sets are listed in Table 1. As we can observe from the results, the largest improvement we discovered from our proposed method is more than 1 BLEU point, and a significant drop is only observed on one test set, MT03, where we lose 0.5 BLEU points. Averaged across all the test sets, the improvement is 0.49 BLEU points on the small training set. When TER is also taken into account, all of the nine test sets showed consistent improvement. The (TER-BLEU)/2 score, which we used as the primary evaluation metric, improved by 0.85 across nine test sets.

As we expected, the coverage of SRL-aware SCFG rules is not as good as the Hiero rules. We analyzed the top-best derivation of the results. Only 1836 out of 7235 sentences in the test sets used SRL-aware SCFG rules. However, the BLEU scores on the 1836 sentences improved from 27.98 in the baseline system to 28.80, while the remaining 5399 sentences only improved from 30.13 to 30.22. The observation suggests the potential for further improvement if we can increase the coverage by using more data or by modifying the mapping from tags to the structures to make rules more general.

We display the hypothesis of a sentence in Figure 5 to demonstrate a concrete example of improvements obtained by using the method,. As this figure demonstrate, the SRL-aware SCFG rules enable the system to pick the correct structure and reordering for the verbs *trigger* and *enter*.

Given the results presented in the paper, the question arises as to whether it is prudent to integrate multiple formalisms or labeling systems, such as

Source	乌克兰 因 总统 选举 引发 的 混乱 进入 第三 周
SRLTag	Ukraine because of the chaos **triggered** by the presidential election has **entered** the third week
Baseline	Ukraine today because of the chaos **triggered in** the third week of the presidential election
References	The chaos **caused** by Ukraine's presidential election has **entered** its third week. The turmoil in Ukraine **triggered** by the presidential election **entered** the third week The chaos **sparked** off by the presidential election in Ukraine has **entered** its third week. Ukraine **heads** into a third week of turmoil **caused** by the presidential election

Figure 5: An example of improvement caused by better attachment of verbs and its arguments

syntactic parsing or SRL labeling. Hierarchical phrase-based machine translation is often criticized for not explicitly incorporating linguistic knowledge. On the other hand, fully syntactic-based machine translation suffers from low coverage of rules. The methodology in this paper, in contrast, introduces linguistic information to assist a formalism that does not incorporate linguistic information. The merits of doing so are obvious. While most parts of the system are not changed, a portion of the system is considerably improved. Also, the system encodes the information in the non-terminal tags, which is widely used in other methods such as SAMT. However, it is not necessary an optimal solution. Huang et al in a very recent work (Huang et al., 2010) proposed using vector space to represent similarity between the syntactic structures. This is also an interesting possible direction to explore in the near future.

7 Conclusion and future work

In this paper we presented a method of utilizing the target side predicate-argument structure to assist Hierarchical Phrase-based Machine Translation. With a hybrid rule extraction algorithm, we can extract SRL-aware SCFG rules together with conventional Hiero rules. Additional conversion rules ensure the generated predicate-argument structures are complete when SRL-aware SCFG rules are used in the decoding procedure. Experimental results showed improvement on BLEU and TER metrics with 9 test sets, and even larger improvements are observed when only considering the sentences in which SRL-aware SCFG rules are used for the top-best derivation.

We are currently following three directions for the future work. The first focuses on improving the quality of the rules and feature estimation. We are investigating different labeling systems other than the relatively simple PropBank labeling system, and plan to experiment with different methods of mapping structure to the SRL-aware labels.

Recent advances in vector space representations on the syntactic structures, which may be able to work with, or replace the SRL-aware non-terminal labels, inspire the second direction.

Finally, the third direction is to incorporate source side semantic role labeling information into the framework. Currently our method can only use target side SRL information, but the source side information is also valuable. Exploring how to build models to represent SRL information from both sides into one complete framework is a promising research direction to follow.

References

Collin F. Baker, Charles J. Fillmore, and Beau Cronin. 2002. The structure of the framenet database. *International Journal of Lexicography*, 16(3):281–296.

David Chiang. 2005. A hierachical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of Association for Computational Linguistics*, pages 263–270.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228.

David Chiang. 2010. Learning to translate with source and target syntax. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1443–1452.

Chris Dyer et al. 2010. cdec: A decoder, alignment, and learning framework for finite-state and context-free translation models. In *Proceedings of the ACL 2010 System Demonstrations*, ACLDemos '10, pages 7–12.

Michel Galley et al. 2006. Scalable inference and training of context-rich syntactic translation models. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 961–968.

Qin Gao and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57, Columbus, Ohio, June. Association for Computational Linguistics.

Hieu Hoang and Philipp Koehn. 2008. Design of the moses decoder for statistical machine translation. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, SETQA-NLP '08, pages 58–65, Morristown, NJ, USA. Association for Computational Linguistics.

Zhongqiang Huang, Martin Cmejrek, and Bowen Zhou. 2010. Soft syntactic constraints for hierarchical phrase-based translation using latent syntactic distributions. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 138–147, Cambridge, MA, October. Association for Computational Linguistics.

Philipp Koehn et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic, June. Association for Computational Linguistics.

Zhifei Li et al. 2009. Joshua: an open source toolkit for parsing-based machine translation. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, StatMT '09, pages 135–139.

Ding Liu and Daniel Gildea. 2008. Improved tree-to-string transducers for machine translation. In *ACL Workshop on Statistical Machine Translation (ACL08-SMT)*, pages 62–69.

Ding Liu and Daniel Gildea. 2010. Semantic role features for machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING-10)*.

Yang Liu and Liang Huang. 2010. Tree-based and forest-based translation. In *Tutorial Abstracts of ACL 2010*, page 2, Uppsala, Sweden, July. Association for Computational Linguistics.

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment template for statistical machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 609–616.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106, Mar.

Sameer S. Pradhan, Wayne Ward, Kadri Hacioglu, James H. Martin, and Daniel Jurafsky. 2004. Shallow semantic parsing using support vector machines. In *Proceedings of the Human Language Technology Conference/North American chapter of the Association for Computational Linguistics annual meeting (HLT/NAACL-2004)*.

Libin Shen, Jinxi Xu, and Ralph Weischedel. 2008. A new string-to-dependency machine translation algorithm with a target dependency language model. In *Proceedings of ACL-08: HLT*, pages 577–585, Columbus, Ohio, June. Association for Computational Linguistics.

Dekai Wu and Pascale Fung. 2009. Semantic roles for smt: a hybrid two-pass model. In *Proceedings of The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 13–16.

Min Zhang, Hongfei Jiang, Aiti Aw, Haizhou Li, Chew Lim Tang, and Sheng Li. 2008. A tree sequence alignment-based tree-to-tree translation model. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL-08 HLT)*, pages 559–567.

Andreas Zollmann and Ashish Venugopal. 2006. Syntax augmented machine translation via chart parsing. In *Proceedings of the Workshop on Statistical Machine Translation*, pages 138–141.