CCG Augmented Hierarchical Phrase-Based Machine Translation

Hala Almaghout, Jie Jiang, Andy Way

CNGL, School of Computing
Dublin City University
Dublin, Ireland
{halmaghout, jjiang, away}@computing.dcu.ie

Abstract

We present a method to incorporate target-language syntax in the form of Combinatory Categorial Grammar in the Hierarchical Phrase-Based MT system. We adopt the approach followed by Syntax Augmented Machine Translation (SAMT) to attach syntactic categories to nonterminals in hierarchical rules, but instead of using constituent grammar, we take advantage of the rich syntactic information and flexible structures of Combinatory Categorial Grammar. We present results on Chinese-English DIALOG IWSLT data and compare them with Moses SAMT4 and Moses Phrase-Based systems. Our results show 5.47% and 1.18% BLEU score relative increase over Moses SAMT4 and Phrase-Based systems, respectively. We conduct analysis on the reasons behind this improvement and we find out that our approach has better coverage than SAMT approach. Furthermore, Combinatory Categorial Grammar-based syntactic categories attached to nonterminals in hierarchical rules prove to be less sparse and can generalize better than syntactic categories extracted according to SAMT method.

1. Introduction

Current trends in statistical machine translation research are trying to provide SMT systems with syntactic knowledge in order to produce more grammatical translations. Incorporating syntax in SMT systems proved to be uneasy task mainly because phrases extracted in SMT systems do not necessarily correspond to syntactic constituents. Moreover, extracting only phrases which are covered by syntactic constituents proved to damage the performance of SMT systems [1]. However, that has not stop researchers from believing that current SMT systems still need to use syntactic knowledge to achieve better translation quality. This led to the emergence of many approaches which try to use syntax in SMT systems while maintaining the advantages of Phrase-Based SMT systems. Some of those approaches, such as the ones presented in [2] and [3], even achieved a promising improvement over baseline Phrase-Based SMT system.

In this paper, we introduce an approach to incorporate Combinatory Categorial Grammar (CCG) [4] in the Hierarchical Phrase-Based (HPB) MT system [2]. Our approach is similar to the one followed by Syntax Augmented Machine Translation (SAMT) [3] with a difference in the type of the grammar used. We try to use rich syntactic information and flexible structures of CCG supertags, which are derived from the CCG parse tree of the target-sentence, to attach syntactic labels to nonterminals in the hierarchical rules of the Hierarchical Phrase-Based MT system.

The rest of this paper is organized as follows: in the next section we review previous syntax based approaches to SMT. In Section 3 we introduce Syntax Augmented Machine Translation (SAMT) upon which we build our approach. Section 4 gives a brief introduction to Combinatory Categorial Grammar (CCG). In Section 5 we introduce our approach. Section 6 presents our experiments. In Section 8 we perform manual analysis for the translation output of our system and compare it with the output of other systems. Section 9 concludes, and provides avenues for further work.

2. Previous work

Many syntax based approaches to SMT use syntax annotation on the target side mainly because it helps to produce more fluent output and partly because a large part of MT research focuses on translation into English, which has many syntactic tools and resources available.

The method described in [5] uses a syntax-based language model in noisy-channel machine translation. Context Free Grammar (CFG) rules are extracted from the parsed target side of the parallel corpus, then a language model is extracted. The decoder used is a bottom up parser, is given a sentence in the foreign language, and looks for the best parse tree of the translation in the target language according to the language model and the translation model. The approach presented in [6] extracts multi-level syntactic translation rules that have multi-level target tree annotations from aligned tree-string pairs. Then, a large number of derivations is constructed. Those derivations include contextually richer rules, and account for multiple interpretations of unaligned words. The method presented in [7] extracts rules with syntactified target language phrases. The model uses feature functions that have been initially developed in phrase-based systems for choosing target translations of the source-language phrases and uses target-language submodels for assembling target phrases into a grammatical output.

The Hierarchical Phrase-Based (HPB) model [2] extracts synchronous context-free grammar (SynCFG) automatically from the parallel corpus without using any syntactic annotation. Those hierarchical rules are hierarchical phrase pairs which contain nonterminals that act as placeholders for inserting other phrase pairs, which may contain nonterminals. Hierarchical Phrase-Based rules take advantage of the strength of phrase pairs extracted according to the method presented in [1] and add to them the ability to translate discontinuous phrases and play the role of reordering at the same time. While nonterminal of hierarchical rules in HPB system do not carry any syntactic annotation, Syntax Augmented Machine Translation (SAMT) [3] augment nonterminals of HPB rules with syntactic labels extracted from the parsed target-side of the corpus. We build our approach upon SAMT with a difference in the type of grammar used to derive the syntactic labels; while SAMT uses constituent grammar, we use Combinatory Categorial Grammar (CCG) [4].

The richness of CGG supertags and their flexible structures inspired many approaches to incorporate them into Statistical Machine Translation systems. Work presented in [8] uses CCG supertags as a factor in Factored Translation Model [9] following two approaches: the first approach generates CCG supertags as a factor in the target side and then apply an n-gram language model over them, while the second approach uses supertags on the source side to direct the decoding process. The method introduced in [10] extends the Direct Translation Model (DTM2) [11] with target language syntax while maintaining linear-time decoding by incremental interpretation of CCG supertags of the target translation. The approach presented in [12] integrates CCG supertags into the target language model and the target side of the translation model in Phrase-Based Statistical Machine Translation (PBSMT) system. It also uses a surface global grammaticality measure based on CCG combinatory operators to check the grammatical correctness of the translation output.

3. Syntax Augmented Machine Translation

SAMT was introduced in [3] as an extension to Hierarchical Phrase-Based MT [2]. Before we explain how SAMT rules are extracted, a short review of the Hierarchical Phrase-Based machine translation approach, upon which both SAMT and our approach are based, will be presented.

3.1. Hierarchical Phrase-Based Machine Translation

The main idea behind the Hierarchical Phrase-Based MT [2] approach is to combine the strengths of both syntax-based and phrase-based translation approaches; it benefits form the strength of phrase pairs extracted according to Phrase-Based MT approach and adds to it by learning reordering and translation of discontinuous phrases using hierarchical rules.

Hierarchical Phrase-Based machine translation model is formally based on synchronous context-free grammar (SCFG), but it is learnt from a parallel text which do not have any syntactic annotation. SCFG elementary structures are rewrite rules with aligned pairs of right-hand sides:

$$X \rightarrow < \alpha, \beta, \sim >$$

Where X is a nonterminal, α and β are both strings of terminals and nonterminals, and $\sim$ is a one-to-one correspondence between nonterminal occurrences in α and nonterminal occurrences in β. The following is an example of the SCFG extracted from the Mandarin-English sentence pair (Aozhou shi yu Beihan you bangjiao de shaoshu guojia zhiyi, Australia is one of the few countries that have diplomatic relations with North Korea) [2]:

$$X \rightarrow < \ yu\ X_1\ you\ X_2\ ,\ have\ X_2\ with\ X_1\ >$$

$$X \rightarrow < \ X_1\ de\ X_2\ ,\ the\ X_2\ that\ X_1\ >$$

We can see from the previous example that SCFG extracted according to the Hierarchical Phrase-Based MT approach capture the hierarchical nature of the language without using any syntactic annotation. X_1 and X_2 nonterminals used in the rules in the previous example are replaced during translation by other phrase-pairs, which can be also hierarchical, meaning that they can contain nonterminals which will replaced by other phrase-pair. Recursive nonterminal replacement is continued until the whole source sentence is covered.

The downside of this approach is that there is no syntactic constraint imposed on the nonterminal replacement during translation and this results in ungrammatical translations. SAMT tries to alleviate this problem by using target-side syntax to assign syntactic categories to nonterminals in hierarchical rules. In this way, syntax-augmented nonterminals act as syntactic constraints on the target side of the phrase-pairs during nonterminal replacement, resulting in more grammatical translations.

3.2. SAMT rule extraction

SAMT rules are extracted according to the following steps. First, each sentence in the target side is parsed according to constituent grammar and assigned a syntactic tree. Then, phrase pairs are extracted from the parallel corpus according to the method presented in [1]. Next, each of the previously extracted phrase pairs is assigned a syntactic label which corresponds to the syntactic constituent in the parse tree that covers the target phrase. This label corresponds to the left-hand side of the hierarchical rule extracted from this phrase-pair. In case the target phrase does not correspond to a constituent in the parse tree, the phrase is assigned an extended category of the form C1+C2, C1/C2, or C2\C1, indicating that the phrase pair's target side spans two adjacent syntactic categories (e.g., she went: NP+V), a partial syntactic category C1 missing a C2 to the right (e.g., the

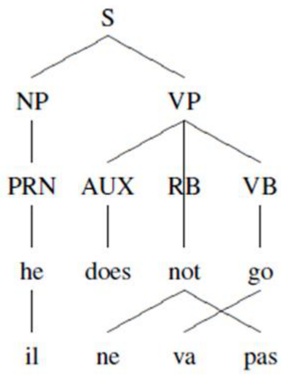

Figure 1: Parse tree of the English sentence (he does not go) along with its aligned French source sentence.

great: NP/NN), or a partial C1 missing a C2 to the left (e.g. great wall: DT\NP), respectively. Last, hierarchical rules are extracted from syntactic-labeled phrases according to the method presented in [2]. Figure 1 illustrates the syntax tree associated with the English sentence (he does not go) along with its aligned source French sentence. Figure 2 shows a sample of the hierarchical rules extracted from the sentence pair (French: il ne va pas, English: he does not go) and its associated target parse tree illustrated in Figure 1 [13].

In our experiments, we use Moses SAMT4 [1]to build our baseline SAMT system. Moses SAMT4 uses the same labeling method proposed by SAMT, but in addition to SAMT basic operators (+,\, /), three additional operators are used :

- ++ to combine two adjacent constituents C1 and C2 which are not children of the same parent.
- // for example C1//C2 indicates partial syntactic category C1 missing a C2 to the right. C2 is not a direct child of C1.
- \\ for example C1\\C2 indicates partial syntactic category C1 missing a C2 to the left. C2 is not a direct child of C1.

For example in Figure 1, the phrase (he does not) is assigned S//VB as a syntactic label. // is used instead of / because VB is not a direct child of S in the tree. The phrase (he does) is assigned PRN++AUX as a syntactic label because PRN and AUX nodes do not have the same direct parent.

4. Combinatory Categorial Grammar (CCG)

CCG [4] is a grammar formalism which consists of a lexicon, which pairs words with lexical categories (supertags), and a set of combinatory rules, which specify how the categories are combined. A supertag is a rich syntactic description that specifies the local syntactic context of the word. Most of the CCG grammar is contained in the lexicon thus CCG has simple combinatory rules to combine CCG supertags. CCG categories are divided into atomic and complex categories. Examples of atomic categories are: S(sentence), N (Noun), NP (noun phrase). Complex categories, such as S\NP and (S\NP)/NP, are functions which specify the type and directionality of their arguments (primitive or complex categories) and the type of their result (primitive or complex category). The notation used to express complex categories is the following:

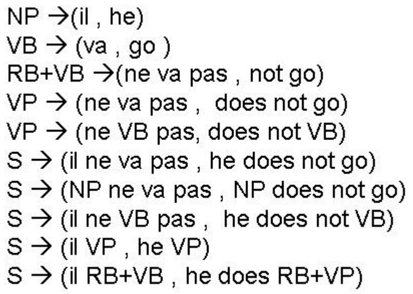
NP →(il , he)
VB → (va , go)
RB+VB →(ne va pas , not go)
VP → (ne va pas , does not go)
VP → (ne VB pas, does not VB)
S → (il ne va pas , he does not go)
S → (NP ne va pas , NP does not go)
S → (il ne VB pas , he does not VB)
S → (il VP , he VP)
S → (il RB+VB , he does RB+VP)

Figure 2: Some hierarchical rules extracted from the sentence pair (French: il ne va pas, English: he does not go) according to SAMT method.

- X\Y is a functor X which takes as an argument the category Y to its left (which might be a primitive or complex category) and the result is the category X (which might also be a primitive or complex category).
- X/Y is a functor which takes as an argument the category Y to its right (which might be a primitive or complex category) and the result is the category X (which might also be a primitive or complex category).

For example the lexical category of the verb (eat) in the sentence (I eat) is S\NP, which means that this category is expecting an NP (which plays the role of the subject in this case) to its left and the result of this category when an NP comes to its left is a sentence (S). By contrast, in the sentence (I eat an apple) the lexical category assigned to the verb eat in this case is (S\NP)/NP, which means that it expects an NP to its left (which plays the role of the subject) and another NP to its right (which plays the role of the object), and the result of this category when all of its arguments are fulfilled is a whole sentence (S). Thus, the complex lexical category (S\NP)/NP represents a transitive verb while the lexical category (S\NP) represents an intransitive verb.

4.1. CCG operators

As most of the CCG grammar is contained in the lexicon, CCG rules consist of three simple combinatory operators: Application operators, composition operators and type raising operators.

[1] http://www.statmt.org/moses/?n=Moses.SyntaxTutorial

Application operators: there are two types of application operators:

- Forward application operator: combines a category X/Y with category Y to its right and the result is category X.

$$\frac{\alpha : X/Y \quad \beta : Y}{\alpha\beta : X} >$$

- Backward application operator: combines a category X\Y with category Y to its left and the result is category X.

$$\frac{\beta : Y \quad \alpha : X\backslash Y}{\beta\alpha : X} <$$

Composition operators: composition operators are divided into:

- Forward composition operator: combines category X/Y with category Y/Z and the result is category X/Z.

$$\frac{\alpha : X/Y \quad \beta : Y/Z}{\alpha\beta : X/Z} >$$

- Backward composition operator: combines category Y\Z with category X\Y category and the result is category X\Z.

$$\frac{\beta : Y\backslash Z \quad \alpha : X\backslash Y}{\beta\alpha : X\backslash Z} <$$

Type raising operators: type raising operators turn arguments into functions over functions over such arguments. There are two types of type raising operators:

- Forward type raising operator: transforms category X into T(/X\T).

$$\frac{\alpha : X}{\alpha : T/(X\backslash T)} >$$

- Backward type raising operator: transforms category X into T\(X/T).

$$\frac{\alpha : X}{\alpha : T\backslash(X/T)} <$$

4.2. CCG parsing

Parsing using CCG can be viewed as a two-stage process: first, assiging lexical categories to the words in the sentence (supertagging). Then, combining the categories together using CCG combinatory operators.

The idea underlying the supertagging approach proposed by [14] is that the computation of linguistic structure can be localized if lexical items are associated with rich descriptions (supertags) that impose complex constraints in a local context. This makes the number of different descriptions for each lexical item much larger than when the descriptions are less complex, thus increasing the local ambiguity for a parser. However, this local ambiguity can be resolved by using statistical distributions of supertag co-occurrences collected from a corpus of parses. Supertag disambiguation results in a representation that is effectively a parse **(an almost parse)**. The advantage of this supertagging approach is that the number of categories assigned to each word can be reduced, with a correspondingly massive reduction in the number of derivations.

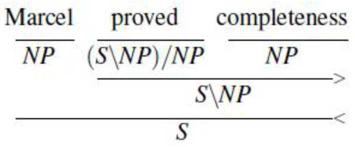

Figure 3: CCG parse tree of the sentence: (Marcel proved completeness).

After supertagging, the parser is required to do less work once the lexical categories have been assigned. That is why supertagging is referred as almost parsing. The parser builds CCG parse tree by using combinatory operators to combine supertags into a derivation tree. Figure 3 illustrates the CCG parse tree of the sentence: (Marcel proved completeness).

5. Our approach

Inspired by SAMT approach, we assign syntactic categories to phrases and nonterminals in hierarchical rules. However, instead of using constituent grammar based categories, we try to benefit from the advantages which CCG formalism provide for use in HPB system. First, and most importantly, a supertag can be assigned to any sequence of words whether they form a syntactic constituent or not. This feature is very important as phrases used in Hierarchical Phrase-Based System do not necessarily correspond to syntactic constituents. Second, because most of the structure of CCG grammar is contained in the lexicon, CCG supertags provide rich syntactic information about the word dependents and its syntactic context at word level. Thus, hierarchical rules with CCG-labeled nonterminals help to produce more grammatical translation by choosing phrases which best fit in their syntactic context to replace nonterminals. Last, because of its simple combinatory rules and rich syntax at lexical level, CCG is parsed efficiently compared to constituent grammar.

SAMT tries to use CCG-like operators to combine constituents derived from the constituent grammar-based parse tree in order to form a syntactic category which looks like a CCG supertag. However, rigid constituent grammar-based structures are not as flexible as CCG structures, which have a large degree of flexibility to be combined using a set of simple operators without imposing a rigid constituent-based structure on the combined substructures. Thus, SAMT approach for nonterminlas labeling would fail to find a label for many phases-pairs which do not comply with any of SAMT labeling rules ,and as a result, SAMT has less coverage in comparison with CCG approach. Moreover, CCG supertags are designed such that only those elements on which the lexical item imposes constraints appear within a given supertag, while SAMT syntactic categories combine constituents that do not necessarily impose syntactic constraints on each other and thus they won't accurately reflect the syntactic constraint imposed on the phrase. This gives another advantage for CCG approach over SAMT by guaranteeing that every CCG based label assigned to a phrase reflects the right syntactic context within which the phrase best fit.

Figure 4 demonstrates the constituent grammar-based parse tree of the sentence:(Australia is one of the countries which have diplomatic relations with North Korea). Suppose that this sentence is a target translation of a sentence in some foreign language and a phrase pair, whose target side is "have diplomatic relations", is extracted. In this case SAMT would assign VP/PP to it as its syntactic category. However, this phrase does not necessarily need a PP as a part of its syntactic context and thus imposing this constraint on the phrase is not correct. Whereas using CCG phrase labeling approach, this same phrase is assigned a syntactic label S[dcl]\NP that results from combining the supertags (S[dcl]\NP)/NP, N/N and N of the words: "have", "diplomatic" and "relations" respectively as illustrated in Figure 5. S[dcl]\NP accurately reflects the syntactic constraint imposed on such phrase; its a verb phrase that needs a noun phrase, which plays the role of the subject, to its right in order to form a complete sentence. In addition, SAMT fails to find a syntactic category for the phrase: "is one of the countries" because this phrase does not comply to any of the SAMT labeling rules, whereas our CCG-based approach succeeds in finding a syntactic category S[dcl]\NP for it by simply combining the supertags: (S[dcl]\NP)/NP, NP, (NP\NP)/NP, NP[nb]/N and N of the words: "is", "one", "of" , "the" and "countries" respectively. This category correctly reflects that this phrase, which plays the role of a verb phrase, needs a noun phrase, playing the role of the subject, to its right in order to from a complete sentence.

Extracting CCG augmented hierarchical rules is done in a way similar to SAMT rules extraction:

- First, each target-side sentence from the parallel corpus is supertagged by assigning the best sequence of CCG supertags to its words.
- Next, phrase pairs are extracted from the parallel corpus according to the method presented in [1].
- Then, each extracted phrase pair is a assigned a CCG supertag that results from combining the supertags of the target phrase words. In case no CCG supertag can be assigned to the phrase, a general X label is assigned to it.
- Finally, hierarchical rules are extracted from sentence-pairs according to the method specified in [2]

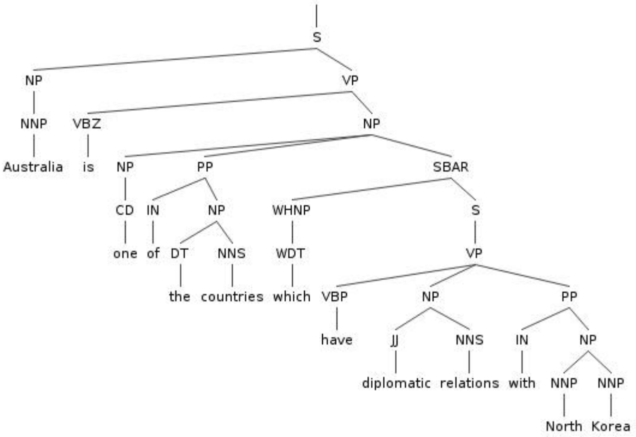

Figure 4: Constituent grammar parse tree of the sentence: (Australia is one of the countries which have diplomatic relations with North Korea).

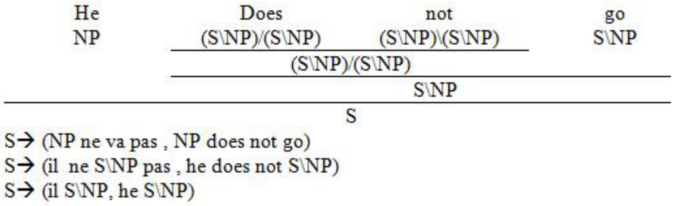

Figure 6: CCG derivation tree along with a sample of hierarchical rules extracted from the sentence pair (French: il ne va pas, English: he does not go) according to our CCG-based approach.

Figure 6 illustrates the CCG parse tree of the English sentence of the aligned sentence pair: (French: il ne va pas, English: he does not go), and a sample of hierarchical rules extracted from this pair.

6. Experiments

We conduct our experiments using Chinese to English DIALOG corpus for IWSLT. The training corpus consists of 63234 sentence pairs taken from the IWSLT 2010 training data. The development and test set are IWSLT evaluation data sets for the DIALOG task for 2008 and 2009 evaluations, respectively with 500 sentence pair each. The development set has 15 references and the test set has 7 references.

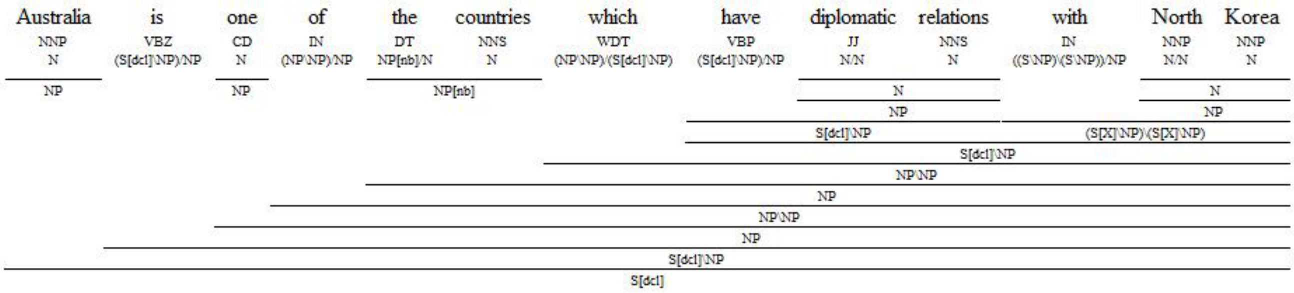

Figure 5: CCG parse tree of the sentence: (Australia is one of the countries which have diplomatic relations with North Korea).

GIZA++ toolkit [2] is used to perform word alignment and "grow-diag-final" refinement method is adopted [1]. Minimum error rate training [15] is performed for tuning. The language model in all experiments is 5-gram trained on the target side from the parallel corpus using The SRILM toolkit [3] with modified Kneser-Ney smoothing [16]. We built and compared three systems:

- Phrase-Based System: built using the Moses Phrase-Based Decoder [4] with maximum phrase length=12.
- Moses SAMT4: built using the Moses Chart-Decoder with the SAMT4 scheme [5]. Maximum phrase length is set to 12 and maximum rule span is set to 15. Rules extracted contain up to 2 nonterminals. The English side of the training corpus is parsed using the Berkeley Parser. [6]
- CCG-based system: built using the Moses Chart-Decoder. Maximum phrase length is set to 12 and maximum rule span is set to 15. Rules extracted contain up to 2 nonterminals. The English side of the training corpus is parsed using CCG parser from C&C tools. [7]

7. Experiments Results

Table 1 shows BLEU scores that result from evaluating the Phrase-Based , the Moses SMAT4 and the CCG-based MT systems on IWSLT evaluation data set for 2009 using 7 references.

Table 1 shows that the CCG-based system has 2.64 BLEU points increase over the SAMT4 system and 0.57 BLEU points increase over the Phrase-Based system. That means the CCG-based system achieves 5.47% and 1,18% relative increase over the SAMT4 and the Phrase-Based systems respectively. Thus, we can see that the CCG-based system outperforms both of the Phrase-Based and the Moses SAMT4 systems. In order to pin down the reasons behind the improvement of the CCG-based system over the SAMT4 system, we investigate the rule tables of both systems in terms of their size, the number of different syntactic labels used in the rule table and the number of phrase pairs failed to have a syntactic label.

System	BLEU
Phrase-Based System	47.69
Moses SAMT4 System	45.62
CCG System	48.26

Table 1: Experiments results.

System	Rule table size
SAMT4 Sys	10,983,675
CCG Sys	6,578,072

Table 2: Rule table size of SAMT4 and CCG systems.

Table 2 shows the number of rules in the rule table of each of the SAMT4 and the CCG-based systems. The rule table of the SAMT4 system has about 1.67 times as many rules as the rule table of the CCG-based system. Given that we use the same rule extraction method with the same phrase length and rule span parameters for both the CCG-based and the SAMT4 systems, the number of rules extracted in each system depends on how many different syntactic labels are assigned to phrase-pairs extracted from the training corpus. The more different syntactic labels used in hierarchical rules, the more sparse the labels and the less able the system to generalize. That is why we conduct a comparison between the CCG-based and the SAMT4 systems by checking the number of different syntactic labels used to name non-terminals in the rule table. Table 3 shows that the SAMT4 system rule table has about 10 times as many different syntactic labels as the rule table of the CCG-based system, which means that

[2] http://fjoch.com/GIZA++.html
[3] http://www.speech.sri.com/projects/srilm/
[4] http://www.statmt.org/moses/index.php?n=Main.HomePage
[5] http://www.statmt.org/moses/?n=Moses.SyntaxTutorial
[6] http://code.google.com/p/berkeleyparser/
[7] http://svn.ask.it.usyd.edu.au/trac/candc/

System	Num of diff syntactic labels
SAMT4 Sys	4969
CCG Sys	456

Table 3: Number of different syntactic labels extracted.

System	% of X-label phrase-paris
SAMT4 Sys	50%
CCG Sys	25%

Table 4: Percentage of phrase-pairs without syntactic label (phrases which are assigned the general X label) out of the total number of labeled phrase-pairs.

SAMT4 syntactic labels are much more sparse than CCG-based syntactic labels. Data sparsity reduces system ability to generalize and it is a key factor affecting the performance of syntax-based Hierarchical Phrase-Based systems.

We also tested the coverage of syntactic categories extracted in the CCG-based and the SAMT4 systems. Table 4 shows that the SAMT4 system fails to label 50% of the total phrase-pairs while the CCG-based system fails to label only 25% of the total phrase-pairs, which means that CCG labeling method has a better coverage than that of the SAMT4 method. Our CCG-based nonterminal labeling takes advantage of the flexibility of CCG structures which proved to better suit non-grammatical phrases extracted in SMT. By contrast, constituent grammar based structures are rigid and designed to deal with grammatical phrases only. That is why SAMT fails more on extracting syntactic categories which cover non-grammatical phrases.

8. Manual analysis

Since the SAMT and CCG-based approaches are focused on the grammatical aspect of the MT systems, sometimes evaluation methods such as BLEU can not show their effect on producing more grammatical translations due to their lack of syntactical nature. To obtain a finer syntactical analysis on evaluation results, instead of adopting a syntactical rich evaluation metric, we carried out manual analysis to spot the effect of using CCG-based labels in comparison with SAMT labels. The following procedures are followed to accomplish manual analysis in this paper:

- Sentence level BLEU scores are calculated for the outputs of both the SAMT4 and the CCG-based systems.
- Sentences pairs from those two systems with higher BLEU differences than a threshold are selected (0.5 is used in this paper).
- A group of Randomly selected sentence pairs are collected from the last step for manual analysis.

Following the previous steps, we organized the sentences on which the CCG-based system performed better than the SAMT4 system. We see that most of those sentences have better translation because they benefit from the context sensitivity of CCG supertags on the target side. For example:

- **Source**: 有 没 有 京都 风景 的 明信片 ?
- **Ref**: do you have any postcards of kyoto scenes ?
- **SAMT4**: do you have *a* kyoto picture postcards ?
- **CCG**: do you have *any* kyoto picture postcards ?

It is interesting that the CCG-based system prefers the word *any* rather than *a* in a question sentence, while the SAMT4 system chooses *a* which is more likely to be correct in an assertive sentence. This example indicates that using CCG based labels improve translation quality by choosing words that better fit their syntactic context.

We also find some examples in which the CCG-based system is clearly better than the SAMT4 system but with a lower BLEU score. For example:

- **Source**: 请问 走 过去 要 多久 ?
- **Ref**: excuse me , how long does it take on foot ?
- **SAMT4**: excuse me , how long will pass , please ?
- **CCG**: excuse me , can you tell me how long it will take ?

In this example, although the SAMT4 system has a better BLEU score than the CCG-based system, the latter produced obviously a more grammatical translation than the first one. It can also be contributed to the context property of CCG, because *how long* is more likely to be related to *it will take* rather than *pass*.

From our manual analysis, we observe that the CCG-based system can produce more syntactically correct sentences due to its embedded context sensitivity features, while the SAMT4 system is more general, thus has less contribution to this aspect of MT systems.

9. Conclusion

In this work, we have presented an approach to attach Combinatory Categorial Grammar-based syntactic categories to nonterminals of Hierarchical Phrase-Based rules.

Our CCG-based system achieved significant improvement (5.47% BLEU score relative increase) over the SAMT4 system due to many reasons. First, a CCG supertag assigned to a phrase contains rich syntactic information about the constituents on which the phrase impose syntactic constraints. This enables it to capture the syntactic context of phrases more accurately than constituent-based syntactic categories. In addition, CCG supertags have a large degree of flexibility because they can be combined using a small set of simple operators while constituent-based labels extracted according to the SAMT method depend on rigid constituent grammar structures. This results in the CCG-based approach being more successful in finding a proper syntactic label for a phrase than SAMT, which fails to derive a syntactic label for many more phrases that do not comply with SAMT labeling rules. Last, CCG-based syntactic labels proved to be less sparse than SAMT-based labels, this reduces the size of the rule table of the model and makes it more able to generalize. Manual analysis we conducted for some sentences showed the ability of our CCG-based system to be more aware of the syntactic context of phrases used to form the translation output than the SAMT system and this explains its ability to produce more grammatical translations.

In future work, we will examine the performance of CCG augmented HPB system on different language pairs. Moreover, we will study the ways in which CCG supertags can be used in the extraction of hierarchical rules of the HPB system.

10. References

[1] P. Koehn, F. Och, and D. Marcu. 2003. *Statistical phrase-based translation.* In Proceedings of the 2003 Human Language Technology Conference and the North American Chapter of the Association for Computational Linguistics, Edmonton, AB, Canada, 2003, pp. 48‘54.

[2] D. Chiang. 2005. *A hierarchical phrase-based model for statistical machine translation.* In ACL-2005: 43rd Annual meeting of the Association for Computational Linguistics, Ann Arbor, MI, pp. 263 - 270.

[3] A. Zollmann and A. Venugopal. 2006. *Syntax augmented machine translation via chart parsing.* HLT-NAACL 2006: Proceedings of the Workshop on Statistical Machine Translation, New York, NY, USA, pp. 138-141.

[4] M. Steedman. 2000 *The Syntactic Process.* MIT Press, Cambridge, MA.

[5] E. Charniak, K. Knight, and K. Yamada. 2003. *Syntax-based language models for statistical machine translation.* MT Summit IX, New Orleans, USA, pp.40-46.

[6] M. Galley, J. Graehl, K. Knight, D. Marcu, S. DeNeefe, W. Wang, and I. Thayer. 2006. *Scalable inference and training of context-rich syntatic translation models.* Coling-ACL 2006: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Sydney, pp.961-968.

[7] D. Marcu, W. Wang, Aa Echihabi, and K. Knight. 2006. *SPMT: statistical machine translation with syntactified target language phrases.* EMNLP-2006: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, Sydney, Australia, pp. 44-52.

[8] A. Birch, M. Osborne, and P. Koehn. 2007. *CCG supertags in factored statistical machine translation.* ACL 2007: Proceedings of the Second Workshop on Statistical Machine Translation, June 23, 2007, Prague, Czech Republic, pp. 9-16.

[9] P. Koehn and H. Hoang. 2007. *Factored translation models.* EMNLP-CoNLL-2007: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, June 28-30, 2007, Prague, Czech Republic, pp. 868-876.

[10] H. Hassan, K. Sima'an, and A. Way. 2009. *A syntactified direct translation model with linear-time decoding.* EMNLP-2009: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Singapore, pp.1182-1191.

[11] A. Ittycheriah and S. Roukos. 2007. *Direct translation model 2.* NAACL-HLT-2007 Human Language Technology: the conference of the North American Chapter of the Association for Computational Linguistics, 22-27 April 2007, Rochester, NY, pp.57-64.

[12] H. Hassan, K. Sima'an, and A. Way. 2007. *Supertagged phrase-based statistical machine translation.* ACL 2007: proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, Prague, Czech Republic, pp. 288-295.

[13] A. Zollmann, A. Venugopal. 2009. *Grammar based statistical MT on Hadoop: an end-to-end toolkit for large scale PSCFG based MT.* Prague Bulletin of Mathematical Linguistics 91, 2009; pp.67-78.

[14] S. Bangalore and A. Joshi. 1999. *Supertagging: An Approach to Almost Parsing.* Computational Linguistics 25(2):237-265, 1999.

[15] F. Och. 2003. *Minimum Error Rate Training in Statistical Machine Translation.* ACL-2003:41st Annual meeting of the Association for Com-putational Linguistics, Sapporo, Japan, pp. 160-167.

[16] R. Kneser and H. Ney. 1995. *Improved Backing-Off for M-Gram Language Modeling.* In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP),Detroit, Michigan, USA, Vol. 1, pp. 181-184.