CCG-Augmented Hierarchical Phrase-Based Statistical Machine Translation

Hala Almaghout

A dissertation submitted in fulfilment of the requirements for the award of

Doctor of Philosophy (Ph.D.)

to the

Dublin City University
School of Computing

Supervisor: Prof. Andy Way and Dr. Jie Jiang

July 2012

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Ph.D. is entirely my own work, that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed:

(Candidate) ID No.:

Date:

Contents

List of Figures

List of Tables

Abstract

Augmenting Statistical Machine Translation (SMT) systems with syntactic information aims at improving translation quality. Hierarchical Phrase-Based (HPB) SMT takes a step toward incorporating syntax in Phrase-Based (PB) SMT by modelling one aspect of language syntax, namely the hierarchical structure of phrases. Syntax Augmented Machine Translation (SAMT) further incorporates syntactic information extracted using context free phrase structure grammar (CF-PSG) in the HPB SMT model. One of the main challenges facing CF-PSG-based augmentation approaches for SMT systems emerges from the difference in the definition of the constituent in CF-PSG and the 'phrase' in SMT systems, which hinders the ability of CF-PSG to express the syntactic function of many SMT phrases. Although the SAMT approach to solving this problem using 'CCG-like' operators to combine constituent labels improves syntactic constraint coverage, it significantly increases their sparsity, which restricts translation and negatively affects its quality.

In this thesis, we address the problems of sparsity and limited coverage of syntactic constraints facing the CF-PSG-based syntax augmentation approaches for HPB SMT using Combinatory Cateogiral Grammar (CCG). We demonstrate that CCG's flexible structures and rich syntactic descriptors help to extract richer, more expressive and less sparse syntactic constraints with better coverage than CF-PSG, which enables our CCG-augmented HPB system to outperform the SAMT system. We also try to soften the syntactic constraints imposed by CCG category nonterminal labels by extracting less fine-grained CCG-based labels. We demonstrate that CCG label simplification helps to significantly improve the performance of our CCG category HPB system. Finally, we identify the factors which limit the coverage of the syntactic constraints in our CCG-augmented HPB model. We then try to tackle these factors by extending the definition of the nonterminal label to be composed of a sequence of CCG categories and augmenting the glue grammar with CCG com-

binatory rules. We demonstrate that our extension approaches help to significantly increase the scope of the syntactic constraints applied in our CCG-augmented HPB model and achieve significant improvements over the HPB SMT baseline.

Acknowledgements

I would like to express my deep gratitude for my supervisors, Andy Way and Jie Jiang, for their guidance, constant support and encouragement, and for the great efforts they made in helping me accomplishing this research and writing my thesis. I would like also to thank my examiners, Jennifer Foster and Chris Callison Burch, for their comprehensive feedback.

I would like to acknowledge Science Foundation Ireland for supporting this research under Grant No. 07/CE/I1142 as part of the Centre for Next Generation Localisation at Dublin City University. I would like also to thank staff at Dublin City University for their collaboration and help.

I would like to thank the people without their love and support I would not have completed this work. I would like to thank my husband Murhaf for giving me the love which keeps me going, sharing with me every difficult and happy moment, and unlimitedly supporting me. I am extremely grateful for my wonderful parents, who first taught me to enjoy and seek learning, and for the great sacrifices they made to help me follow my ambitions and fulfil my dreams. I finally would like to thank my brothers and my friends, especially my cousin Hala, and my big family back home for their warm-heartedness, support and encouragement.

Chapter 1

Introduction

Despite the demonstrated success of Statistical Machine Translation (SMT) as a fast and cost-effective approach to perform automatic translation, translation quality of SMT systems is far from perfect. The earliest efforts to enhance the quality of SMT systems are represented in the evolution of the translation unit from words in word-based SMT models (Brown et al., 1988, 1990) through continuous word segments (phrases) in the Phrase-Based (PB) SMT model (Koehn et al., 2003) to hierarchical phrases in the Hierarchical Phrase-Based (HPB) SMT model (Chiang, 2005). However, translation units in these approaches are extracted via purely statistical methods and use the surface form of the words and their structure without incorporating any deeper syntactic information. The lack of syntactic knowledge in these SMT models causes them to err in performing translation, which can negatively affect translation quality. This has led to the emergence of approaches to incorporating syntactic knowledge represented as the syntactic structure and/or function of words/phrases in the source side, the target side and both sides of SMT systems.

HPB SMT tries to model the hierarchy of statistically extracted phrases, which is one aspect of language syntax. The HPB SMT model extracts a synchronous context free grammar (SCFG) from the parallel corpus without using any syntactic annotation. One of the earliest attempts to incorporate syntactic annotation into HPB SMT is Syntax Augmented Machine Translation (SAMT) (Zollmann and

Venugopal, 2006). SAMT uses context free phrase structure grammar (CF-PSG) to annotate phrases and nonterminals in the HPB SMT model. One of the challenges facing the incorporation of CF-PSG into the HPB SMT model emerges from the difference in the definition of the constituent in CF-PSG and the 'phrase' in SMT. A constituent in CF-PSG is a group of words which acts grammatically as a single unit. In contrast, the notion of the 'phrase' in SMT systems extends to including any continuous sequence of words extracted on statistical basis. Annotating phrases in SMT systems with CF-PSG constituent labels covers only phrases which correspond to syntactic constituents according to the CF-PSG formalism, thus leaving many phrases without syntactic annotation. As the flexible notion of the phrase in SMT systems is one of its most important strengths, excluding phrases which do not correspond to syntactic constituents will have a negative effect on performance (Koehn et al., 2003). On the other hand, phrases which are syntactically unannotated escape the control of syntactic constraints, which can have a negative effect on translation quality.

SAMT tries to increase the coverage of the syntactic annotation for HPB SMT over a limited set of CF-PSG constituent labels using a set of operators which combine CF-PSG constituent labels into more complex labels. These operators are inspired from Combinatory Categorial Grammar (CCG) (Steedman, 2000), which defines constituent labels in terms of functors and arguments. Although these 'CCG-like' operators help SAMT to increase the coverage over the CF-PSG constituent labels, they are not genuine in the sense that they are not part of the original CF-PSG formalism as is the case in CCG, which we believe limits the benefits sought from including them.

In light of previous research which demonstrated the advantages of incorporating CCG in PB SMT (Hassan et al., 2007, 2009), we argue that CCG has many advantages over CF-PSG when incorporating it in the HPB SMT model, which leads us to our first research question:

RQ1: *Is CCG better than CF-PSG when using it to annotate nonterminals and*

phrases in the HPB SMT model?

In this thesis, we follow the SAMT approach to augmenting phrases and nonterminals in the HPB SMT model with syntactic labels. We use CCG categories instead of CF-PSG-based labels to annotate phrases and nonterminals in the HPB SMT model. We compare our CCG-augmented HPB system with the SAMT system in terms of label coverage, sparsity, and expressiveness:

- Label coverage measures the proportion of phrases which are syntactically annotated (i.e. assigned a left-hand-side label) in the training data.
- Label sparsity measures the proportion of the labels which have rare occurrences in the training data.
- Label expressiveness refers to how accurate the labels are in expressing the syntactic function of the phrases without being redundant (i.e. different labels express the same syntactic function).

We demonstrate that CCG category labels provide more coverage for phrases than SAMT labels. This results from CCG's flexible structures, which provide the ability to label phrases even when they do not correspond to grammatical constituents in traditional terms. Furthermore, we show that CCG category labels are less sparse than SAMT labels although they present richer syntactic information. This leads to smaller translation models and more reliable rule probabilities. Moreover, we demonstrate that CCG category labels represent richer and more accurate syntactic information than SAMT labels. SAMT tries to enrich the CF-PSG constituent labels using 'CCG-like' operators. This helps to increase the coverage of the labels but creates redundant labels, which fragments label probability and increases label sparsity. In contrast, CCG categories are the building units of a grammar formalism. This makes nonterminal labels based on CCG categories inherently rich, accurately reflecting the syntactic context and dependents of the word/phrase without being redundant, which gives them larger expressive power

than SAMT labels. These advantages of CCG categories over SAMT labels are reflected in our experimental results, which demonstrate that our CCG-augmented HPB system outperforms the SAMT system in terms of BLEU (Papineni et al., 2002), METEOR (Lavie and Abhaya, 2007) and TER (Snover et al., 2006) scores in most of the experiments conducted on Arabic-to-English and Chinese-to-English translation. These results were re-enforced by our manual analysis which compares the translation output of our CCG-augmented HPB system to the HPB baseline and SAMT systems, and gives insights into the strengths and weaknesses of our CCG-augmented HPB system.

Although our CCG category system achieves better performance than the SAMT system, it does not outperform the HPB baseline system in most of the conducted experiments. We argue that this might be due to the sparsity of CCG labels, which impose strict syntactic constraints on the decoding search space. Thus, we want to examine whether softening these syntactic constraints by extracting less fine-grained CCG-based labels than CCG categories will help to improve the performance of our CCG category HPB system. This leads us to our second research question:

RQ2: *Does softening the syntactic constraints imposed by CCG category nonterminal labels by simplifying them help to improve performance?*

We address this question following two approaches. The first approach represents only contextual information of CCG categories in nonterminal labels. The second approach removes syntactic features held by some CCG categories from the nonterminal label representation. We also combine these two approaches together. Although these approaches reduce the amount of syntactic information represented in the nonterminal labels and thus might affect their accuracy, the simplified CCG labels still represent rich syntactic information but impose less strict syntactic constraints than CCG categories, which might overcome the damage that results from using less accurate labels and thus improve performance. We conduct experiments which examine the effect on performance of our label simplification approaches under different factors: the language pair, the size and domain of the data, and the size

of the language model. Our experiments demonstrate that CCG label simplification helps to significantly improve performance over the CCG category HPB system for Arabic-to-English, Chinese-to-English and French-to-English translation. Furthermore, our experiments demonstrate that the performance of the different label simplification approaches varied according to the different examined factors.

When addressing **RQ1**, we demonstrate that CCG categories have better coverage for phrases and nonterminals than SAMT labels. However, CCG categories do not provide a full coverage for phrases and nonterminals in the HPB SMT model. This leads to the next research question:

RQ3: *Why is the coverage of the syntactic constraints limited in our CCG-augmented HPB system ?*

The research conducted to address **RQ1** partially answers the previous research question. Our experiments demonstrate that up to 70% of extracted phrases are labelled with CCG categories in our CCG category HPB system. This means, of course, that at least 30% of the extracted phrases are left unannotated, which is one reason for the limited coverage of the syntactic constraints in our CCG-augmented HPB system. Another important reason for the limited scope of the syntactic constraints not only in our CCG-augmented HPB system but also in the SAMT system is that only part of the HPB grammar is augmented with syntax, namely hierarchical rules, whereas glue grammar rules are not syntactically augmented, which means they do not apply any syntactic constraints. We show that the application of glue grammar rules constitutes at least 30% of the total rules used in the derivations of the SAMT and CCG-augmented HPB systems, which indicates the limited coverage of syntactic constraints applied in these systems. We argue that non-syntactically aware translation rules and phrases are one of the major causes for the production of ungrammatical translations. Accordingly, we argue that extending the syntactic constraints in our CCG-augmented HPB system helps to improve its performance, which leads us to our fourth research question **RQ4**:

RQ4: *Does extending the syntactic constraints in our CCG-augmented HPB*

system help to improve performance?

We address this research question following a two-fold approach. Firstly, we extend the definition of the syntactic label to be composed of more than one CCG category. Secondly, we augment glue grammar with CCG combinatory rules which are applied during glue grammar rule application. We also try to apply our extension approaches in a soft manner under the Preference Grammars paradigm (Venugopal et al., 2009) in order to avoid the negative effect on performance of applying strict syntactic constraints. We conduct experiments which examine the application of our extension approaches on data from different domains. We also explore the effect of sentence length on the performance of our systems. Our experiments demonstrate that our extension approaches help to significantly improve the performance of our CCG-augmented HPB system over the HPB baseline for Arabic-to-English and Chinese-to-English translation. Furthermore, our experiments show that our extension approaches help to significantly increase the coverage of the syntactic constraints applied in our CCG-augmented HPB system. Our experimental results demonstrate that our extension approaches show different performance trends depending on the sentence length, data domain and the strictness of the syntactic constraints. We therefore conduct an in-depth analysis to explain these different trends and discuss the problems facing our approaches.

1.1 Main Findings of the Thesis

The main findings of the thesis are summarised as follows:

- Using CCG categories extracted from CCG parsing chart as nonterminal labels in the HPB SMT model is demonstrated to achieve better performance than the CF-PSG-based SAMT labels. The CCG category labels are demonstrated to be more expressive, less sparse and have better coverage than the SAMT labels.

- Extracting less fine-grained CCG-based nonterminal labels from CCG cate-

gories helps to significantly improve performance over the CCG category HPB system.

- The syntax-free glue grammar and the syntactically unannotated phrases are the factors limiting the coverage of the syntactic constraints in our CCG-augmented HPB system.

- Extending the coverage of the syntactic constraints in our CCG-augmented HPB system by increasing the coverage of the CCG-based nonterminal labels and augmenting glue grammar rules with CCG combinatory rules helps to significantly improve the performance of our CCG-augmented HPB system over the HPB baseline system.

1.2 Thesis Structure:

The remainder of this thesis is organized as follows:

- Chapter 2 reviews work related to the research presented in this thesis. The chapter puts a special focus on the PB, HPB and SAMT SMT models, which are closely related to our models. The chapter also provides an overview of different syntax augmentation approaches for SMT systems in addition to an introduction to CCG, which is the grammar formalism we use in our research.

- Chapter 3 introduces our CCG-augmented HPB system which uses CCG categories as nonterminal labels. The chapter presents the research conducted to address our first research question (**RQ1**).

- Chapter 4 presents our approaches towards the simplification of CCG-based nonterminal labelling. The chapter presents the research conducted to address our second research question (**RQ2**).

- Chapter 5 discusses the limitations on the coverage of the syntactic constraints in our CCG-augmented HPB system, which is related to our third research

question (**RQ3**). The chapter also introduces our approaches to target these limitations by extending the syntactic constraints in our CCG-augmented HPB system, which addresses our last research question (**RQ4**).

- Chapter 6 presents conclusions and provides avenues for future work.

1.3 Publications

Publications which are based on the research conducted in this thesis include:

- (Almaghout et al., 2010a) entitled "CCG augmented hierarchical phrase-based machine translation" was published in proceedings of the 7th International Workshop on Spoken Language Translation (IWSLT 2010).

- (Almaghout et al., 2010b) entitled "The DCU machine translation systems for IWSLT 2010" was published in proceedings of the 7th International Workshop on Spoken Language Translation (IWSLT 2010).

- (Almaghout et al., 2011) entitled "CCG contextual labels in hierarchical phrase-based SMT" was published in proceedings of the 15th conference of the European Association for Machine Translation (EAMT 2011).

- (Banerjee et al., 2011) entitled "The DCU machine translation systems for IWSLT 2011" was published in proceedings of the International Workshop on Spoken Language Translation (IWSLT 2011).

- (Almaghout et al., 2012) entitled " Extending CCG-based syntactic constraints in hierarchical phrase-based SMT" was published in proceedings of the 16th conference of the European Association for Machine Translation (EAMT 2012).

Chapter 2

Related Work

In this chapter, we shed light on the work conducted in the Statistical Machine Translation (SMT) research field related to the work presented in this thesis. We start by giving an introduction to SMT in Section 2.1, which is followed by an introduction to word-based SMT in Section 2.2. We then give a detailed description of Phrase-Based and Hierarchical Phrase-Based SMT, which are state-of-the-art SMT approaches on which our work is based, in Sections 2.3 and 2.4, respectively. In Section 2.5, we provide an overview of the syntax augmentation approaches for SMT systems. This is followed in Section 2.6 by a detailed overview of Syntax Augmented Machine Translation, with which our work shares many similarities. Finally, Section 2.7 provides an introduction to Combinatory Categorial Grammar (CCG), which is the grammar formalism we use throughout our work, in addition to an overview of previous CCG-augmented SMT systems.

2.1 Statistical Machine Translation

Early MT systems based on rule-based transfer models met with a limited success. Manually transforming the translation expertise of the translator into a large number of complex machine-understandable transfer rules has been demonstrated to be difficult and expensive to implement and maintain. That is why these MT systems

had a limited application using domain-specific language only. The increase in computational power and the availability of translated texts in electronic format opened new avenues for MT research which explore the extraction of translation knowledge automatically form already translated texts using statistical machine learning methods. These approaches are known as Statistical Machine Translation (SMT).

SMT is by far the dominant MT approach used today. SMT uses a statistical machine learning algorithm to extract translation knowledge encapsulated in already translated texts through a process called training. The model that is built via the training process is then used to translate a new input text in the source language into an output text in the target language, a process known as decoding. The earliest SMT system was introduced by IBM researchers (Brown et al., 1988, 1990). Brown et al. (1990) define the problem of SMT as finding the most probable translation $\hat{e}$ for a given source sentence f, as in (2.1):

$$\hat{\mathbf{e}} = \underset{e}{\operatorname{argmax}}\, p(e|f) = \underset{e}{\operatorname{argmax}}\, p(f|e)p(e) \tag{2.1}$$

Equation (2.1) represents the noisy channel model for SMT, which is a generative model that considers the source-language sentence (observed sentence) f as a distorted version of the target-language sentence e. The decoding process in this model tries to find the best target-language sentence $\hat{e}$ which produced the observed sentence f. Equation (2.1) combines two models which participate in the decoding process: the translation model component $p(f|e)$ and the language model (LM) component $p(e)$. The LM provides an estimation of the fluency of the target-translation e. The translation model calculates the probability of translating the source sentence f into the target sentence e. In early SMT models (Brown et al., 1988, 1990), translation model probability was calculated based on word-to-word translation probability (cf. Section 2.2). However, in state-of-the-art SMT models, the calculation of the translation model probability is based on the probability of a longer translation unit, which is called the phrase (cf. Section 2.3).

2.2 Word-Based SMT

Brown et al. (1993) define what have come to be known as the *IBM Models*, a set of word-based generative SMT models that use the word as its main translation unit. These models learn mappings between the words of the source-language sentence and the words of the target-language sentence, in a process known as word alignments. IBM Models use Expectation Maximization (Dempster et al., 1977) to learn word alignments which maximise the likelihood of a sentence-aligned corpus during training. They define the translation probability $p(f|e)$ according to the equation in (2.2):

$$p(f|e) = \sum_{a} p(f, a|e) \tag{2.2}$$

where a is a vector which defines for each word f_i in f the position of the corresponding word e_j in e. Each word in e can have at most one aligned word from f, whereas each word in f can have one or more aligned words from e. Figure 2.1 shows word alignments between the Arabic source sentence and its corresponding English translation *shura council rejects issuance of new press law*. We can see that all the English words are aligned to at most one Arabic word, whereas some Arabic words such as اصدار are aligned to more than one English word. We can also notice the different ordering of some words between the Arabic and English sentences.

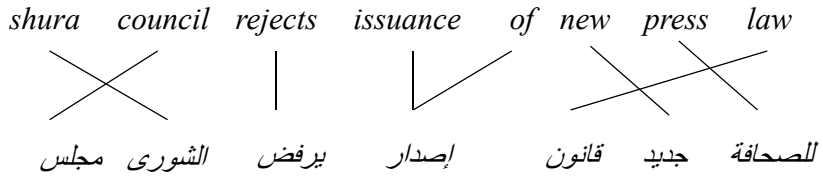

Figure 2.1: Word alignments between an Arabic source sentence and its corresponding English target translation.

Word-based SMT systems do not achieve a good translation quality. However, they are able to produce good quality alignments (Brown et al., 1993). Nowadays,

word-to-word SMT models are only used by current Phrase-Based SMT systems to learn word alignments as part of the training process.

2.3 Phrase-Based SMT

The use of the word as the basic translation unit cannot capture the translation and reordering of multi-word blocks. That led to the development of approaches which use a more complex translation unit composed of a continuous string of words – the phrase – which is why these approaches are called Phrase-Based (PB) SMT (Koehn et al., 2003). Figure 2.2 demonstrates the pipeline of the PB SMT system, which consists of three main stages: training, tuning and testing (decoding). Training consists of two main stages: word alignment (cf. Section 2.2) and phrase extraction (cf. Section 2.3.1).

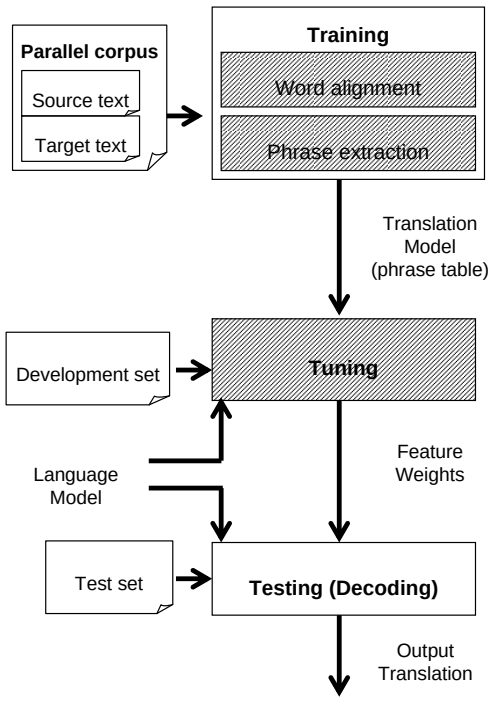

Figure 2.2: The pipeline of the PB SMT system.

The power of PB SMT does not only lie in the use of the phrase as the basic translation unit, but also in the following points (Goutte et al., 2009):

- The use of the log-linear framework (Och and Ney, 2002) to calculate translation probability (cf. Section 2.3.2). This allows for the incorporation of different models than the translation and language models used in the noisy channel model.

- The use of Minimum Error Rate Training (MERT) (Och, 2003) (cf. Section 2.3.3) to tune the weights of the different features incorporated in the log-linear model in terms of an MT evaluation metric such as BLEU (Papineni et al., 2002).

- Efficient heuristic beam search performed by the PB SMT system during decoding avoids conducting an exhaustive search of translation possibilities (cf. Section 2.3.4).

- PB SMT uses rescoring procedures to select the best translation from a set of candidate translations (cf. Section 2.3.5), which allows for an easy and fast incorporation of new features.

These aforementioned concepts have been used by not only the PB SMT system but also by other phrase-based systems such as Hierarchical Phrase-Based (HPB) SMT (Chiang, 2005) (cf. Section 2.4) and SMT with Syntactified Target Language Phrases (SPMT) system (Marcu et al., 2006) (cf. Section 2.5.2).

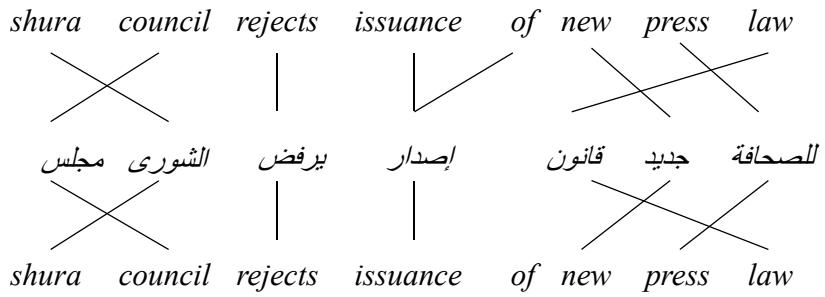

Figure 2.3: Source-to-target and target-to-source word alignments between an Arabic source sentence and its corresponding English target translation.

2.3.1 Phrase Extraction

Och and Ney (2003) propose an approach to extracting phrases in PB SMT from word alignments. As word alignments are asymmetric (cf. Section 2.2), the first step in the phrase extraction process is to calculate the alignments between the sentence pair in both source-to-target and target-to-source directions. Figure 2.3 shows word alignments in both directions for the sentence pair in Figure 2.1. We can see that the only difference between the alignments in different directions is that the word *of* is aligned to an Arabic word in the English-to-Arabic direction, whereas it is not aligned to any word in the other direction. After calculating the bidirectional alignments, the intersection and union of the alignment points in both directions are calculated. Figure 2.4 shows the extraction of the intersection and union alignment points from the alignment matrices of the sentence pair illustrated in Figure 2.3. The intersection presents a high-precision alignment whereas the union presents a high-recall alignment (Koehn et al., 2003). The phrase extraction process starts from the intersection alignment and uses it as a backbone to which alignment points from the union alignment are subsequently added according to some heuristics such as "diag" and "diag-and" (Koehn et al., 2003).

After applying the previous steps, phrase pairs which are consistent with word alignments are extracted. This means that words in each phrase are only aligned to each other and do not align to any other word outside the phrase. Figure 2.5 shows some valid phrase pairs extracted according to the previous steps from the sentence pair illustrated in Figure 2.3.

After extracting phrase pairs from the parallel corpus, the phrase translation probability distribution is calculated. Koehn et al. (2003) use relative frequency to calculate the probability of the target phrase $\bar{e}$ given the source phrase $\bar{f}$, which is called the direct translation probability, as in (2.3):

$$p(\bar{e}|\bar{f}) = \frac{count(\bar{e}, \bar{f})}{\sum_{\bar{e}} count(\bar{e}, \bar{f})} \tag{2.3}$$

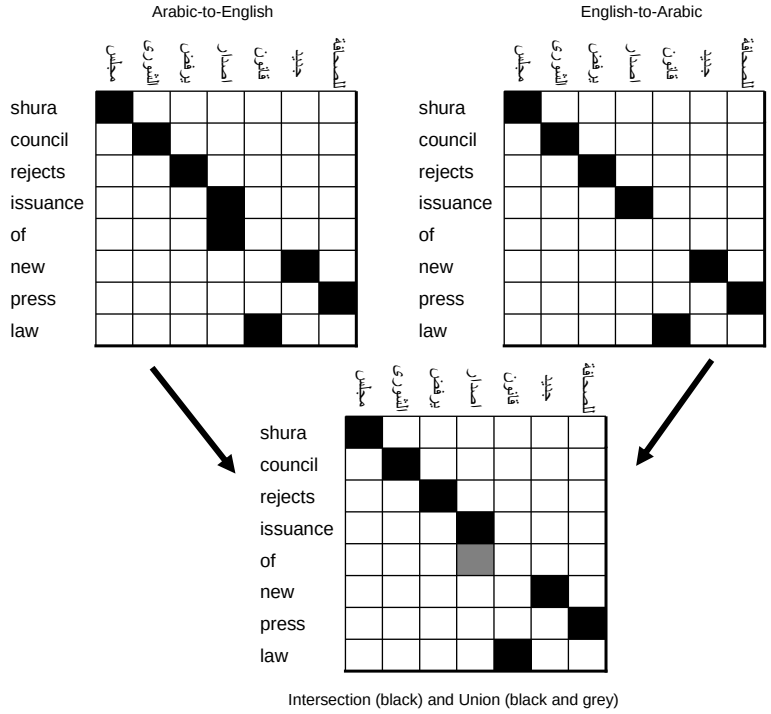

Figure 2.4: Extraction of the intersection and union alignment points from the alignment matrices of the sentence pair illustrated in Figure 2.3.

Koehn et al. (2003) show that extracting syntactic phrases which correspond to syntactic constituents only is harmful to the performance. It is worth mentioning here that the approaches we present in this thesis use CCG, which recognizes non-traditional syntactic constituents.

2.3.2 The Log-Linear Model

The log-linear model (also known as the maximum entropy model in the NLP literature) has been utilized for a number of NLP tasks such as POS tagging (Ratnaparkhi, 1996) and statistical parsing (Johnson et al., 1999), due to its ability to easily incorporate complex overlapping discriminative features in the model. In equation (2.1), the calculation of the translation probability is based on the the reversed translation model $p(f|e)$ and the LM $p(e)$ only. By contrast, the log-linear model of SMT allows more models to participate in calculating translation probability as in (2.4):

الشورى	shura	جديد للصحافة	new press
مجلس	council	قانون جديد للصحافة	new press law
قانون	law	مجلس الشورى يرفض	shura council rejects
جديد	new	يرفض اصدار	rejects issuance of
إصدار	issuance of	مجلس الشورى يرفض إصدار	shura council rejects issuance of
مجلس الشورى	shura council	إصدار قانون جديد للصحافة	issuance of new press law

Figure 2.5: A set of phrase pairs extracted according to word alignments from the sentence pair illustrated in Figure 2.3.

$$p(e|f) = \prod_i h_i(f,e)^{\lambda_i} \tag{2.4}$$

$$\log\, p(e|f) = \sum_i \lambda_i \log\, h_i(f,e)$$

where $h_i(f,e)$ represents the feature functions and λ_i are the weights of these feature functions. Feature functions typically used in PB SMT are (Koehn et al., 2003; Och and Ney, 2004):

- Direct phrase translation probability $p(\bar{e}|\bar{f})$ calculated according to equation (2.3).
- Inverse phrase translation probability $p(\bar{f}|\bar{e})$.
- The LM $p(e)$.
- Direct lexical weighting $p_w(\bar{e}|\bar{f})$.
- Inverse lexical weighting $p_w(\bar{f}|\bar{e})$.
- Word/Phrase penalty which is the number of words/phrases in the target translation (cf. Koehn (2010) page 140).

- Relative distortion probability $d(a_i - b_{i-1})$ where a_i is the start position of the source phrase f_i which is translated into the target phrase e_i, and b_{i-1} is the end position of the source phrase f_{i-1} which is translated into the target phrase e_{i-1}.

- Lexicalised reordering model $p(orientation|\bar{e}, \bar{f})$ which learns how likely each orientation of the current phrase pair (monotone, swap and discontinuous) is with respect to the previous phrase pair (Koehn et al., 2005).

Each of these features try to improve one aspect affecting the final translation quality. Phrase translation probabilities reflect how well the target and source phrases translate to each other. The LM feature helps to ensure the fluency of the produced translation. The lexical weighting features examine how well the words of the source and target phrases translate to each other. The Word/Phrase penalty helps to avoid the tendency of PB systems to produce short translations. Relative distortion probability enables the phrase reordering range to be limited, because large phrase movement have been demonstrated to negatively affect the translation performance (Koehn et al., 2005). Lexicalised reordering model provides more sophisticated reordering than the distortion probability by encouraging reordering patterns based on lexical evidences from the source and target phrases. We use the aforementioned features in the log-linear model of the HPB SMT systems we build in our experiments, except for the relative distortion probability and lexicalised reordering model features which are specific to the PB SMT systems.

2.3.3 Minimum Error Rate Training

Minimum Error Rate Training (MERT) (Och, 2003) is an algorithm for finding the log-linear weights λ which achieve the best translation quality in terms of an automatic MT evaluation metric on a training corpus as in (2.5):

$$\hat{\lambda} = \underset{\lambda}{\operatorname{argmax}} \, BLEU(\hat{E}, R) \tag{2.5}$$

$$\begin{aligned}\hat{E} &= \text{argmax}_E \log p(E|F) \\ &= \text{argmax}_E \sum_{(e,f)} \log p(e|f) \\ &= \text{argmax}_E \sum_{(e,f)} \sum_i \lambda_i h_i(f,e)\end{aligned} \tag{2.6}$$

where F and E represent the source and target side of the data used to train the log-linear model, respectively. In equation (2.5), the BLEU (Papineni et al., 2002) automatic MT evaluation metric is used to evaluate the best translation $\hat{E}$ using the reference set R. However, optimization can be performed with respect to other automatic MT evaluation metrics. Och (2003) approximates the maximization in equation (2.6) by calculating the n-best candidate translations of the source training data F. Thus, the optimization of the log-linear model weights λ is performed according to the following steps:

1. Set $\hat{\lambda}$ to manually defined values.
2. Find the n-best translations for F using the current values of $\hat{\lambda}$.
3. Combine the current n-best translations with the previous ones. If the n-best translations do not change then stop.
4. Calculate the new values for $\hat{\lambda}$ based on the n-best translations according to Equation (2.5), then go to step 2.

The algorithm is guaranteed to converge as, in the worst case, the set of n-best translations would contain all possible translations. MERT works well for a small number of features (typically less than 10 features). However, its performance degrade when using it with a large number of features.

2.3.4 Decoding

The basic PB SMT translation algorithm (Koehn, 2004a) tries to translate a source sentence f by first segmenting it into all possible continuous word segments. Then, different translation options for each segment are explored with the possibility of

translating the segments out of order. Each of the partial translation states (or 'hypotheses') constructed during decoding has a number of properties which include its probability, the words it covers from the source sentence, the target translation of the covered segment and a pointer to the previous hypothesis. The probability of a hypothesis is the probability of translating its source segment multiplied by the probability of its previous hypothesis. The algorithm continues to produce hypotheses until all the words of the source sentence have been covered, forming a connected search graph. Finally, the 'best' translation (i.e. the translation with the highest probability) is retrieved from the search graph.

The number of hypotheses generated according to this algorithm is exponential in the source sentence length, which makes MT an NP-complete problem (Knight, 1999). This means that conducting an exhaustive search for the best translation is infeasible. Consequently, a set of heuristic-based techniques are applied in order to reduce the size of the search space and guide the search process. A beam search algorithm, which was originally developed for speech recognition (Jelinek, 1997), is used to achieve this goal. The beam search algorithm for SMT (Koehn, 2004a) uses a number of stacks which act as priority queues. Each stack contains hypotheses which cover the same number of source words sorted according to their probability. Whenever a hypothesis is expanded, the new hypothesis is placed in the appropriate stack. If the newly added hypothesis falls outside the beam of its stack, it will be pruned away. Two types of pruning can be applied: histogram pruning and threshold pruning (Koehn, 2004a). Histogram pruning keeps a fixed number of hypotheses whereas threshold pruning prunes hypotheses whose probability is less than the best hypothesis probability by a certain factor. The complexity of the beam search algorithm for PB SMT decoding is quadratic with the sentence length and linear with the stack size (Koehn et al., 2003).

2.3.5 Rescoring

After generating the n-best translations by the decoder, they can be rescored using models which are either hard or expensive to be incorporated during the n-best generation such as a large-order LM and sentence-level features. Rescoring also helps to test the addition of new features in a fast and easy way without the need to reimplement the search algorithm for each feature.

2.4 Hierarchical Phrase-Based SMT

Hierarchical Phrase-Based (HPB) SMT (Chiang, 2005, 2007) is a tree-based SMT model which extracts a synchronous Context Free Grammar (SCFG) (Lewis and Stearns, 1968) from a parallel corpus without using syntactic annotation. SCFGs are a generalization of inversion transduction grammars (Wu, 1997). SCFGs have the ability to generate pairs of strings at the same time instead of a single string as in the normal CFG. In HPB SMT, the synchronous CFG is used to parse the source sentence while generating the target translation. Synchronous CFG rules used by HPB SMT are called hierarchical rules or hierarchical phrases, which are the basic translation unit used by the HPB SMT model.

HPB SMT was designed to build upon the strengths of PB SMT. Continuous phrase pairs extracted in PB SMT form the basis for hierarchical rule extraction. Thus, the HPB SMT grammar combines the ability of continuous phrases to translate expressions and perform word reordering with the ability of HPB SMT hierarchical rules to translate discontinuous phrases and learn phrase reordering. In addition, the HPB SMT framework uses the log-linear model to combine the various component models which participate in the calculation of translation probability (cf. Section 2.3.2), and performs tuning through Minimum Error Rate Training just like the PB SMT framework. Furthermore, HPB SMT uses the beam search decoding algorithm used in PB SMT decoding (cf. Section 2.3.4) combined with chart decoding in addition to rescoring techniques originally developed for PB SMT (cf.

Section 2.3.5).

The pipeline of the HPB SMT system shares many similarities with the pipeline of the PB SMT system illustrated in Figure 2.2. The shaded stages in this figure denote the shared stages between HPB SMT and PB SMT. The HPB SMT system needs an extra stage after phrase extraction during training, which is rule extraction. Another difference between HPB SMT and PB SMT is the decoding algorithm used. HPB SMT uses chart decoding (c.f. Section 2.4.3) but it combines it with the beam search pruning used by PB SMT to prune the search space during decoding.

2.4.1 HPB SMT Grammar

The HPB SMT grammar uses two types of rules: hierarchical rules and glue grammar rules. In the following we provide the definition, extraction and function of each of the hierarchical rules and glue grammar rules.

Hierarchical Rules

Hierarchical rules are rewrite rules with aligned pairs of right-hand sides, taking the following form:

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

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. α represents the source side of the hierarchical rule while β represents the target side of the hierarchical rule. Hierarchical rules are extracted from a word-aligned parallel corpus $< F, E, A >$, where F and E are the source and target sides of the parallel corpus, respectively, and A is the bidirectional alignment between source and target sentences in the parallel corpus. The hierarchical rule extraction starts by extracting continuous phrases (called 'initial phrases') according to the approach of Och and Ney (2003) (cf. Section 2.3.1). Afterwards,

LHS	α	β
X	جديد X	*new* X
X	قانون جديد X	*new X law*
X	قانون X للصحافة	X *press law*
X	مجلس X	X *council*
X, S	مجلس الشورى يرفض X	*shura council rejects* X
X, S	X_1 يرفض X_2	X_1 *rejects* X_2
S	مجلس الشورى X_1 إصدار X_2 جديد للصحافة	*shura council* X_1 *issuance of new press* X_2
S	مجلس الشورى X_1 إصدار قانون X_2 للصحافة	*shura council* X_1 *issuance of* X_2 *press law*
X, S	X_1 يرفض إصدار X_2	X_1 *rejects issuance of* X_2

Figure 2.6: A set of hierarchical rules extracted from the sentence pair illustrated in Figure 2.3.

hierarchical rules are extracted by recursively subtracting initial phrase pairs from bigger initial phrase pairs and replacing the subtracted phrase pairs with the nonterminal symbol X, which denotes a gap. More than one phrase pair can be subtracted simultaneously from its container phrase pair, leading to more than one nonterminal on each side of the hierarchical rule. In this case, co-indexation is used to align source-side nonterminals with target-side nonterminals. The initial rules extracted from the sentence pair in Figure 2.3 are the same as PB SMT phrases illustrated in Figure 2.5. Figure 2.6 shows a set of hierarchical phrases extracted from these initial rules. The rule whose target side is X *council* captures the inversion of the nouns in addition construction between Arabic and English. Note also that the rule whose target side is X *press law* captures the differences in reordering of adjectives and nouns between Arabic and English.

Extracting hierarchical rules according to the aforementioned approach produces a large number of rules. Thus, Chiang (2005) proposes a set of techniques which limit the size of the extracted grammar:

- The length of initial phrases is limited to 10 words on the source side. The number of words and nonterminals in the source side is limited to 5.

- Adjacent nonterminals are prohibited in the source side of the rule in order to

avoid superious ambiguity.

- The number of nonterminals on each side of the rule is limited to 2.
- Unaligned words are prohibited at the boundaries of the initial phrase.
- The rule should have at least one pair of aligned words (e.g. every rule must contain at least one terminal symbol).

In contrast to PB SMT, HPB SMT does not need a separate phrase reordering model given the availability of hierarchical phrases, which capture highly lexicalized phrase reordering, whereas continuous phrases only capture word reordering. Initial rules, which are identical to phrase pairs used in PB SMT, give HPB SMT the power of continuous phrases too. Furthermore, hierarchical rules enable the translation of discontinuous phrases. This is important to capture many linguistic phenomena which are not directly (or even impossible to be) captured by PB SMT. For example, negation in French consists of two words *ne* and *pas* between which the negated verb lies. Thus, one of the rules which translate a negated verb in French to English has the form as in (2.7):

$$X \rightarrow < \ ne\ X\ pas\ ,\ does\ not\ X\ > \tag{2.7}$$

During translation, the nonterminal on the source side is replaced with the French verb and the nonterminal on the target side is replaced with the corresponding English verb translation. However, PB SMT cannot directly model this translation pattern and needs to divide the negation expression into two parts and then translate each part separately, which might lead to a poor translation of the French negation expression.

Glue Grammar Rules

Glue grammar rules perform monotone phrase concatenation, which means that they combine target phrases together without performing any reordering. They consist

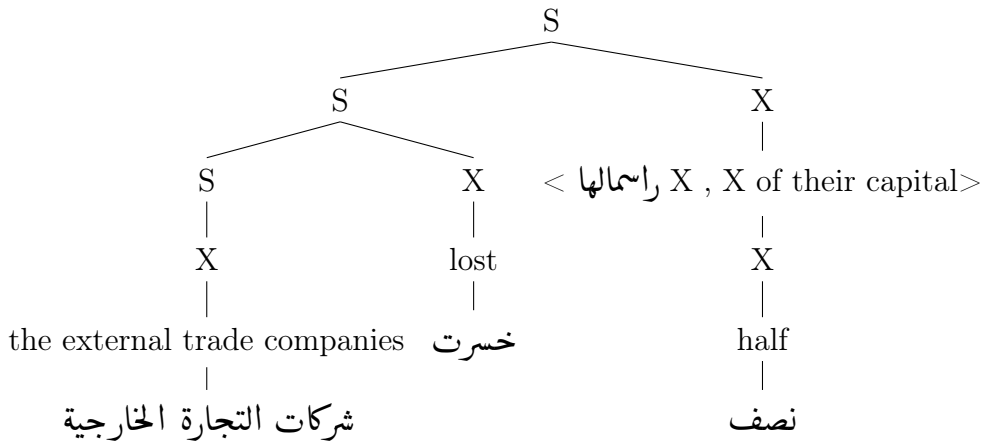

Figure 2.7: Derivation tree of the English translation produced by the HPB SMT system for the Arabic sentence شركات التجارة الخارجية خسرت نصف راسمالها.

of the following two rules as in (2.8) and (2.9):

$$S \rightarrow <\ S\ X\ ,\ S\ X\ > \tag{2.8}$$

$$S \rightarrow <\ X\ ,\ X\ > \tag{2.9}$$

The main role of the rule (2.8) is to produce translation by concatenating target subphrases. The rule (2.9) initiates glue grammar tree building by transforming the nonterminal X to the symbol S to be later used in rule (2.8). Glue grammar helps to avoid producing an empty translation which results from hierarchical rules sparsity. Furthermore, glue grammar rules are used to reduce the complexity of chart decoding by limiting the application of the hierarchical rules to a certain threshold above which only glue grammar rules are applied. This has the same effect as limiting reordering in PB SMT. Glue grammar rules can also be applied below this limit but their application cannot alternate with hierarchical rules, and they always form a left-balanced binary tree on top of the hierarchical rules in the derivation tree as illustrated in Figure 2.7.

2.4.2 Training

For each rule $X \rightarrow < \alpha, \beta >$, HPB SMT uses features originally developed in PB SMT to score the rule (cf. Section 2.3.2). These features include the direct and inverse phrase translation probabilities $p(\alpha|\beta)$ and $p(\beta|\alpha)$, direct and inverse lexical weighting probabilities $p_w(\alpha|\beta))$ and $p_w(\beta|\alpha)$, and word/phrase penalty. These features are combined using the log-linear model (cf. Section 2.3.2) and their weights are tuned using Minimum Error Rate Training (Och, 2003) (cf. Section 2.3.3).

Each target sentence might be derived from the source sentence in many possible ways. Thus, in order to calculate the phrase translation and lexical weighting probabilities for each rule in the grammar, we need to know all possible derivations of each target sentence in the corpus. However, we only have the observed sentences and derivations are a hidden variable. That is why heuristics are used to calculate these probabilities. These heuristics assign each initial phrase pair occurrence a count equals to one. Then, this count is distributed equally among all the rules extracted from this initial phrase pair. Finally, phrase translation probabilities are calculated according to relative-frequency estimation.

2.4.3 Decoding

Translation in HPB SMT is a parsing-generation process. During this process, the source side of the sentence to be translated is parsed using the source side of the hierarchical rules. At the same time, the target translation is generated simultaneously using the target side of the hierarchical rules. The decoding process is continued until the whole source sentence is parsed. There are many possible ways to parse the source sentence. This leads to a derivation forest which produces multiple candidate translations. At the end of the translation process, the 'best' translation $\hat{\mathbf{e}}$ of these candidate translations is the one which maximizes the probability of the derivation D as in (2.10):

$$\hat{\mathbf{e}} = \underset{D:f(D)=f}{\operatorname{argmax}}\, p(D) \tag{2.10}$$

where the probability of the derivation D is calculated according to the log-linear model as in (2.11):

$$p(D) = \prod_{(X \rightarrow <\alpha,\beta>) \in D} \prod_{i} \lambda_i h_i(X \rightarrow <\alpha, \beta>) \tag{2.11}$$

In order to perform efficient translation with the HPB SMT model, a chart parsing algorithm is used to perform decoding. Chart parsing (Kay, 1980) is a dynamic programming-based approach which uses a data structure called the chart. Chart parsing helps to cut parsing computational costs by reusing previously parsed segments stored in the chart instead of re-parsing them. Various chart parsing algorithms have been used to perform decoding in HPB SMT. CKY+ (Chappelier and Rajman, 1998), which is a variant of the CKY (Cocke-Kasami-Younger) algorithm, has been used in several HPB decoders (Zollmann and Venugopal, 2006; Hoang et al., 2009a; Li et al., 2009; Stein et al., 2011). Other HPB approaches (Watanabe et al., 2006; Cai et al., 2009) used Earley parsing (Earley, 1970) to perform HPB decoding.

During the chart decoding process, the parser uses translation rules as inference rules to prove weighted chart items. The process is continued until the parser reaches the goal which is the S (sentence) symbol at the root of the chart. Chart items take the form $[X, i, j]$, which indicates a subtree spanning the source words i to j and its root has the X symbol as a label. Each chart item $[X, i, j]$ has a weight w, which is the multiplication of the weight of the rule $R : X \rightarrow <\alpha, \beta>$ which produced $[X, i, j]$ with the weights of the subitems which substituted the nonterminals in the rule R to produce $[X, i, j]$. The goal item in this case is $[S, 0, n]$, where n is the length of the source sentence f. The complexity of this decoding process is $\mathcal{O}(n^3)$.

Incorporating the LM in scoring chart items during HPB SMT decoding further complicates the process. To calculate the LM feature of order m for a sequence

of words, the m-1 context words at the start and end of the sequence should be known. Thus, representing chart items as $[X, i, j]$ is not sufficient to calculate the LM feature during decoding. Instead, chart items should take into account the m-1 target boundary words at each side, which are necessary to calculate the LM feature for future items. Therefore, chart items should take the form $[X, i, j, e]$, where e is the part of the target string produced by the subtree spanned by the item which is necessary to calculate the LM feature for future items. However, incorporating e in chart items leads to an explosion of the number of different chart items produced by the parser, increasing decoding complexity as in (2.12):

$$\mathcal{O}\left(n^3 \left[|T|^{2(m-1)}\right]^K\right) \tag{2.12}$$

where $|T|$ is the number of English terminal symbols and K is the number of non-terminals permitted on each side of the rule. This makes the decoding process intractable (Chiang, 2007), and necessitates the application of pruning techniques to reduce the size of the explored search space.

One other possible way of incorporating the LM in HPB SMT decoding is to use it to rescore the n-best list of translations. However, Chiang (2007) argues that the size of the n-best list should be extremely high in this case, because the number of different translations is exponential in the length of the source sentence. Chiang (2007) proposes a better solution which is called cube pruning. Cube pruning is a compromise between rescoring the n-best translations using the LM, and incorporating the LM in scoring the chart items during decoding. It imposes a beam limit on the number of chart items added to a cell in the chart, and calculates LM score-augmented chart items which fall inside the beam only. Cube pruning for a consequent chart cell starts by sorting its antecedent chart items by their score including the LM score. Afterwards, rules are applied on antecedent chart items, resulting in consequent chart items, which are then added to the consequent chart cell. Rule application is continued until a consequent chart item falls outside the

beam of the consequent chart cell.

2.4.4 HPB SMT vs. PB SMT

The HPB SMT system has been demonstrated as being capable of beating PB systems in large-scale evaluations (Chiang, 2005). Several studies have been conducted to investigate the reasons behind the improvement which the HPB SMT system has over the PB SMT system. Lopez (2008) argues that HPB SMT advantages over PB SMT lie in its ability to perform better lexicalized reordering and translate discontinuous phrases. He also suggests that the performance of the HPB SMT system can be further improved using a better parameterization through specialized features similar to the ones used by lexicalized reordering models used in PB SMT (Koehn et al., 2005). Zollmann et al. (2008) compare the performance of PB and HPB systems on Chinese-to-English and Arabic-to-English translation under different LM size conditions. They found that for Chinese-to-English translation, the HPB SMT system maintained an improvement over the PB SMT system even under large LM condition, which helps the PB reordering model. However, for Arabic-to-English translation, the HPB SMT system did not achieve a consistent improvement over the PB SMT system. They conclude that the HPB SMT system outperforms the PB SMT system for language pairs which are sufficiently non-montonic (i.e. they need long reordering ranges) such as Chinese–English, whereas both systems perform equally on language pairs which have weaker reordering phenomena, such as Arabic–English. In terms of expressiveness, Zollmann et al. (2008) found that the PB SMT system was not able to produce 22% of the translations generated by the HPB SMT system for the Chinese–English NIST MT06 test set using forced translation (i.e. forcing the system to generate only hypotheses which are consistent with a specific output), which indicates that HPB SMT has a better expressiveness thanks to its ability to translate discontinuous phrases. Auli et al. (2009) extend the work of Zollmann et al. (2008) by exploring the mutual reachability of the PB and HPB SMT systems to the 1-best translation generated by the other system.

They show that most of the translations generated by the HPB SMT system which was not reachable by the PB SMT system were due to the stronger ability of hierarchical rules to model word deletion during translation. Furthermore, their study shows that most of the translations generated by the PB SMT system which was not reachable by the HPB SMT system were due to the ability of the PB SMT system to model reordering of phrases without lexical evidence, which is not possible in the HPB SMT model. Despite these structural differences between the HPB and PB SMT models, Auli et al. (2009) demonstrate that there is a high overlap in the search spaces of the PB and HPB SMT systems. They conclude that the difference in performance between the HPB and PB SMT systems is due to different parameterization performed by each system.

Despite HPB SMT's more powerful reordering and better expressiveness compared to PB SMT, the latter still has a smaller translation model and a less complex decoding process, which enables a faster and less resource-demanding translation.

2.5 Syntax Augmentation for SMT Systems

Since the emergence of the first SMT models, many approaches have been devised to provide these models with syntactic knowledge. The evolution of SMT models has been accompanied with a comparable development of SMT syntax augmentation methods which tried to integrate syntax into the source side, the target side or both sides of the translation process. Such SMT syntax augmentation methods vary in complexity and expressiveness from using simple syntactic descriptions applied at the word or phrase level, to more complex trees which represent multilevel syntactic structures. These methods also explored the integration of syntax extracted using different grammar formalisms such as context free phrase structure grammar, categorial grammar and dependency grammar into SMT systems.

The main goal behind incorporating syntax in the source side of SMT systems is to improve the adequacy of translation, which indicates the extent to which the

meaning of the source sentence is preserved by its translation. Source-side syntax helps to disambiguate the meaning of words/phrases in the input according to their syntactic role (Carpuat and Wu, 2007; Stroppa et al., 2007; Haque et al., 2009). Furthermore, using syntax in the source side helps to perform more complex linguistically motivated reordering guided by source-side syntactic information. Systems which use syntactic structures in the source side are sometimes called tree-to-string models (Huang et al., 2006; Liu et al., 2006). Incorporating syntactic information in the target side of SMT systems helps to improve the fluency (i.e. grammaticality) of translation output. Target-side syntax helps to ensure that the translation output conforms to the target-language grammar. Systems which use syntactic structures in the target side are sometimes called string-to-tree models (Galley et al., 2006). Recently, a number of SMT systems have tried to improve both translation adequacy and fluency by incorporating both source- and target-side syntactic information. These systems try to define a set of linguistically motivated rules which transform the source syntactic structure into a target syntactic structure, hence they are called tree-to-tree models (Zhang et al., 2008; Liu et al., 2009; Chiang, 2010).

2.5.1 Source Syntax Augmentation for SMT Systems

Quirk et al. (2005) use source-side dependency parse trees to extract dependency treelet translation pairs, which are arbitrary connected subgraphs of the dependency tree, and a tree-based reordering model. In contrast to continuous phrases extracted in PB SMT, dependency treelets are able to translate discontinuous phrases. In addition, dependency treelets allow the application of more powerful, linguistically motivated reordering models. Liu et al. (2006) define a translation model based on tree-to-string alignment templates, which are automatically extracted from the source side of a parsed parallel corpus. Alignment templates produce terminals and nonterminals and define how to transform a source-side parse tree into a target string. Huang et al. (2006) define a syntax-driven translation model which extracts transformation rules based on extended tree-to-string transducers that have multi-

level trees on the source side. The translation starts by parsing the source sentence. Then, transformation rules are used to recursively generate the target string from the source analysis. Mylonakis and Sima'an (2011) define a joint probability translation model which extracts a binary synchronous context free grammar from a parallel corpus. Nonterminals in the extracted grammar are labelled with source-side syntax-based labels similar to the ones used by Zollmann and Venugopal (2006). This model learns the latent translation structure separately from the emission of translation output, which helps to capture complex reordering patterns.

2.5.2 Target Syntax Augmentation for SMT Systems

One of the first attempts to incorporate syntax in the earliest noisy channel MT model is the work of Yamada and Knight (2001). They describe a translation model which modifies the noisy channel model to accept parse trees rather than words as an input. The channel produces the output string by performing a set of operations at each node of the input tree. These operations, namely word translation, reordering and insertion, try to model the linguistic differences between translated languages such as case and word order. The decoder based on this translation model (Yamada and Knight, 2002) searches for the most probable parse tree in the output language given a string of the foreign language in a way similar to parsing a sentence.

While the LM used in the decoder of Yamada and Knight (2002) is a regular n-gram-based LM, Charniak et al. (2003) accompany the translation model of Yamada and Knight (2001) with a syntax-based LM proposed by Charniak (2001). This LM searches for the most probable parse tree in a parse forest built for the translation output by the decoder. Schwartz et al. (2011) provide a formal definition of an incremental syntax-based LM which is incorporated into the PB SMT model. The LM works in a left-to-right fashion which is the same mechanism used in PB SMT decoding.

Galley et al. (2004) define a method under which the minimum set of syntax-based transformation rules which explain the parallel data are extracted. These

rules are extracted automatically from source strings aligned to target parse trees. Galley et al. (2004) demonstrate that the minimal linguistically motivated transformation rules learnt according to this method are capable of fully explaining the data in the parallel corpus and of capturing complex translation patterns beyond local child node reordering suggested by previous models (Yamada and Knight, 2001). Galley et al. (2006) extend the method of Galley et al. (2004) by adopting multiple derivations instead of a single one in the rule extraction process, in addition to accounting for multiple interpretations for unaligned words. They found that using rules with a larger context helps to improve the performance over using minimal rules only.

The SPMT translation approach (Marcu et al., 2006) extracts minimal transformation rules under extended tree-to-string (**xRs**) transducers (Knight and Graehl, 2005). The SPMT model utilises feature functions originally developed under the PB SMT model (Koehn et al., 2003) and combines them using the log-linear model (Och, 2003). The difference between the transformation rules extracted by SPMT and the ones extracted by the model proposed by Galley et al. (2004) is that SPMT rules do not allow discontinuous phrases on the source side, while the rules in the model of Galley et al. (2004) are able to translate discontinuous phrases, which gives them a larger expressive power than SPMT rules (Marcu et al., 2006). In addition, SPMT extracts minimal transformation rules with respect to the statistically extracted source phrase compared to the whole source sentence in Galley et al. (2004).

String-to-dependency MT approaches take advantage of target-side dependency-based syntactic information to capture syntactic relations between distant words and provide better coverage for phrases which do not correspond to syntactic constituents. Shen et al. (2008) use dependency grammar in the target side of the translation process. They extract hierarchical string-to-tree translation rules whose target sides are well-formed dependency structures. They show that the flexibility of dependency structures compared to phrase structure grammar helps to provide a better coverage for non-constituent rules. While Shen et al. (2008) exclude phrases

which correspond to malformed dependency structures from participating in the translation process, Stein et al. (2010) do not exclude these phrases. They instead use a binary feature to mark the phrases which correspond to well-formed dependency structures. In addition, they use a feature in the log-linear model to judge the validity of the dependency structure that results from a phrase replacing a nonterminal in a hierarchical rule. A dependency-based LM is also used to rerank the n-best translations. Peter et al. (2011) extends the work of Stein et al. (2010) by exploring the effect of integrating the dependency-based LM in the decoding process, which was demonstrated to achieve better performance than using the dependency-based LM in rescoring the n-best translations only.

2.5.3 Source and Target Syntax Augmentation for SMT Systems

Zhang et al. (2008) propose a tree sequence alignment-based translation model. A tree sequence (Liu et al., 2007) is an ordered sequence of tree fragments covering a phrase. Using tree sequences in rule extraction gives more flexibility and thus helps to increase rule coverage. Furthermore, tree sequences enable complex reordering patterns to be captured and discontinuous phrases to be translated while at the same time being able to handle phrases which do not correspond to syntactic constituents. Liu et al. (2009) argue that using the 1-best parse for rule extraction in syntax-based SMT systems – especially tree-to-tree systems – is one of the main challenges facing these systems. They believe that using the 1-best parse only leads to the extraction of noisy translation rules, which damages the performance of tree-to-tree systems. As a solution, they propose to use a packed forest on both the source and the target sides of the parallel corpus to extract translation rules based on synchronous tree substitution grammar (Eisner, 2003), which helped to achieve a significant improvement over conventional tree-based models. Chiang (2010) tries to tackle problems affecting the performance of tree-to-tree translation models, which are limited rule

coverage and strong syntactic constraints imposed by the model. He tries to increase rule coverage by using fuzzy hierarchical rule extraction similar to the approaches used by Zollmann and Venugopal (2006) and Zhang et al. (2008). This rule extraction approach provided full coverage for extracted phrases in the translation model. Chiang (2010) also tries to improve the performance of the tree-to-tree system by using soft syntactic constraints in the model, which allows mismatches between rule nonterminal labels and the labels of phrases substituting them. He incorporates a set of syntax-based source and target features into the log-linear model to measure how unlabelled derivations used during decoding conform with the syntactic constraints learnt from syntactically annotated training data. This gives the model more flexibility to decide how and when to apply these constraints.

2.6 Syntax Augmented Machine Translation

HPB SMT tries to model one aspect of language syntax, which is the hierarchical structures of the language. The HPB SMT rules are extracted from the parallel corpus according to word alignments without using any syntactic annotation. Thus, the lack of syntactic knowledge in the HPB SMT model may lead to the production of ungrammatical translations.

Syntax Augmented Machine Translation (SAMT) (Zollmann and Venugopal, 2006) provides the HPB SMT model with syntactic knowledge extracted from context free phrase structure grammar-based parse trees of target-side sentences in the training corpus. This knowledge is represented as syntactic labels attached to nonterminals and left-hand sides of hierarchical rules. These labels act as syntactic constraints which restrict nonterminal replacement during decoding to only those phrases bearing the same syntactic label as that of the nonterminal. In this way, the syntactic functions of the phrases will direct the decoding process.

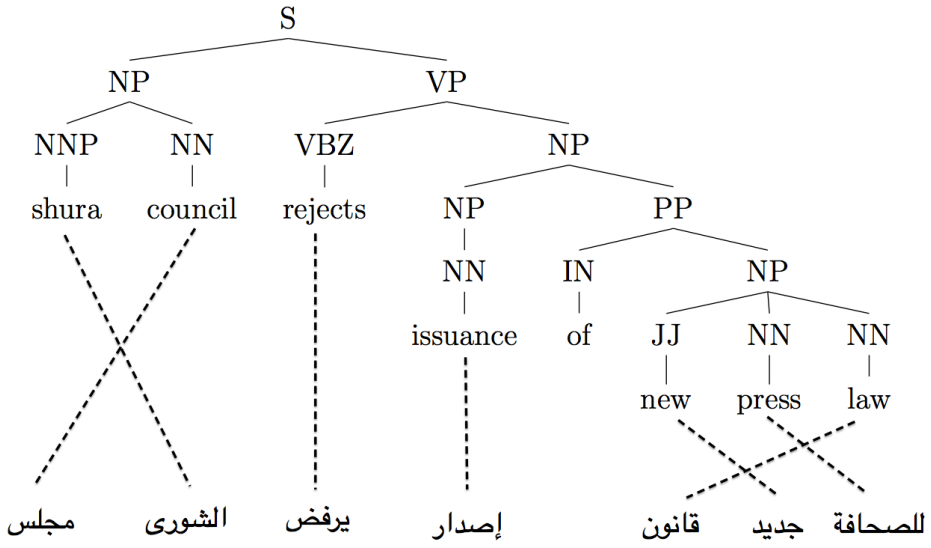

Figure 2.8: Parse tree of the English sentence in Figure 2.1 along with its aligned Arabic words.

2.6.1 Rule Extraction

SAMT rules are extracted according to the following steps:

- Each sentence in the target side is assigned a context free phrase structure grammar-based parse tree. Figure 2.8 shows the parse tree of the English sentence in Figure 2.1 along with its aligned Arabic words.

- Phrase pairs are extracted from the parallel corpus according to the PB SMT phrase extraction method (cf. Section 2.3.1).

- A syntactic label is assigned to each of the previously extracted phrase pairs. This syntactic label corresponds to the syntactic constituent in the parse tree that covers the target phrase. In case the target phrase does not fully span a constituent in the parse tree, the phrase is assigned an extended category according to the following cases:

  - C1+C2 if the target side of the phrase pair spans two adjacent syntactic constituents C1 and C2. In Figure 2.8, the phrase *new press* spans two

adjacent constituents: *JJ* and *NN*. Thus, the label assigned to this phrase is *JJ+NN*.

- C1/C2 if the phrase spans a part of a syntactic constituent C1 missing a constituent C2 to its right. In Figure 2.8, the phrase *shura council rejects* spans a sentence *S* missing a noun phrase *NP* to its right, which corresponds to the phrase *issuance of new press law*. Therefore, the phrase *shura council rejects* is assigned *S/NP* as a syntactic label.
- C1\C2 if the phrase spans a part of a syntactic constituent C1 missing a constituent C2 to its left.
- If the phrase does not correspond to any of the previous cases, it is assigned a general symbol X, which means that there is no syntactic constraint imposed on the phrase. Nonterminals holding the X label do not impose any syntactic constraint on the phrases replacing them, allowing for any phrase to replace the nonterminal no matter what syntactic label it holds.

Note that the forward and backward slash operator used in SAMT emulate the forward and backward operators in CCG categories (cf. Section 2.7.1).

- Hierarchical rules are extracted from the syntactically annotated phrases according to the hierarchical rule extraction algorithm (Chiang, 2005) (cf. Section 2.4.1).

Figure 2.9 shows a set of initial rules extracted from the sentence pair illustrated in Figure 2.8 along with the syntactic labels assigned to their left-hand sides. Figure 2.10 shows some of the SAMT hierarchical rules extracted from these initial rules.

SAMT uses the same glue grammar rules as HPB SMT (cf. Section 2.4.1), which means that phrases are concatenated regardless of their nonterminal labels during glue grammar rule application.

LHS	*α*	*β*
NNP	الشورى	*shura*
NN	مجلس	*council*
NN	قانون	*law*
JJ	جديد	*new*
NP/NP	إصدار	*issuance of*
NP	مجلس الشورى	*shura council*
JJ+NN	جديد للصحافة	*new press*
NP	قانون جديد للصحافة	*new press law*
S/NP	مجلس الشورى يرفض	*shura council rejects*
VP/NP	يرفض إصدار	*rejects issuance of*
S/NP	مجلس الشورى يرفض إصدار	*shura council rejects issuance of*
NP	إصدار قانون جديد للصحافة	*issuance of new press law*

Figure 2.9: A set of SAMT initial rules extracted from the Arabic–English example in Figure 2.8.

2.6.2 Decoding

SAMT uses the same chart decoding algorithm as HPB SMT (cf. Section 2.4.3). Attaching labels to nonterminals blows up the size of the grammar, as each unlabelled HPB SMT rule would have many differently labelled counterparts. During decoding, a chart item $[X, i, j, e]$ in HPB SMT would have multiple corresponding chart items, each of which has a different left-hand-side label instead of the general X label. This further complicates the decoding process, raising its complexity when incorporating the LM as in (2.13):

$$\mathcal{O}\left(n^3 \left[|N||T|^{2(m-1)}\right]^K\right) \tag{2.13}$$

where $|N|$ is the number of different nonterminal symbols used by the system, which is equal to one in equation (2.12).

LHS	α	β
NP	X قانون جديد	new NN law
NP	قانون X للصحافة	JJ press law
NP	مجلس X	NNP council
S	مجلس الشورى يرفض X	shura council rejects NP
S/NP	مجلس الشورى يرفض X	shura council rejects NP/NP
S	X1 يرفض X2	NP1 rejects NP2
S	مجلس الشورى X1 إصدار X2 جديد للصحافة	shura council VBZ1 issuance of new press NN2
S	مجلس الشورى X1 إصدارقانون X2 للصحافة	shura council VBZ1 issuance of JJ2 press law
S	X1 يرفض إصدار X2	NP1 rejects issuance of NP2

Figure 2.10: A set of SAMT hierarchical rules extracted from the Arabic–English example in Figure 2.8.

2.7 Combinatory Categorial Grammar

Combinatory Categorial Grammar (CCG) (Steedman, 2000) is a grammar formalism based on Categorial Grammar (Ajdukiewicz, 1935; Bar-Hillel, 1953). Most of the language grammar in CCG is stored in the lexicon, in contrast with context free phrase structure grammar, which expresses the grammar of the language as context free production rules (Chomsky, 1957). The CCG lexicon contains words paired with rich syntactic categories called "supertags". CCG uses a small set of simple combinatory rules to combine CCG categories during parsing. These rules allow the construction of free derivation structures and non-standard constituents, which enables CCG to handle grammatical phenomena such as coordination and long-range dependencies quite naturally. Another important feature of CCG is its transparent interface between syntax and compositional semantics. CCG lexical categories and rules have a well-defined semantic interpretation, which enables a semantic representations to be built directly during parsing. An English CCG grammar is extracted from the CCGbank which is a corpus of CCG derivations translated semi-automatically from the Penn Treebank (Hockenmaier, 2003; Hockenmaier and Steedman, 2007), and augmented with local and long-range word-word dependen-

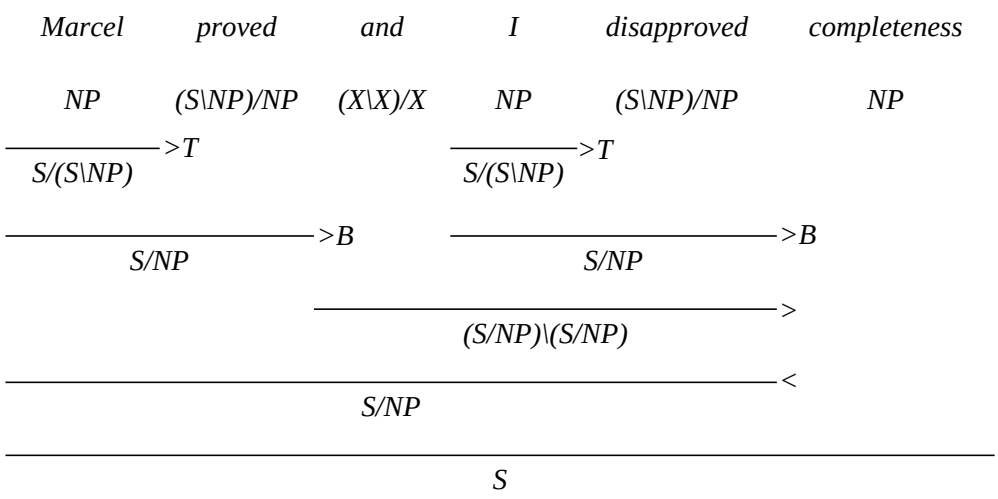

Figure 2.11: CCG parse tree of the sentence *Marcel proved and I disproved completeness.*

cies. At the time of writing of this thesis, a CCGbank and a CCG parser (Clark and Curran, 2007) were fully developed for English only. That is why all our experiments focus on translation into English. It is worth noting that the approaches presented in this thesis are not limited to translation into English; it can be applied on translation into any language for which a CCG parser has been developed. Currently, approaches to extracting CCG treebanks from context free phrase structure grammar-based treebanks are being developed for other languages (Hockenmaier, 2006; Boxwell and Brew, 2010; Tse and Curran, 2010). Figure 2.11 illustrates a CCG derivation tree assigned to the English sentence *Marcel proved and I disproved completeness.* The supertags assigned to each word in the sentence represent the first level of the CCG derivation tree.

2.7.1 CCG Categories

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 and the type of their result. Complex categories come in the following formats:

- X\Y is a functor 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).

Representing CCG categories as functors and arguments directly reflects the dependents and local context of the word or the phrase. For example, the lexical category of the verb *read* in the sentence *I read* is S\NP, which means that this category needs an NP (which plays the role of the subject in this case) as a left argument and the result of this category when an NP comes to its left is a sentence S. By contrast, in the sentence *I read a book*, the lexical category assigned to the verb *read* in this case is (S\NP)/NP, which means that it needs an NP as a left argument (which plays the role of the subject) and another NP as a right argument (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.

2.7.2 CCG Combinatory Rules

CCG combinatory rules define how to combine CCG categories. As most of the CCG grammar is contained in the lexicon, CCG combinatory rules consist of three simple rules: application rules, composition rules and type-raising rules. Application rules are originally part of the Categorial Grammar, which is a context free grammar. Composition and type-raising rules provide CCG with enough expressive power to be a mildly context-sensitive grammar (Joshi, 1985) and enable it to deal with many

grammatical phenomena such as non-constituent coordination (cf. Figure 2.11), crossed dependencies, relative clauses and long-range dependencies.

Application Rules

There are two types of application rules:

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

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

  In Figure 2.11, the forward application rule is applied between the category *S/NP*, which corresponds to the phrase *Marcel proved and I disproved* and the category *NP*, which corresponds to the word *completeness*, and the result is the sentence category *S*.

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

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

  In Figure 2.11, the backward application rule is applied between the category *(S/NP)\(S/NP)*, which corresponds to the phrase *and I disproved* and the category *S/NP*, which corresponds to the phrase *Marcel proved*, and the result is the category *S/NP*.

Composition Rules

Composition rules are divided into:

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

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

In Figure 2.11, the forward composition rule is applied between the category *S/(S\NP)*, which is the category that results from type-raising the category *NP* of the word *Marcel*, and the category *(S\NP)/NP*, which corresponds to the word *proved*, and the result is the category *S/NP*.

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

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

Type-raising Rules

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

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

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

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

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

In Figure 2.11, the forward type-raising rule is applied on the category *NP* of each of the words *Marcel* and *I*, resulting in the category *S/(S\NP)*. Type-raising in this case allowed the application of the forward composition rule between the subject *Marcel* and the verb *proved* to obtain a single category *S/NP* corresponding to the phrase *Marcel proved*. The category *S/NP* of the phrase *I disproved* is obtained in the same way.

2.7.3 CCG Supertagging

According to lexicalized grammars such as Lexicalized Tree Adjoining Grammar (LTAG) (Joshi and Schabes, 1991), Head-driven Phrase Structure Grammar (Pollard and Sag, 1994) and CCG, each word has a number of different lexical descriptors (supertags), which correspond to the different syntactic contexts in which the word may appear. The richness of syntactic information reflected by these descriptors massively increases their number. Thus, exploring every possible supertag for each word in the sentence during parsing will create a huge search space for the parser and reduce its speed significantly. A solution to this problem is supertagging (Bangalore and Joshi, 1999), which is the process of disambiguating the set of supertags assigned to the words of the sentence according to the word context before parsing. Bangalore and Joshi (1999) use statistics about supertag co-occurrences collected from a parsed corpus to reduce the number of supertags assigned to the words of the sentence. This removes the burden of supertag disambiguation from the parser and reduces the derivation search space, which results in faster parsing. Supertagging was originally developed for LTAG (Bangalore and Joshi, 1999). Later, a number of CCG supertagging approaches were devised (Clark, 2002a; Curran and Clark, 2003; Clark and Curran, 2004, 2007). After supertagging, the CCG parser only has to use combinatory rules to combine supertags assigned to the words to parse the sentence. That is why supertagging a sentence is considered to be "almost parsing" (Bangalore and Joshi, 1999). This helps to build an efficient CCG parser in spite of CCG spurious ambiguity, which was one of the main hurdles in the way of building an efficient CCG parser. Clark and Curran (2004) integrate the Maximum Entropy CCG supertagger of Curran and Clark (2003) into a wide-coverage CCG parser, which helped to obtain accurate and robust parsing that is significantly faster than comparable systems using other common grammar formalisms (Clark and Curran, 2004). Recently, Hassan et al. (2009) built an incremental linear-time CCG parser which is about ten times faster than the parser of Clark and Curran (2007).

2.7.4 Incorporating CCG in SMT Systems

CCG has many unique qualities which makes it an attractive grammar formalism to be incorporated in SMT systems. First, CCG allows for flexible structures thanks to its combinatory rules. Thus, it is possible to assign a CCG category to phrases which do not represent standard syntactic constituents. This is an important feature for SMT systems as SMT phrases are statistically extracted, and do not necessarily correspond to syntactic constituents in other theories of grammar. For example, in Figure 2.11, the phrase *Marcel proved and I disproved* has a CCG category *S/NP* although it is not considered a standard syntactic constituent in CF-PSG-based grammar theories. Second, CCG supertags present rich syntactic information at the lexical level about the dependents and local context of each word in the sentence. Therefore, CCG supertags reflect important information about the syntactic structure of the sentence without the need to build a full parse tree. This allows SMT systems to build grammaticality metrics based on examining sequences of CCG supertags of the words of the translation output. Third, being a mildly context-sensitive grammar, CCG can deal with many grammatical phenomena such as long-range dependencies, coordination and relative clauses, which current SMT systems have difficulty representing. Fourth, CCG categories are rich syntactic descriptors which reflect rich syntactic information about the syntactic function of the word/phrase. Thus, annotating phrases in SMT systems with CCG categories provides rich information about their grammatical function, which helps to use the phrases in their appropriate grammatical context during translation. Finally, CCG can be efficiently parsed thanks to the process of supertagging performed prior to parsing (cf. Section 2.7.3). This is especially important for computationally complex SMT systems. For these reasons, in addition to previous research in SMT, our research has tried to incorporate CCG into SMT in order to improve translation quality.

CCG-augmented SMT Systems

One of the earliest works which incorporates CCG into an SMT system was that of Hassan et al. (2007). They integrate CCG supertags into the target LM and the target side of the translation model of the PB SMT model. They also integrate a grammaticality metric over supertag sequences into the n-gram LM. This metric penalizes the number of violations of combinatory rules in a sequence of supertags. (Birch et al., 2007) use CCG supertags as a factor in the factored PB SMT translation model (Koehn and Hoang, 2007) following two approaches: the first approach generates CCG supertags as a target-side factor in the factored translation model, and then applies an n-gram language model over them. The second approach uses supertags as a source-side factor to direct the decoding process. Their experiments showed that sequence models over target-side CCG supertags performed better than the model which does not use supertags, as well as the model which uses sequence models over POS tags. However, they show that the improvement gained from using target-side CCG supertags is largely due to better local reordering. They suggest that using better reordering models or larger language models will lead to similar and more reliable improvements.

Hassan et al. (2009) integrate target-side CCG parsing in the Direct Translation Model (Ittycheriah and Roukos, 2007). They use a linear time incremental CCG parser, which is based on an incremental interpretation of the mechanisms of CCG. This parser builds CCG parsing states which are associated with translation states during decoding. Each parsing state posits a syntactic constraint on the next parsing state in such a way that the subsequent parsing states define a valid dependency structure. This helps to prune hypotheses which do not constitute a valid parsing state. In addition, they integrate a set of syntactic features based on CCG supertags, combinatory rules and parsing states in the direct translation model.

Haque et al. (2009) provide the PB SMT system with source-side contextual information by incorporating CCG supertags as source-language context features for phrases in the PB SMT model. They demonstrate that CCG-based contextual

features are more powerful than contextual features based on words and part of speech tags. In later work, Haque et al. (2010) incorporate the source-side CCG contextual features into the HPB SMT model. Similar to the results obtained for the PB SMT system, incorporating CCG contextual features in the HPB SMT system was demonstrated to outperform contextual features based on words and part of speech tags.

Recently, Mehay and Brew (2012) built a CCG-based syntactic reordering model to be incorporated into the PB SMT model. They use CCG categories extracted from the CCG parse chart to label target-side phrases in the translation model. Then, a syntactic reordering model is trained on the annotated data to learn phrase orientation (monotone, swap, discontinuous) based on the CCG label of the target phrase. They demonstrate that their CCG-based syntactic reordering model helps to achieve significant improvements over the bidirectional MSD lexicalized reordering model (Tillmann, 2004) for Urdu-to-English translation.

2.8 Summary

In this chapter, we provided a detailed description of the PB and HPB SMT frameworks, which are the most dominant approaches in SMT nowadays, and on which many approaches including the one presented in this thesis are based. In addition, we provided an overview of the syntax augmentation approaches for SMT systems with a special focus on SAMT. Finally, we introduced CCG, on which we base our syntax augmentation approach for HPB SMT, in addition to the previous approaches which incorporated CCG into SMT. In the next chapter, we present our CCG-based syntax augmentation approach for HPB SMT model and compare it with the context free phrase structure grammar-based SAMT approach.

Chapter 3

CCG Categories as Nonterminal Labels in Hierarchical Rules

In this chapter, we present an approach to incorporating CCG-based syntactic information in the HPB SMT model. Our goal is to improve the grammaticality of the translation output, that is why we use CCG categories to label target-side nonterminals and phrases in hierarchical rules. Augmenting the HPB SMT model with syntactic information by labelling nonterminals and phrases in hierarchical rules with syntax-based labels was first explored in SAMT (Zollmann and Venugopal, 2006). Our approach follows SAMT, but with a difference in the grammar formalism used; while SAMT uses context free phrase structure grammar (CF-PSG), our approach uses CCG. In this chapter, we attempt to answer our first research question (**RQ1**), by addressing whether CCG is better than CF-PSG in labelling nonterminals in the HPB SMT model.

In Section 3.1, we explain the motivation behind the idea of using CCG instead of CF-PSG to label nonterminals and phrases in the HPB SMT model. Section 3.2 reviews related work in the area of target-side sytnax-augmentation for HPB SMT. Section 3.3 provides an explanation of our algorithm for using CCG categories to label nonterminals in hierarchical rules. In Section 3.4, we present a comparison between the CCG-based nonterminal labels extracted by our approach and the CF-

PSG-based labels extracted by SAMT. In Section 3.5, we present the experiments we conducted to compare the performance of our approach with the state-of-the-art HPB and PB baseline systems in addition to the SAMT system. Furthermore, we examine the coverage of the syntactic labels, their sparsity and the size of the translation model in both the CCG-augmented HPB system and the SAMT system. In addition, we manually analyse the output of our CCG-augmented HPB system in comparison with the SAMT and HPB systems. Section 3.6 provides the conclusions for this chapter.

3.1 Motivation

One of the major challenges facing the incorporation of CF-PSG into SMT models results from the difference in the definition of the phrase between SMT and CF-PSG. A phrase in CF-PSG is a group of words which behave grammatically as a single unit. By contrast, 'phrases' in SMT are continuous groupings of words heuristically extracted from word alignments (cf. Section 2.3.1) with no requirement on linguistic grammaticality. Accordingly, the syntactic labels used by CF-PSG to label constituents cannot express the syntactic function of many SMT phrases. SAMT (Zollmann and Venugopal, 2006) (cf. Section 2.6) tries to tackle this problem by extending the standard CF-PSG-based constituent labels using a number of CCG-like slash operators and a 'plus' operator, which combine the standard constituent labels into more complex syntactic labels. This helps to express the syntactic function of a bigger set of phrases (i.e. better label coverage). Nonetheless, using such a method to invent new syntactic labels is largely influenced by the parse tree structure from which the labels are extracted, which leads to the extraction of different syntactic labels for the same phrase even when it plays the same syntactic role but within different tree structures. This creates redundant labels and limits the ability of the labels to accurately express the correct syntactic function of the extracted phrases. Furthermore, label redundancy leads to the enlargement of the

number of different syntactic labels used to label nonterminals and phrases in the HPB SMT model, which increases label sparsity and leads to the extraction of larger and sparser translation models in addition to tightening the syntactic restrictions imposed by the model, making them hard to satisfy at decoding time.

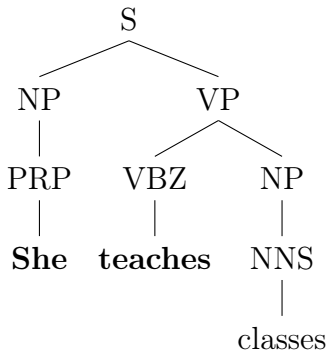

Figure 3.1: The CF-PSG parse tree of the English sentence *She teaches classes*.

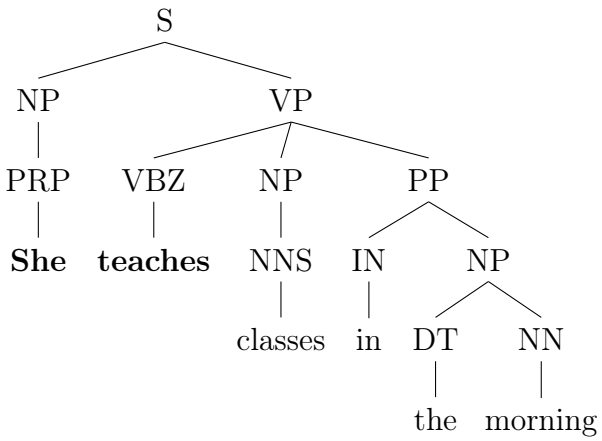

Figure 3.2: The CF-PSG parse tree of the English sentence *She teaches classes in the morning*.

Figure 3.1 shows the CF-PSG parse tree of the English sentence *She teaches classes*. The SAMT label assigned to the phrase *She teaches* in this sentence is S/NP, which means that it is a sentence lacking a noun phrase playing the role of the object to its right. Figure 3.2 shows the same phrase *She teaches* participating in a different syntactic tree structure which is of the sentence *She teaches classes in the morning*. In this case, SAMT assigns the phrase *She teaches* PRP+VBZ as a syntactic label, although it has the same syntactic function, namely a transitive verb with its subject, in both sentences. However, the attachment of the prepositional

phrase *in the morning* as an adjunct to the verb *teaches* in the second case prevents SAMT from assigning S/NP as a syntactic label to the phrase *She teaches*. This example highlights the sensitivity of the SAMT labelling approach to the specific structure of the parse tree from which the phrases are extracted, which what causes the phrase *She teaches* to have redundant labels. The same example highlights issues with SAMT label accuracy. The syntactic label PRP+VBZ attached to the phrase *She teaches* in Figure 3.2 does not accurately reflect the syntactic function of this phrase, because the label does not indicate that the verb *teaches* is a transitive verb which needs a noun phrase playing the role of the object to its right. A SAMT rule, whose target side is $< \ PRP + VBZ \ classes \ in \ the \ morning \ >$, might be extracted from this tree, which allows an intransitive verb and its subject to replace the nonterminal, leading to the production of ungrammatical translation.

In our approach, we try to extract nonterminal labels which are rich and provide better coverage for SMT phrases than SAMT labels while being less sparse at the same time. To achieve this we use another grammar formalism, namely Combinatory Categorial Grammar (CCG) (Steedman, 2000), to label phrases and nonterminals in hierarchical rules. We believe that CCG simple combinatory rules in addition to its lexicalized nature allow more flexible structures compared to CF-PSG, which gives CCG category labels the potential to provide better coverage for SMT phrases than SAMT labels. Furthermore, CCG categories are inherently rich, accurately reflecting information about the syntactic context and dependents of a word/phrase. Moreover, CCG categories are the constituent labels in the CCG formalism, which helps CCG category labels to avoid label redundancy problem from which SAMT labels suffer. Thus, CCG enables us to extract more accurate and richer nonterminal labels, while at the same time being less sparse compared to SAMT labels. Figures 3.3 and 3.4 show the best sequence of CCG supertags assigned to the words of the same sentences in Figures 3.1 and 3.2, respectively. If we combine the supertags of the words of the phrase *She teaches* in Figure 3.3, we obtain S[dcl]/NP as the category representing the syntactic function of the phrase, namely a sentence

lacking a NP to its right. The same CCG category results from combining the supertags of the words of the same phrase in Figure 3.4. Thus, if we assign the CCG category that results from combining the supertags of the words of the phrase as its syntactic label, we guarantee the extraction of rich syntactic labels which are not redundant.

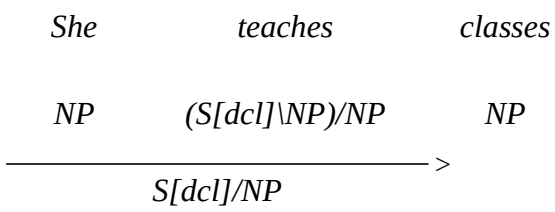

Figure 3.3: The best sequence of CCG supertags assigned to the English sentence *She teaches classes.*

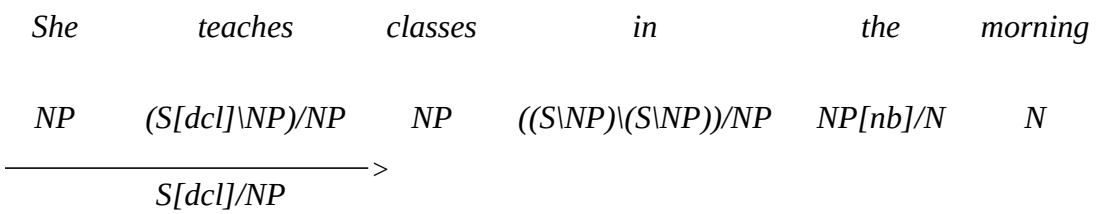

Figure 3.4: The best sequence of CCG supertags assigned to the English sentence *She teaches classes in the morning.*

3.2 Related Work

Several researches have tried to augment the HPB SMT model with target-side syntax by imposing syntax-based restrictions on phrases replacing nonterminals during decoding. One of the most prominent works in this field is the SAMT approach of Zollmann and Venugopal (2006) (cf. Section 2.6). SAMT attaches CF-PSG-based

labels to target-side nonterminals and phrases in the HPB SMT translation model. These labels ensure that the phrase with the correct syntactic function is used to replace each nonterminal during decoding. Vilar et al. (2008) use CF-PSG to extract syntactic features which score source and target phrases according to the degree to which they match a syntactic constituent. If a phrase is not spanned by a single constituent, a score is calculated based on the minimum number of words that need to be added or deleted from the phrase so that it matches a syntactic constituent. Chiang (2010) extends the SAMT-based nonterminal labelling approach so that an indefinite number of constituents can be incorporated into the label. This approach is used to extract tree-to-tree translation rules which provide full coverage of the training data.

Vilar et al. (2010) derive labels for nonterminals in hierarchical rules using a clustering method that does not use any syntactic information. Phrase pairs are clustered according to classes assigned to source and target words. This method has the advantage of using a small set of labels without the need for any syntactic preprocessing for the training data. Zollmann and Vogel (2011) use lexical tags to label phrases and nonterminals in hierarchical rules. These lexical tags can be POS tags or word-based classes extracted using an automatic clustering method. The first word and the last word tags in the phrase are used to extract its left-hand-side label. They also use a clustering-based approach to label phrases based on feature vectors which contain information about the tags of the words within the phrase in addition to the tags of the contextual words surrounding the phrase. Then, clusters are extracted from the feature vectors using the K-means (MacQueen, 1967) algorithm. Finally, each phrase is labelled with the appropriate cluster to which its feature vector belongs. Weese et al. (2012) used CCG categories to label nonterminals in the HPB SMT model. They explored two label extraction approaches. The first approach extracts CCG categories from the 1-best parse trees of target-side sentences in the parallel corpus. The second approach extracts CCG labels from the CCG parsing charts of target-side sentences. They compared their CCG-based

models with the SAMT model in terms of rule table size and label sparseness. They found that their CCG-based nonterminal labelling approaches extracted less sparse labels than the SAMT approach. Furthermore, they found that their CCG-based systems were faster and have smaller rule tables than the SAMT system. Li et al. (2012) augment the HPB SMT model with syntactic information extracted from source-side dependency structures. They annotate nonterminals in hierarchical rules with syntactic labels extracted from the POS tags of the head words within source-side phrases. They demonstrate that their head-driven system outperforms a SAMT system which uses syntax on the source side.

Recently, approaches which try to use semantic information in nonterminal labelling have emerged. Baker et al. (2010) try to incorporate semantic information into syntax-based nonterminal labels extracted using CF-PSG. The syntactic labels extracted from target-side parse trees are augmented with named entities and modalities. Gao and Vogel (2011) incorporate target-side predicate-argument structures into HPB SMT. They use Semantic Role Labelling (SRL) to annotate nonterminals in the HPB SMT model to form an SRL-aware SCFG which works side by side with the basic HPB SMT unlabelled SCFG.

3.3 Using CCG Categories to Label Nonterminals in Hierarchical Rules

In light of previous research which demonstrated the advantages of using CCG in SMT systems (Birch et al., 2007; Hassan et al., 2007, 2009) (cf. Section 2.7.4), and following the SAMT approach (Zollmann and Venugopal, 2006) in imposing target-side syntactic constraints in HPB SMT by syntactically labelling nonterminals in hierarchical rules, we try to employ target-side CCG-based syntactic constraints in HPB SMT. This is accomplished by using CCG categories extracted from CCG forest trees of target-side sentences in the training corpus to label phrases and nonterminals in the HPB SMT model. The CCG-based labelling algorithm proceeds according to

the following steps:

- Each target-side sentence from the parallel corpus is supertagged by assigning the best sequence of CCG supertags to its words.

- CCG forest trees are built for each target-side sentence based on its best sequence of supertags. The forest trees are represented in a parsing chart which contains all possible CCG categories for each subspan in the sentence. Figure 3.5 shows only the best CCG parse tree of the English sentence *Shura council rejects issuance of new press law* and its aligned Arabic sentence.

- Phrase pairs are extracted from the parallel corpus according to the PB SMT phrase extraction approach (cf. Section 2.3).

- Each target-side phrase is assigned a syntactic label which corresponds to the highest-scoring CCG category covering the phrase in the chart. In other words, each target-side phrase is assigned a CCG category which results from the highest-scoring combination of the supertags of its words. In case there is no CCG category covering the phrase, a general X label is assigned to it. This X label indicates that there is no syntactic constraint imposed on the phrase. X-labelled nonterminals do not impose any syntactic constraint on the phrases replacing them. Figure 3.6 shows a set of initial rules, which have CCG categories assigned to their left-hand sides, extracted from the sentence pair in Figure 3.5.

- Hierarchical rules are extracted from the CCG-labelled sentence pairs according to the same HPB SMT rule extraction method (Chiang, 2005) (cf. Section 2.4.1). This results in hierarchical rules which have CCG categories assigned to their nonterminals and left-hand sides. Figure 3.7 shows a set of hierarchical rules augmented with CCG categories extracted from the sentence pair in Figure 3.5.

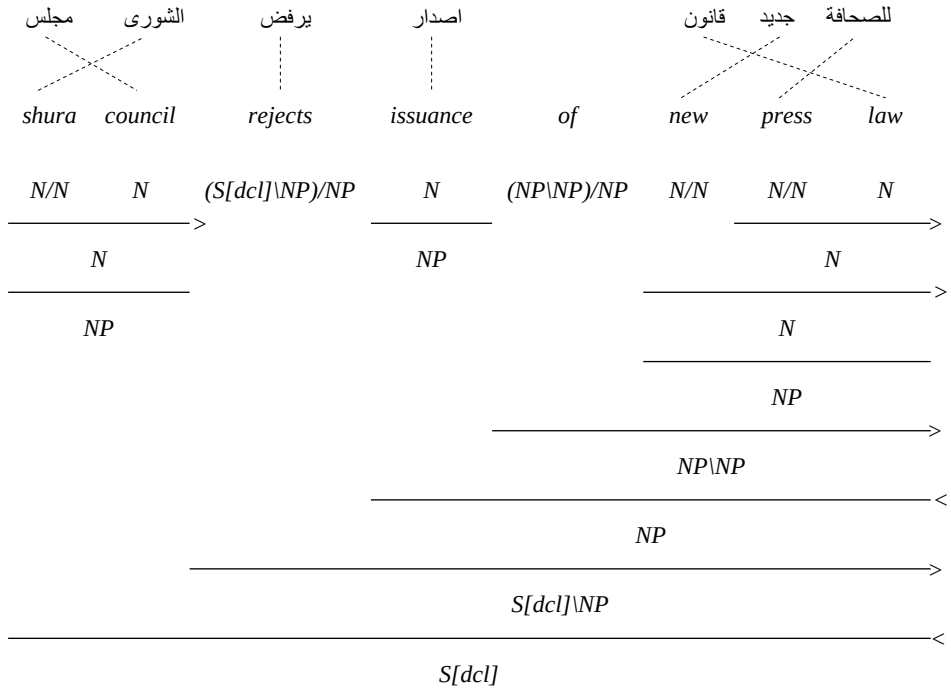

Figure 3.5: A CCG parse tree of the the English sentence *Shura council rejects issuance of new press law* and its aligned Arabic sentence.

LHS	***α***	***β***
N/N	*الشورى*	*shura*
N	*مجلس*	*council*
N	*قانون*	*law*
N/N	*جديد*	*new*
NP/NP	*إصدار*	*issuance of*
NP	*مجلس الشورى*	*shura council*
N/N	*جديد للصحافة*	*new press*
NP	*قانون جديد للصحافة*	*new press law*
S[dcl]/NP	*مجلس الشورى يرفض*	*shura council rejects*
(S[dcl]\NP)/NP	*يرفض إصدار*	*rejects issuance of*
S[dcl]/NP	*مجلس الشورى يرفض إصدار*	*shura council rejects issuance of*
NP	*إصدار قانون جديد للصحافة*	*issuance of new press law*

Figure 3.6: A set of initial rules augmented with CCG categories extracted from the Arabic–English sentence pair in Figure 3.5.

LHS	*α*	*β*
NP	*X* قانون جديد	*new N/N law*
NP	قانون *X* للصحافة	*N/N press law*
NP	مجلس *X*	*N/N council*
S[dcl]	مجلس الشورى يرفض *X*	*shura council rejects NP*
S[dcl]/NP	مجلس الشورى يرفض *X*	*shura council rejects NP/NP*
S[dcl]	X_1 يرفض X_2	NP_1 *rejects* NP_2
S[dcl]	مجلس الشورى X_1 إصدار X_2 جديد للصحافة	*shura council* $(S[dcl]\backslash NP)/NP_1$ *issuance of new press* N_2
S[dcl]	مجلس الشورى X_1 إصدارقانون X_2 للصحافة	*shura council* $(S[dcl]\backslash NP)/NP_1$ *issuance of* N/N_2 *press law*
S[dcl]	X_1 يرفض إصدار X_2	NP_1 *rejects issuance of* NP_2

Figure 3.7: A set of hierarchical rules augmented with CCG categories extracted from the Arabic–English sentence pair in Figure 3.5.

We argue that CCG is more suitable to label nonterminals in hierarchical rules than CF-PSG, which corresponds to our first research question (**RQ1**). In Sections 3.4 and 3.5, we try to give logical and empirical arguments in favour of this hypothesis, respectively.

3.4 CCG-based Labels vs. CF-PSG-based Labels

Being a lexicalized grammar, CCG has many properties which favour its integration in SMT systems over CF-PSG. Firstly, CCG has flexible structures which can be easily combined using a small set of CCG combinatory rules. This enables the CCG parser to assign CCG categories to words as well as phrases even when they do not correspond to a standard syntactic constituent. This is a very important feature for SMT, as phrases used in SMT systems are extracted according to purely statistical methods; therefore, they do not necessarily correspond to grammatical constituents. To increase the coverage of CF-PSG constituent labels, SAMT used a set of CCG-like slash operators and a 'plus' operator, which extend the set of syn-

tactic labels used to label nonterminals, and enable the system to extract syntactic labels for part of the phrases which do not correspond to grammatical constituents (cf. Section 2.6.1). Nevertheless, SAMT labelling rules have limited flexibility in comparison with the flexibility of CCG structures and will fail to label phrases which do not comply with SAMT labelling rules. Secondly, a CCG category attached to a word/phrase presents information about the local syntactic context and dependents of the word/phrase, combining only the elements on which the word/phrase imposes syntactic constraints. This makes a CCG category a rich syntactic description which accurately represents the syntactic function of the word/phrase. In contrast, SAMT labels, which are formed by combining CF-PSG-based syntactic constituents using binary operators, might combine constituents that do not necessarily impose syntactic constraints on one another. As a result, SAMT labels are not syntactically as rich or as accurate as CCG categories. Thirdly, most of the CF-PSG parsers are trained on the Penn Treebank (Marcus et al., 1993), which has many flat structures. Flat structures are a major problem facing the extraction of syntax-augmented translation rules, because they lead to the extraction of translation rules with weak generalization ability (Wang et al., 2010). A solution to the problem of flat structures is to perform tree binarization. Wang et al. (2010) use Expectation Maximization (Dempster et al., 1977) to learn binarization preference (left or right) for each tree node as a priori to extract tree transducer-based translation rules (Knight and Graehl, 2005). By contrast, CCG trees are all binary trees, because all CCG combinatory rules are binary. This eliminates the problem of flat structures and enables translation rules with good generalization ability to be extracted. Lastly, CCG supertagging, which disambiguates the set of supertags assigned to the words of the sentence before parsing, drastically reduces the parsing search space and makes CCG efficient to parse compared to CF-PSG (Clark and Curran, 2004) (cf. Section 2.7.3).

Figures 3.8 and 3.9 illustrate the CF-PSG-based parse tree and the CCG parse tree of the sentence: *Australia is one of the countries which have diplomatic relations*

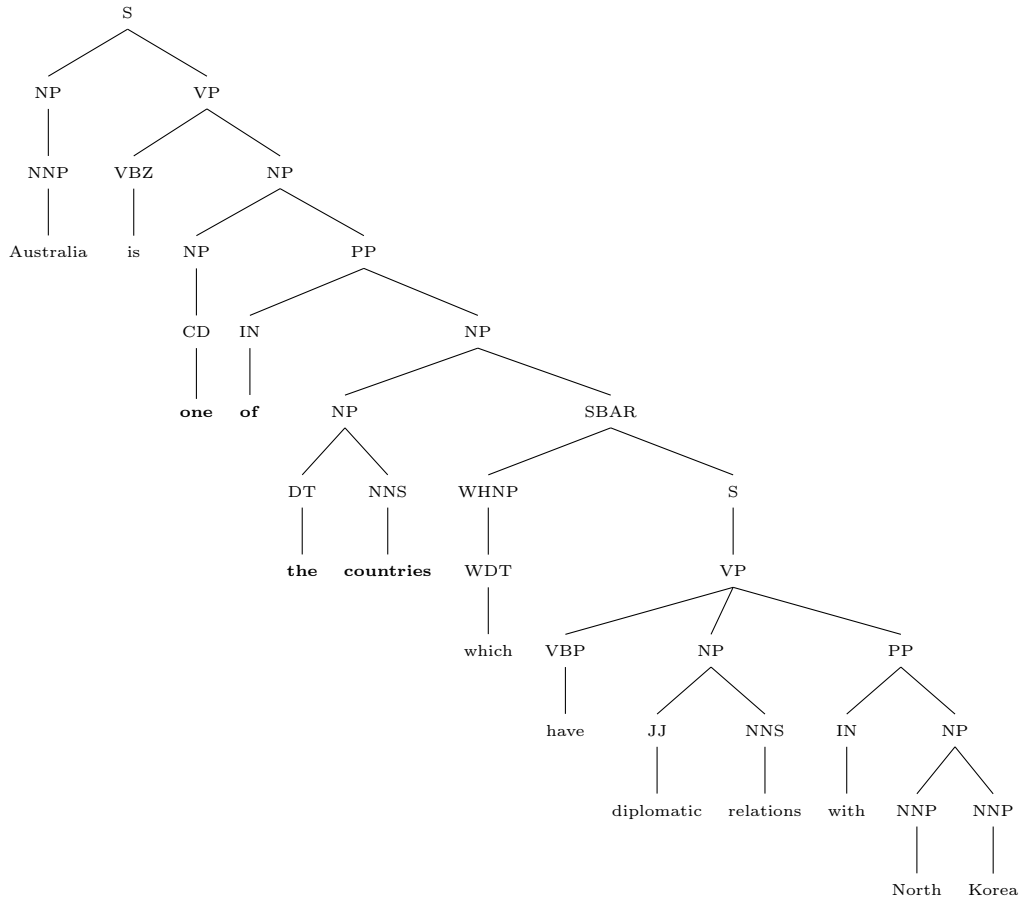

Figure 3.8: CF-PSG parse tree of the sentence *Australia is one of the countries which have diplomatic relations with North Korea.*

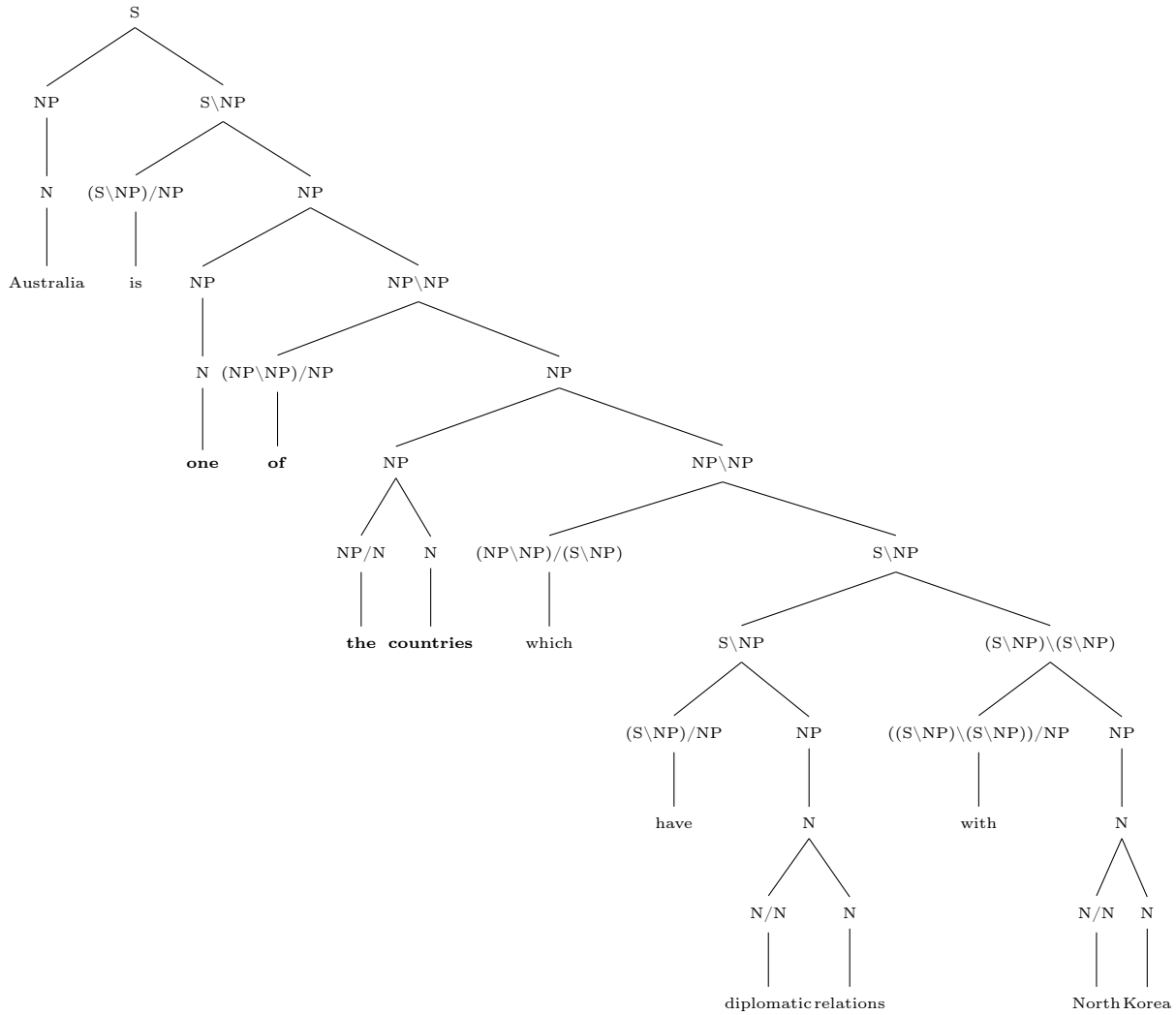

Figure 3.9: CCG parse tree of the sentence *Australia is one of the countries which have diplomatic relations with North Korea*

with North Korea, respectively. Suppose that this sentence is a target translation of a sentence in some foreign language, and a phrase pair, whose target side is *one of the countries*, is extracted from this sentence. In this case, SAMT assigns NP/SBAR to this phrase as its syntactic category. However, this phrase apparently does not necessarily need an SBAR as a part of its syntactic context and thus this label does not accurately reflect the syntactic function of this phrase. By contrast, a CCG parser assigns the category NP to this phrase, which results from combining the supertags NP, NP\NP, NP/N and N of the words *one, of, the* and *countries*, respectively as illustrated in Figure 3.9. The category NP accurately reflects the syntactic function of the phrase *one of the countries*. It is a complete noun phrase on its own and does not need an SBAR constituent to its right as part of its syntactic context as its assigned SAMT label indicates. We can see from the CCG parse tree in Figure 3.9 that CCG uses the supertag (NP\NP)/(S\NP) of the relative pronoun *which* to link the relative clause *have diplomatic relations with North Korea* with the noun phrase *one of the countries*, rather than putting the responsibility of linking the relative clause on the noun phrase. Moreover, SAMT fails to find a syntactic category for the phrase: *is one of the*, because this phrase crosses phrase boundaries and thus does not comply with any of the SAMT labelling rules. By contrast, our CCG-based approach succeeds in extracting a syntactic category (S\NP)/N for this phrase by simply combining the supertags (S\NP)/NP, NP, (NP\NP)/NP and NP/N of the words *is, one, of* and *the*, respectively. This category indicates that this phrase needs a noun phrase playing the role of the subject to its left and a noun to its right in order to form a complete sentence.

3.5 Experiments

This section presents the results of our experiments which compare our CCG-augmented HPB SMT system with both a SAMT system and a HPB SMT baseline system. We conducted experiments in the news and travelling speech expressions

domains for Arabic-to-English and Chinese-to-English translation. The travelling speech expressions experiments were conducted as part of our participation in the IWSLT 2010 (International Workshop on Spoken Language Translation) evaluation campaign[1] in each of the BTEC (basic travelling expressions) Arabic-to-English translation task and the DIALOG (spoken dialogues in travel situations) Chinese-to-English translation task. Our system was the third and fourth best-performing system on the BTEC Arabic-to-English 2009 and 2010 test sets, respectively.

3.5.1 Data and Settings

In our speech expressions experiments, we used the data provided by the IWSLT 2010 evaluation campaign.[2] The Chinese–English training corpus consists of 63234 sentence pairs from the IWSLT 2010 Chinese–English training data for the DIALOG task. The development and test sets are IWSLT evaluation data sets provided for the Chinese-to-English DIALOG task for 2008 and 2009 evaluations, respectively with 500 sentence pairs each. The development set has 15 references and the test set has 7 references. The Arabic–English training corpus consists of 21484 sentence pairs from the IWSLT 2010 training data provided for the Arabic-to-English BTEC task. The development and test sets are the IWSLT evaluation data sets provided for the BTEC task for 2007 and 2008 evaluations, respectively with 489 sentence pairs in the development set and 507 sentence pairs in the test set. The development set has 7 references and the test set has 16 references. For the Arabic-to-English news experiments, we used a corpus comprised of 48065 sentence pairs selected randomly from the Arabic News corpus from LDC.[3] For the Chinese-to-English news experiments, we used data comprised of 51044 sentence pairs selected randomly from the FBIS corpus. The development and test sets for the Arabic-to-English and Chinese-to-English news experiments have 500 sentence pairs each with a single reference for each sentence. Table 3.1 summarizes the data sets used in our experiments.

[1] http://iwslt2010.fbk.eu/node/1
[2] http://iwslt2010.fbk.eu/node/27
[3] http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2004T17

Corpus	Training set	Development set	Test set
AE_{news}	48065	500	500
CE_{news}	51044	500	500
AE_{IWSLT}	21484	489	507
CE_{IWSLT}	63234	500	500

Table 3.1: Data used in our experiments.

All the English data used in our experiments is lower cased and tokenized. The Arabic data is segmented according to the D3 segmentation scheme (Sadat and Habash, 2006) using MADA (Morphological Analysis and Disambiguation for Arabic) (Habash et al., 2009). The Chinese data is segmented using ICTCLAS Chinese lexical analyser (Zhang et al., 2003). The GIZA++ toolkit[4] is used to perform word and phrase alignment and the "grow-diag-final-and" refinement method is adopted (Koehn et al., 2003). Minimum error rate training (Och, 2003) is performed to tune all our SMT systems. The 5-gram language models in all the experiments are built from the target side of the training corpus using the SRILM toolkit[5] (Stolcke, 2002) with modified Kneser-Ney smoothing (Kneser and Ney, 1995).

We used paired bootstrap resampling (Koehn, 2004b) to compute the statistical significance at p-level=0.05 for all the improvements in our experiments in this thesis.

3.5.2 Systems

The HPB and PB Baselines

We build our HPB baseline using the Moses Chart Decoder[6] (Hoang et al., 2009b) with maximum phrase length and maximum rule span set to 12 words. Extracted hierarchical rules contain up to 2 nonterminals. Maximum chart span is set to 20 words. Cube pruning pop-up limit is set to 1000. The PB baseline system is built using the Moses Phrase-Based Decoder (Koehn et al., 2007) with maximum phrase

[4]http://fjoch.com/GIZA++.html
[5]http://www-speech.sri.com/projects/srilm/
[6]http://www.statmt.org/moses/?n=Moses.SyntaxTutorial

length set to 12 words.

The SAMT System

For the SAMT system, we parse the English side of the training corpus using the Berkeley parser which is trained on the Wall Street Journal section of the Penn Treebank with 6 merge-split-smooth cycles (Petrov and Klein, 2007).[7] We build our SAMT system using the Moses Chart Decoder with SAMT4 scheme.[8] For the SAMT system, we also set maximum phrase length and maximum rule span to 12 words. Rules extracted contain up to 2 nonterminals. Cube pruning pop-up limit and maximum chart span are set to the same values as the HPB baseline system.

The 1-best CF-PSG System

This system uses labels extracted from the 1-best CF-PSG-based parse trees of the target side of the training data. The Berkeley parser is used to parse the target side of the training data. Then, we label each target-side phrase with the label of the constituent which covers the phrase in the parse tree. If the phrase is not spanned by a constituent in the parse tree, it is discarded. The Moses Chart Decoder is then used to extract the HPB grammar and perform decoding with the same settings as the HPB baseline system.

The CCG-Augmented HPB System

We use the CCG parser from C&C tools[9] (Clark and Curran, 2007) to build the parsing chart for each sentence in the English side of the training data for our CCG-augmented HPB system experiments. We then annotate target-side phrases with CCG categories extracted from the parsing chart as described in Section 3.3. Finally, we build our CCG-augmented HPB system using the Moses Chart Decoder. We use the same phrase and rule extraction settings as the HPB baseline system. Cube

[7] http://code.google.com/p/berkeleyparser/
[8] http://www.statmt.org/moses/?n=Moses.SyntaxTutorial
[9] http://svn.ask.it.usyd.edu.au/trac/candc/

pruning pop-up limit and maximum chart span are also set to the same values as the HPB baseline system.

The 1-best CCG System

This system uses labels extracted from the 1-best CCG parse trees of the target side of the training data. The C&C parser is used to parse the target side of the training data. Each target-side phrase is labelled with the CCG category which covers the phrase in the 1-best parse tree. If the phrase does not correspond to a CCG category in the parse tree, it is discarded. The Moses Chart Decoder is then used to extract the HPB grammar and perform decoding with the same settings as the HPB baseline system.

3.5.3 Arabic-to-English Experimental Results

Tables 3.2 and 3.3 show the BLEU, TER and METEOR scores of the HPB and PB baselines, the CCG-augmented HPB and SAMT systems, and the 1-best CCG and CF-PSG systems for Arabic-to-English news and speech expressions translation, respectively.

Table 3.2 shows that the PB baseline system is the best-performing system in terms of BLEU, METEOR and TER scores, outperforming the next best-performing system, namely our CCG-augmented HPB system by a slight improvement of 0.16 absolute BLEU points, which corresponds to a 0.96% relative BLEU improvement. The result of the paired bootstrap test shows that the PB baseline system is better than our CCG-augmented HPB system in 72% of the samples, which is not statistically significant at p-level=0.05. Our CCG-augmented HPB system outperforms the HPB baseline on TER and BLEU scores, beating it by 0.37 absolute BLEU points, which corresponds to a 1.63% relative improvement. The result of the paired bootstrap resampling test demonstrates that our CCG-augmented HPB system is better than the HPB baseline system in 88% of the samples at p-level=0.05, which is not statistically significant. In addition, our CCG-augmented HPB system outperforms

the SAMT system on BLEU, TER and METEOR scores. Our CCG-augmented system is better than the SAMT system by 0.13 absolute BLEU points, which corresponds to a 0.57% relative improvement. However, the paired bootstrap resampling test shows that our CCG-augmented HPB system is better than the SAMT system in 71% of the samples, which is not statistically significant at p-level=0.05. Table 3.2 also shows that the 1-best CCG and CF-PSG systems underperform in comparison with our CCG-augmented HPB system and the SAMT system by 0.18 and 0.2 absolute BLEU points, which corresponds to relative decreases of 0.78% and 0.88%, respectively. The table also demonstrates that the 1-best CCG system outperforms the 1-best CF-PSG systems by 0.16 absolute BLEU points, which corresponds to a 0.7% relative improvement. However, this improvement is statistically insignificant at p-level=0.05.

System	**BLEU**	**TER**	**METEOR**
PB	**23.13**	**64.00**	**57.11**
CCG	22.97	64.12	56.15
SAMT	22.83	64.63	55.96
CCG 1-best	22.79	64.12	56.79
CF-PSG 1-best	22.63	64.65	56.59
HPB	22.60	64.32	56.21

Table 3.2: BLEU, TER and METEOR scores of the HPB and PB baselines, the CCG-augmented HPB and SAMT systems, and the 1-best CCG and CF-PSG systems for Arabic-to-English news translation.

For Arabic-to-English speech expressions translation, Table 3.3 shows that the HPB baseline system is the best-performing system in terms of BLEU, TER and METEOR scores, outperforming the second best-performing system, namely the SAMT system by 0.85 absolute BLEU points, which corresponds to a 1.87% relative improvement. The paired bootstrap resampling test demonstrates that this improvement is statistically significant at p-level=0.05. Our CCG-augmented HPB system performs very closely to the SAMT system, with only 0.01 absolute BLEU points the difference between the two systems. The PB baseline performs also so closely to our CCG-augmented system with a small difference of 0.01 absolute BLEU

points between the two systems. The 1-best CCG and CF-PSG systems underperform in comparison with our CCG-augmented HPB system and the SAMT system by 3.04 and 4.12 absolute BLEU points, which corresponds to relative decreases of 7.9% and 5.8%, respectively. The 1-best CCG system outperforms the 1-best CF-PSG system by 1.07 absolute BLEU points, which corresponds to a 2.2% relative improvement. However, this improvement is not statistically significant at p-level=0.05.

System	**BLEU**	**TER**	**METEOR**
HPB	**53.20**	**30.95**	**71.43**
SAMT	52.33	31.13	70.45
CCG	52.32	31.89	70.86
PB	52.31	32.42	70.80
CCG 1-best	49.28	33.81	69.01
CF-PSG 1-best	48.21	35.14	67.59

Table 3.3: BLEU, TER and METEOR scores of the HPB and PB baselines, the CCG-augmented HPB and SAMT systems, and the 1-best CCG and CF-PSG systems for Arabic-to-English speech expressions translation.

3.5.4 Chinese-to-English Experimental Results

Tables 3.4 and 3.5 show the BLEU, TER and METEOR scores of the HPB and PB baselines, the CCG-augmented HPB and SAMT systems, and the 1-best CCG and CF-PSG systems for Chinese-to-English news and speech expressions translation, respectively.

Table 3.4 shows that our CCG-augmented HPB system outperforms the SAMT system in terms of BLEU, METEOR and TER scores. Our CCG-augmented HPB system outperforms the SAMT system by 0.34 absolute BLEU points, which corresponds to a 1.48% relative improvement. The result of the paired bootstrap resampling test shows that our CCG-augmented HPB system is better than the SAMT system in 84% of the samples, which is not statistically significant at p-level=0.05 . Our CCG-augmented HPB system does not outperform the HPB baseline system in any of the evaluation metrics. The HPB baseline outperforms our CCG-augmented

System	BLEU	TER	METEOR
PB	**23.74**	67.78	52.56
HPB	23.65	**66.81**	**52.76**
CCG	23.30	67.50	52.32
SAMT	22.96	68.01	51.43
CCG 1-best	22.87	67.31	52.25
CF-PSG 1-best	22.66	67.60	50.98

Table 3.4: BLEU, TER and METEOR scores of the HPB and PB baselines, the CCG-augmented HPB and SAMT systems, and the 1-best CCG and CF-PSG systems for Chinese-to-English news translation.

HPB system by 0.35 absolute BLEU points, which corresponds to a 1.5% relative improvement. The result of the paired bootstrap resampling test shows than the HPB baseline system is better than our CCG-augmented HPB system in 84% of the samples, which is not statistically significant at p-level=0.05. The table also shows that the 1-best CCG and CFG-PSG systems underperform in comparison with our CCG-augmented HPB system and the SAMT system by 0.43 and 0.3 absolute BLEU points, which corresponds to relative decreases of 1.9% and 1.3%, respectively. The 1-best CCG system outperforms the 1-best CF-PSG by 0.21 absolute BLEU points, which corresponds to a 0.93% relative improvement. However, this improvement is not statistically significant at p-level=0.05.

For Chinese-to-English speech expressions translations, Table 3.5 shows that the HPB baseline system is also the best-performing system in terms of BLEU, TER and METEOR scores, outperforming the second best-performing system, namely the CCG-augmented HPB system by 2.63 absolute BLEU points, which corresponds to a 5.54% relative improvement. Our CCG-augmented HPB system outperforms the SAMT system in terms of BLEU, METEOR and TER scores. Our CCG-augmented HPB system outperforms the SAMT system by 2.66 absolute BLEU points, which corresponds to a 5.83% relative improvement. The result of the paired bootstrap resampling test shows that our CCG-augmented HPB system is better than the SAMT system in 100% of the samples, which is statistically significant at p-level=0.05. In addition, our CCG-augmented HPB system outperforms the PB baseline system by

System	BLEU	TER	METEOR
HPB	**50.89**	**32.53**	**66.81**
CCG	48.26	36.17	64.88
PB	47.69	34.20	65.35
CCG 1-best	45.93	38.16	64.35
SAMT	45.60	36.20	64.16
CF-PSG 1-best	43.88	39.29	63.46

Table 3.5: BLEU, TER and METEOR scores of the HPB and PB baselines, the CCG-augmented HPB and SAMT systems, and the 1-best CCG and CF-PSG systems for Chinese-to-English speech expressions translation.

0.57 absolute BLEU points, which corresponds to a 1.19% relative improvement. The result of the paired bootstrap resampling test shows that our CCG-augmented HPB system outperforms the PB baseline system in 54% of the samples at p-level=0.05, which is not statistically significance. Our CCG-augmented HPB system does not outperform the PB baseline in METEOR and TER scores. The table also demonstrates that the 1-best CCG and CF-PSG systems underperform in comparison with our CCG-augmented HPB system and the SAMT system by 2.33 and 1.72 absolute BLEU points, which corresponds to relative decreases of 4.8% and 3.8%, respectively. The 1-best CCG system outperforms the 1-best CF-PSG system by 2.05 absolute BLEU points, which corresponds to a 4.7% relative improvement. This improvement is statistically significant at p-level=0.05. It is worth noting that the 1-best CCG system outperforms the SAMT system by 0.33 absolute BLEU points, which corresponds to a relative improvement of 0.7%.

3.5.5 Analysis

The experimental results presented in Sections 3.5.3 and 3.5.4 show that our CCG-augmented HPB system outperformed the SAMT system in terms of BLEU score in all the experiments except for the Arabic-to-English speech expressions translation experiment, in which both systems performed similarly. However, our CCG-augmented HPB system was not able to outperform the HPB baseline system in most of the experiments. In order to analyse the performance of the SAMT and

CCG-augmented HPB systems beyond automatic evaluation metrics, we compare the models of these systems in terms of rule table size, label sparsity, and label coverage. We also perform manual analysis comparing the translation quality of our CCG-augmented HPB system with each of the HPB baseline and the SAMT systems on selected sentence pairs.

Rule Table Size

Table 3.6 shows the number of rules/phrases in the translation model of the HPB and PB baselines, the CCG-augmented HPB and SAMT systems, the 1-best CCG and CF-PSG systems built on the Arabic–English and Chinese–English news and IWSLT data. We can see that the SAMT system has bigger rule tables than the CCG-augmented HPB system on all the data sets. The rule table of the SAMT system has about 1.67 and 1.44 times as many rules as the rule table of the CCG-augmented HPB system built on the Chinese–English IWSLT and news data, respectively. The size of the rule table of the SAMT system is about 1.81 and 1.48 times the size of the rule table of the CCG-augmented system built on the Arabic–English IWSLT and news data, respectively. The table shows that the translation models of our CCG-augmented HPB and the SAMT systems are significantly larger than the translation models of the HPB and PB baseline systems. The 1-best CCG and CF-PSG systems have significantly less rules than the CCG-augmented HPB and SAMT systems. The table also demonstrate that the 1-best CCG system has 1.54 and 1.44 times as many rules as the 1-best CF-PSG system on Chinese–English IWSLT and news data, respectively. For Arabic-to-English translation on news and IWSLT data, the 1-best CCG system has 1.5 and 1.42 times as many rules as the 1-best CF-PSG system, respectively. The CCG category labels are richer and thus more fine-grained than CF-PSG constituent labels, which is why the 1-best CCG systems have more rules than the 1-best CF-PSG systems. The 1-best CCG and CF-PSG systems have smaller translation models than the PB and HPB baseline systems, because phrases which do not correspond to syntactic constituents in the 1-best systems are

System	$\mathbf{CE}_{news}$	$\mathbf{AE}_{news}$	$\mathbf{CE}_{IWSLT}$	$\mathbf{AE}_{IWSLT}$
HPB	25,062,994	36,226,020	3,468,512	1,722,831
PB	4,433,124	6,281,738	981,070	494,843
SAMT	45,761,909	58,208,162	10,983,675	5,248,422
CCG	31,667,277	39,370,226	6,578,072	2,895,825
CCG 1-best	1,881,731	3,347,679	542,884	274,945
CF-PSG 1-best	1,219,192	2,217,336	376,897	193,655

Table 3.6: Translation model size in terms of number of rules/phrases in the HPB and PB baselines, the CCG-augmented HPB and SAMT systems, and the 1-best CCG and CF-PSG systems built on the Arabic–English and Chinese–English news and IWSLT data. AE stands for Arabic–English, and CE stands for Chinese–English.

discarded.

Label Sparsity

Given that we use the same rule extraction method with the same phrase length and rule span parameters for both the CCG-augmented HPB and SAMT systems, the number of rules extracted by each system depends on how many different syntactic labels assigned to phrase pairs extracted from the training corpus. The number of different syntactic labels used in syntax-augmented hierarchical SMT models is a key factor affecting their performance. Sparse syntactic labels increase the size of the model and produce low-probability rules, which causes the generalization ability of the system to deteriorate. Furthermore, the enlargement of the number of different syntactic labels used in the model poses hard to satisfy restrictions on the decoding search space, preventing the system from producing many potentially better translations. That is why we conduct a comparison between the CCG-based and CF-PSG-based HPB systems by checking the number of different syntactic labels used to annotate the target side of the training data. Table 3.7 shows the number of different syntactic labels used by the SAMT system, the CCG-augmented HPB system, the 1-best CCG and CF-PSG systems to annotate the target side of the Chinese–English and Arabic–English news and IWSLT data. Table 3.7 demonstrates that the SAMT system uses about 10 times as many different syntactic labels as

System	$\mathbf{CE}_{news}$	$\mathbf{AE}_{news}$	$\mathbf{CE}_{IWSLT}$	$\mathbf{AE}_{IWSLT}$
SAMT	6113	5808	4969	4103
CCG	545	528	456	409
CCG 1-best	800	788	521	434
CF-PSG 1-best	69	69	67	65

Table 3.7: Number of different syntactic labels used by the SAMT system, the CCG-augmented HPB system, the 1-best CCG and CF-PSG systems to annotate the target side of the Chinese–English and Arabic–English news and IWSLT data

the CCG-augmented HPB system to annotate the target side of the Arabic–English and Chinese–English news and IWSLT data. The table also shows that the 1-best CCG systems built on different data sets use significantly more labels than the corresponding 1-best CF-PSG systems.

To obtain a closer look at label sparsity in each system, we calculate the distribution of label frequency counts at frequencies ranging from 10 to 10,000,000 (log scale) in Figures 3.10 and 3.11 for the labels used by each system to annotate the target side of the Arabic–English and Chinese–English news data, respectively. Each column represents the number of labels which occurred a number of times in the training data less than or equal to the number which corresponds to the column label and more than the number which corresponds to the previous column label. Having more labels at low frequencies indicates increased label sparsity. We can see that about 70% of the labels in the SAMT system occur less than 100 times in the Chinese–English and Arabic–English news data. By contrast, about 50% of the labels in our CCG-augmented HPB system occur more than 1000 times in the training data, which means that our CCG-augmented HPB system uses less sparse labels than the SAMT system. The figures also demonstrate that the labels used in the 1-best CCG system are sparser than the labels used in the 1-best CF-PSG system.

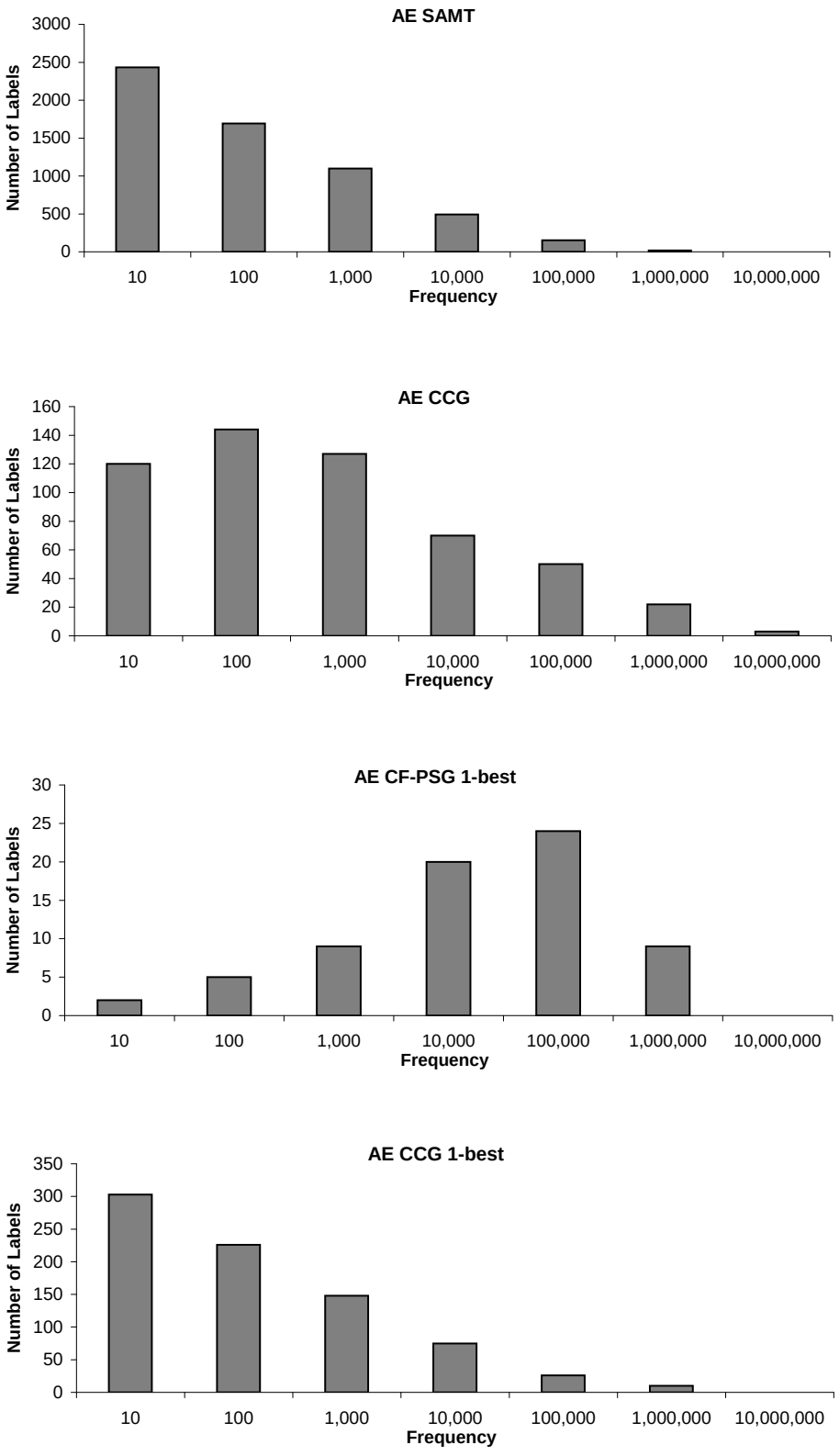

Figure 3.10: Label frequency counts for the CCG-augmented HPB system, the SAMT system, and the 1-best CCG and CF-PSG systems built on the Arabic–English news data.

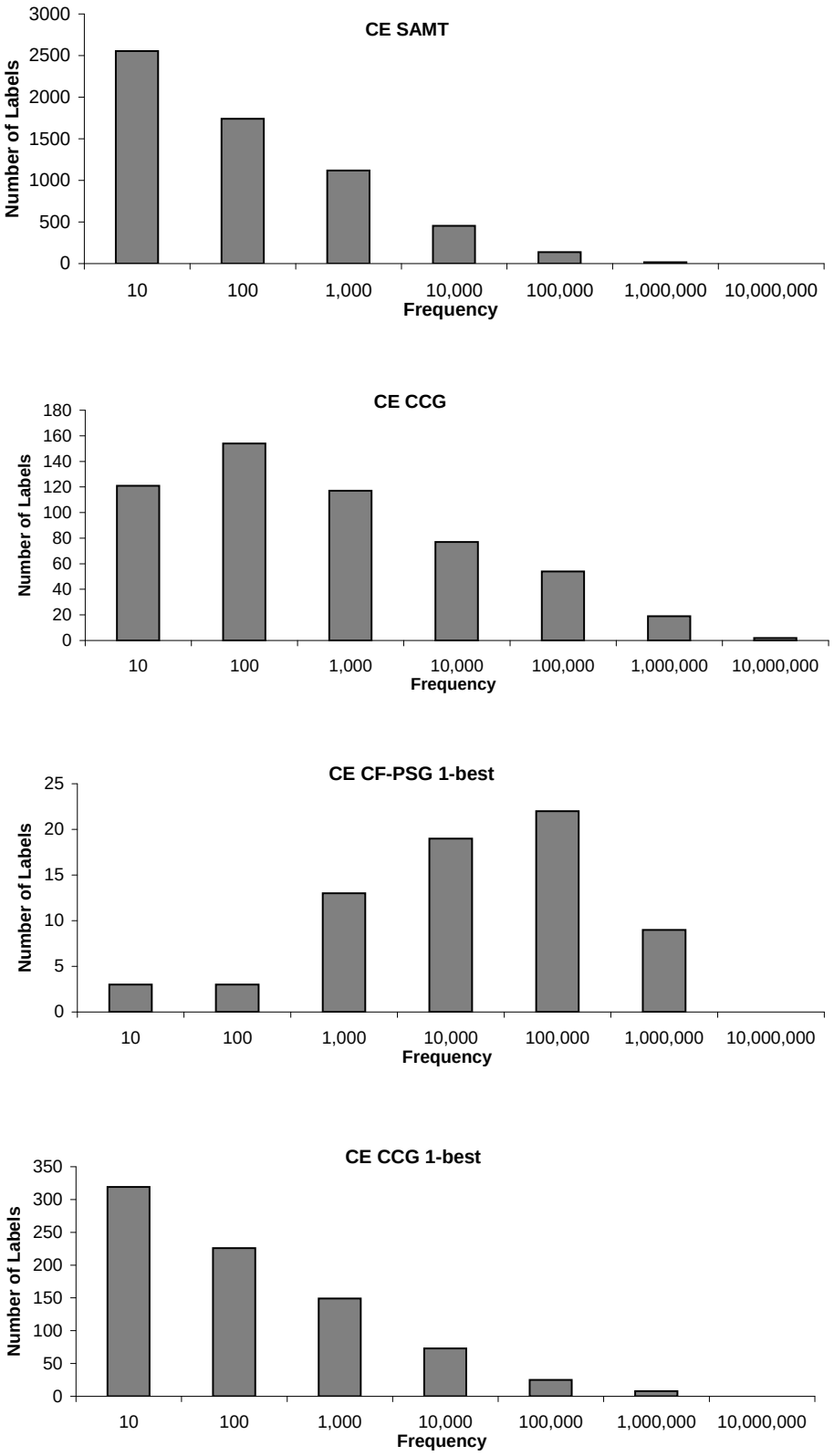

Figure 3.11: Label frequency counts for the CCG-augmented HPB system, the SAMT system, and the 1-best CCG and CF-PSG systems built on the Chinese–English news data.

System	$\mathbf{CE}_{news}$	$\mathbf{AE}_{news}$	$\mathbf{CE}_{IWSLT}$	$\mathbf{AE}_{IWSLT}$
$\text{SAMT}_{phrases}$	70%	69%	50%	44%
$\text{CCG}_{phrases}$	46.81%	47.44%	33%	30%
SAMT_{rules}	71%	55%	64%	58%
CCG_{rules}	60%	52%	53%	49%

Table 3.8: Percentage of target-side X-labelled phrases and rules extracted by the SAMT and CCG-augmented HPB systems built on the Chinese–English and Arabic–English IWSLT and news data.

Label Coverage

We investigate the percentage of phrases which bear the general X label out of the target-side phrases in the Chinese–English and Arabic–English news and IWSLT training data annotated by our CCG-augmented HPB system and the SAMT system. We also calculate the percentage of rules which use the X label on at least one of its target-side nonterminals or on its left-hand side out of the total rules in the translation models of our CCG-augmented HPB system and the SAMT system. Table 3.8 shows that the SAMT system fails to assign a syntactic label to 50% of the total phrase pairs whereas the CCG-augmented HPB system assigns the X label to 33% of the total phrase pairs extracted from the Chinese–English IWSLT training corpus. For the Chinese–English news data, the SAMT system fails to assign a syntactic label to 70% of the total phrase pairs compared to 46.81% of the total phrase pairs assigned the X label by the CCG-augmented HPB system. For Arabic–English data, the figures are similar to the Chinese–English data with the CCG-augmented HPB system succeeding in labelling more phrases that the SAMT system. The table also shows that the SAMT system extracts more rules which use the X label than our CCG-augmented HPB system on the different Arabic–English and Chinese–English data sets. This indicates that our CCG labelling method has better coverage than the SAMT method. This is no doubt due to the flexibility of CCG structures and combinatory rules which help to achieve better coverage than extending the CF-PSG constituent labels using SAMT operators.

We also conduct experiments which examine the effect on performance of ex-

cluding the X-labelled phrases and rules from the translation model of each of the SAMT and CCG-augmented HPB systems built on the Chinese–English and Arabic–English news and IWSLT data. Tables 3.9 and 3.10 show the BLEU, METEOR and TER scores for the resulting systems for Arabic-to-English and Chinese-to-English translation, respectively. When we compare Table 3.9 with Tables 3.2 and 3.3, we can see that excluding the X-labelled rules and phrases from the translation model decreases the performance of both the SAMT and the CCG-augmented HPB systems on the Arabic–English news and IWSLT data. We obtain the same result when comparing Table 3.9 with Tables 3.4 and 3.5 for Chinese–English news and IWSLT data. Thus, discarding the X-labelled rules and phrases has a negative effect on performance, which indicates that they play an important role in translation.

System	**BLEU**	**TER**	**METEOR**
CCG_{news}	22.21	64.52	56.10
SAMT_{news}	22.08	64.76	55.53
CCG_{IWSLT}	51.07	32.87	69.62
SAMT_{IWSLT}	51.18	32.37	69.81

Table 3.9: BLEU, TER and METEOR scores of the CCG-augmented HPB and SAMT systems when excluding X-labelled phrases and rules from the translation model built on the Arabic–English news and IWSLT data.

For Arabic-to-English translation, Table 3.9 shows that excluding X-labelled phrases and rules from the rule table of the CCG-augmented HPB system causes decreases of 0.76 and 1.25 absolute BLEU points, which corresponds to relative decreases of 3.3% and 2.4% on news and IWSLT data, respectively. For the SAMT system, discarding X-labelled phrases and rules leads to 0.75 and 1.15 absolute BLEU points decreases, which corresponds to relative decreases of 3.3% and 2.2% on news and IWSLT data, respectively. Table 3.9 shows that the CCG-augmented HPB system outperforms the SAMT system by 0.13 absolute BLEU points on the news data, which corresponds to a 0.59% relative improvement. This improvement is not statistically significant at p-level=0.05. By contrast, the SAMT system outperforms the CCG-augmented HPB system on the IWSLT data by 0.11 absolute BLEU

points, which corresponds to a 0.21% relative improvement. This improvement is not statistically significant at p-level=0.05.

System	BLEU	TER	METEOR
CCG_{news}	23.18	66.74	52.56
$SAMT_{news}$	22.80	67.27	51.61
CCG_{IWSLT}	47.62	33.86	65.97
$SAMT_{IWSLT}$	45.28	37.24	63.20

Table 3.10: BLEU, TER and METEOR scores of the CCG-augmented HPB and SAMT systems when excluding X-labelled phrases and rules from the translation model built on the Chinese–English news and IWSLT data.

For Chinese-to-English translation, Table 3.10 shows that excluding X-labelled phrases and rules from the rule table of the CCG-augmented HPB system causes decreases of 0.12 and 0.64 absolute BLEU points, which corresponds to relative decreases of 0.5% and 1.3% on news and IWSLT data, respectively. For the SAMT system, discarding X-labelled phrases and rules leads to 0.16 and 0.32 absolute BLEU points decreases, which corresponds to relative decreases of 0.7% and 0.7% on news and IWSLT data, respectively. Table 3.10 shows that the CCG-augmented HPB system outperforms the SAMT system by 0.38 and 2.34 absolute BLEU points, which corresponds to relative improvements of 1.7% and 5.1% on news and IWSLT data, respectively. The improvement achieved on the IWSLT data only is statistically significant.

System	CE_{news}	AE_{news}	CE_{IWSLT}	AE_{IWSLT}
CCG	1046374	1119984	163177	62125
SAMT	890508	946405	143787	58915

Table 3.11: Number of phrases which are annotated by the CCG-augmented HPB system but not by the SAMT system and vice versa.

We also calculate the number of initial phrases in the rule table which are annotated by the CCG-augmented HPB system but not by the SAMT system and vice versa as illustrated in Table 3.11. The table shows that our CCG-augmented HPB system is able to label more initial phrases which the SAMT system fails to

label. For Arabic-to-English news and speech expressions translation, our CCG-augmented HPB system annotates 173579 and 3210 more phrases which the SAMT system fails to label, which constitutes 18.3% and 5.5% relative increase on the phrases annotated by the SAMT system and not by the CCG-augmented HPB system, respectively. For Chinese-to-English news and speech expressions translation, our CCG-augmented HPB system annotates 155866 and 19390 more phrases which the SAMT system fails to label, which constitutes 6.6% and 13.5% relative increases on the phrases annotated by the SAMT system and not by the CCG-augmented HPB system, respectively. This highlights the fact that our CCG-augmented HPB system uses labels with a better coverage than the SAMT system. An example of the phrases which are labelled by our CCG-augmented HPB system but not by the SAMT system is the phrase:

considered a gross mistake for which they should be held accountable

which is extracted from the following sentence:

The deputies said that the ministry's insistence on such a practice is ***considered a gross mistake for which they should be held accountable.***

Our CCG-augmented HPB system assigns the aforementioned phrase S[pss]\NP (which means a past participle verb phrase in passive mode) as a syntactic label, whereas the SAMT system fails to label this phrase.

Manual Analysis

We used BLEU, METEOR and TER automatic MT evaluation metrics to compare the performance of our CCG-augmented HPB system with the SAMT and the HPB baseline systems. Although these metrics provide a fast and convenient way to evaluate the performance of MT systems and have been demonstrated to correlate well with human judgement (Papineni et al., 2002; Snover et al., 2006; Lavie and Abhaya, 2007), they suffer from many shortcomings. BLEU, METEOR and TER evaluate a translation output by the MT system by measuring its resemblance to a human-generated translation of the source sentence (called the 'reference') based on

different criteria. Some data sets have more than one reference for each sentence.[10] However, natural languages are creative and have a large degree of flexibility, which makes it impossible to restrict the correct possible translations of a sentence to a single or even a number of reference translations. Moreover, it has been demonstrated that the effect of the number of references used in tuning with MERT varies among different automatic evaluation metrics (He and Way, 2010). Furthermore, due to efficiency reasons, most automatic evaluation metrics including the ones used in our experiments do not perform any deep syntactic analysis of the translation output in order to evaluate its grammaticality. For these reasons, automatic evaluation metrics have the potential to misjudge the accuracy and adequacy of the translation output, causing better translations according to human evaluators to be sometimes poorly scored by these metrics, and vice versa.

In order to provide a closer look at the strengths and weaknesses of our CCG-augmented HPB system compared to the other HPB systems (the SAMT and HPB baseline systems), we conducted a manual analysis on selected sentence pairs taken from the translation output produced by these systems for the Arabic–English IWSLT test set we used in our experiments. The selection procedure of sentence pairs to undergo manual analysis from the output of the two systems to be compared was performed as follows:

- Sentence-level BLEU scores were calculated for each sentence in the translation output of each system. It is worth noting that BLEU is designed to evaluate translation quality on the corpus level, not on the sentence level. However, we use sentence-level BLEU scores to obtain an approximate estimation of sentence-level translation quality.

- The BLEU score difference was calculated for each sentence pair in the translation outputs of the systems to be compared.

[10]Such as the NIST 2009 Open Machine Translation (OpenMT) Evaluation data set http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC2010T23 and the IWSLT 2010 data set used in our experiments http://iwslt2010.fbk.eu/node/27.

- Sentence pairs whose BLEU score difference was larger than an empirically defined threshold were selected for manual analysis. The threshold was gradually increased until the number of selected sentences is less than 100.

Case	**Number**	**Percentage**
Missing verb	3	10%
Missing translation of word/phrase	7	24%
Wrong reordering of adjective and noun	1	3%
Wrong translation of word/phrase	8	29%
Equally right	3	10%
Equally wrong	6	21%
Better quality	1	3%
Total	29	

Table 3.12: Cases that result from manually comparing higher BLEU score sentences from the output of the CCG-augmented HPB system to their counterparts from the output of the SAMT system.

After selecting sentence pairs from the translation outputs of the systems to be compared according to the previous steps, we divided the sentence pairs into two groups. The first group contains the sentence pairs which have a higher BLEU score for the first system compared with the second system, whereas the second group contains the sentence pairs which have a higher BLEU score for the second system compared with the first system. Then, I manually analysed sentence pairs in each group and classified each sentence pair according to the reason for which the system with the lower BLEU score produced worse translation quality than the other system.

CCG-augmented HPB System vs SAMT System Tables 3.12 and 3.13 show the results of the manual analysis conducted on sentence pairs from the output of the SAMT and CCG-augmented HPB systems which have a BLEU score difference larger than a threshold equal to 0.1. We obtained 29 sentences for which the CCG-augmented HPB system outperformed the SAMT system and 36 sentences in the reverse case, which constitute 5.7% and 7% of the total sentences in the test set, respectively. This means that less than 13% of the total sentence pairs from the

output translations have a BLEU sore difference larger than 0.1, which indicates that the SAMT and CCG-augmented HPB systems achieve similar BLEU score for most sentences in the test set. While Table 3.12 shows the percentage of different cases found for sentences which have a higher BLEU score for the CCG-augmented HPB system compared to their counterparts from the SAMT system, Table 3.13 shows the reverse case. The 'equally right' case means that both sentences are correct despite the difference in their BLEU score. The 'equally wrong' case means that both sentences are wrong despite the difference in their BLEU score. 'Better quality' means that the sentence which has the lower BLEU score has a better quality than the sentence with the higher BLEU score.

Source: هل هناك محل هدايا ؟
Reference: *is there a gift shop ?*
SAMT: is there a gift ?
CCG: is there a gift shop ?

Figure 3.12: An example of a missing word in the output of the SAMT system, which was captured by the CCG-augmented HPB system.

Table 3.12 shows that the main cases in which the CCG-augmented HPB system produces better translation quality are when words/phrases are translated wrongly by the SAMT system (29% of the total cases) in addition to capturing the translation of words/phrases missed by the SAMT system (24% of the total cases). Figure 3.12 shows an example in which the CCG-augmented HPB system was able to capture the translation of the Arabic word محل, corresponding to the English word *shop*, while the SAMT system missed it. Figures 3.13 and 3.14 show the derivation trees of the translation produced by the SAMT and the CCG-augmented HPB systems for the same sentence.[11] Figure 3.13 shows that the SAMT system uses a glue grammar rule to combine the phrases *is there a* and *gift*. The unaligned Arabic word محل

[11] Q in the derivation trees replaces the S symbol in the original glue grammar rules (cf. glue grammar rules (2.8) and (2.9) page 24) to avoid confusion with the S symbol used in syntactic constituents.

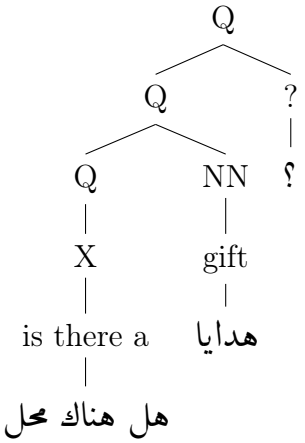

Figure 3.13: The derivation tree of the English translation produced by the SAMT system for the Arabic sentence هل هناك محل هدايا ؟.

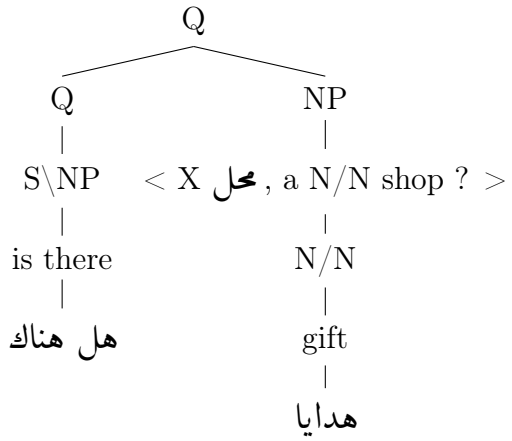

Figure 3.14: The derivation tree of the English translation produced by the CCG-augmented HPB system for the Arabic sentence هل هناك محل هدايا ؟.

Source: هل يمكنك تسليمها بحلول الغد ؟
Reference: *can you deliver it by tomorrow ?*
SAMT: could you hand it by tomorrow ?
CCG: could you delivered them by tomorrow ?

Figure 3.15: An example of a wrong translation of a word produced by the CCG-augmented HPB system.

at the edge of the first phrase is what caused the omission of the word *shop* from the output translation. The derivation tree in 3.14 shows that the CCG-augmented HPB system was able to capture the word *shop* using the hierarchical rule $NP \rightarrow <$ X محل$, a\ N/N\ shop\ >$.

Case	**Number**	**Percentage**
Missing verb	0	0 %
Missing translation of word/phrase	5	14%
Wrong reordering of adjective and noun	2	5%
Wrong translation of word/phrase	6	17%
Equally right	10	28%
Equally wrong	6	17%
Better quality	7	19%
Total	36	

Table 3.13: Cases that result from manually comparing higher BLEU score sentences from the output of the SAMT system to their counterparts from the output of the CCG-augmented HPB system.

Table 3.13 shows that the main cases in which the SAMT system produces better translation quality than the CCG-augmented HPB system are when capturing the translation of words/phrases missed by the CCG-augmented HPB system (14% of the total cases), and correctly translating a word/phrase wrongly translated by the CCG-augmented HPB system (17% of the total cases). Figure 3.15 shows an example in which the CCG-augmented HPB system uses the right verb but in the wrong tense (*delivered* instead of *deliver*) in the output translation. The CCG-augmented HPB system also wrongly translates the object pronoun to *them* instead of *it*. The SAMT system produces a correct translation by using another verb *hand* in the correct from, and correctly translating the object pronoun to *it*.

Comparing Table 3.12 with Table 3.13, we can see that the SAMT system misses

the insertion of a verb in the output translation in 10% of the examined sentences, while the CCG-augmented HPB system does not commit this error in any of the examined sentences. This could be one way of improvement that the CCG-augmented HPB system achieved over the SAMT system. We have demonstrated that the SAMT system produces sparser and less coverage syntactic labels than our CCG-augmented HPB system (cf. Tables 3.7 and 3.8), which limits the coverage of the syntactic constraints applied during decoding and leads to the production of ungrammatical translations. Figure 3.16 shows an example of a missing verb *is* and its subject *there* in the translation output of the SAMT system. For the same example, the CCG-augmented HPB system produced the verb and its subject in a correct translation, which perfectly matches the reference.

Source: هل تعتقد باحتمال أن أصاب بدوار البحر ؟
Reference: *do you think there 's a chance i 'll get seasick ?*
SAMT: do you think a chance i 'll get seasick ?
CCG: do you think there 's a chance i 'll get seasick ?

Figure 3.16: An example of a missing verb in the output of the SAMT system. The CCG-augmented system captures the verb translation.

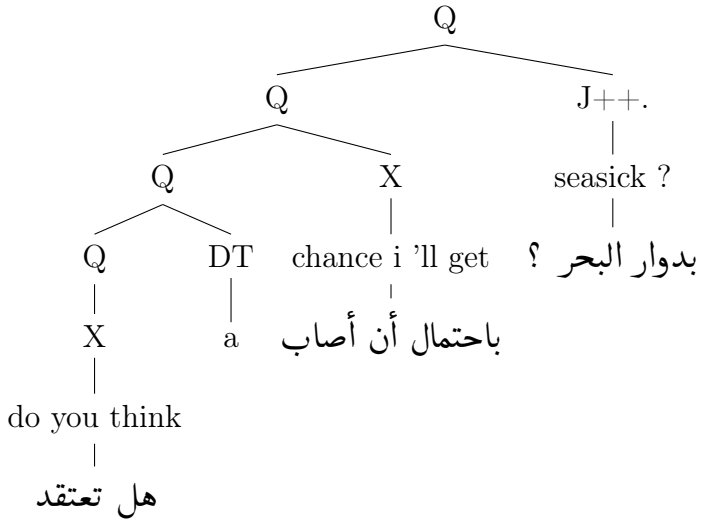

Figure 3.17: The derivation tree of the English translation produced by the SAMT system for the Arabic sentence هل تعتقد باحتمال أن أصاب بدوار البحر ؟.

Figures 3.17 and 3.18 show the derivation trees of the translation produced by each of the SAMT and the CCG-augmented HPB systems for the same sentence,

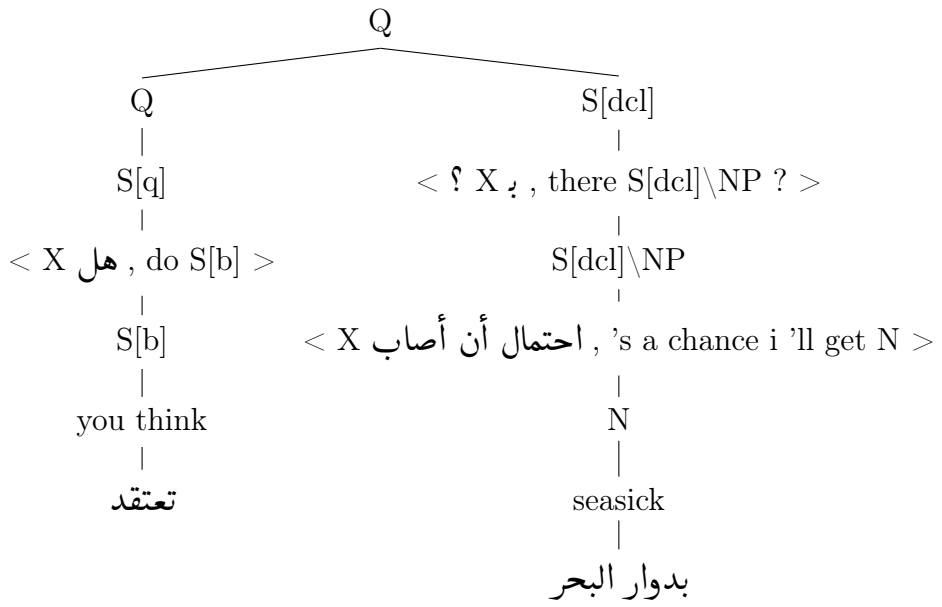

Figure 3.18: The derivation tree of the English translation produced by the CCG-augmented HPB system for the Arabic sentence هل تعتقد باحتمال أن أصاب بدوار البحر ؟.

respectively. Figure 3.17 shows that the SAMT system was not able to assign syntactic labels to the phrases *do you think* and *chance i'll get*. Furthermore, the SAMT system used glue grammar rules to assemble the phrases of the output translation. Thus, the SAMT system did not use any syntactic constraint to build the translation in this example, which led to the production of ungrammatical translation. By contrast, Figure 3.18 shows that the CCG-augmented HPB system uses CCG-augmented hierarchical rules to build the two parts of the translation: the phrases *do you think* and *there's a chance i'll get seasick*, which means that CCG-based syntactic constraints are used to produce these phrases. Then, a glue grammar rule is used to concatenate these two phrases. Thus, the CCG-augmented HPB system was able to apply syntactic constraints more widely during translation, which helped to produce a grammatical translation. It is worth noting here that CCG effect on improving the grammaticality of the translation by helping to insert verbs missed by other systems was also demonstrated by Hassan (2009). Hassan (2009) conducted a manual analysis which demonstrated that one of the main reasons for which CCG helped to improve the grammaticality of the translation produced by the PB SMT

system was inserting verbs omitted by the PB SMT baseline system.

Source: متى يمكنك احضارها ؟
Reference: *when can you bring it ?*
SAMT: when can you bring it ?
CCG: what time will you bring it ?

Figure 3.19: An example of equally right translations produced by the SAMT and CCG-augmented HPB systems. The SAMT system output has a higher BLEU score than that of the CCG-augmented HPB system.

Interestingly, the sentence pairs which have equivalent translation quality according to the manual analysis constitute 45% of the total sentence pairs which have a higher BLEU score for the SAMT system compared with the CCG-augmented HPB system (cf. Table 3.13). By contrast, 31% of the total sentence pairs which have a higher BLEU score for the CCG-augmented HPB system compared with the SAMT system have equivalent translation quality according to the manual analysis (cf. Table 3.12). Figure 3.19 shows an example of equally right translations produced by the CCG-augmented HPB and SAMT systems, although the SAMT output has a higher BLEU score.

Source: أرغب في مشاهدة مسرحية .
Reference: *i 'd like to see a play .*
SAMT: i 'd like to see a .
CCG: i 'd like to see a doctor .

Figure 3.20: An example of wrong translations produced by the SAMT and CCG-augmented HPB systems. The SAMT system output has a higher BLEU score than that of the CCG-augmented HPB system.

Figure 3.20 shows an example of equally wrong translations produced by the CCG-augmented HPB and SAMT systems despite of the fact that the SAMT system has a higher BLEU score. Furthermore, 19% of the total sentence pairs which have a higher BLEU score for the SAMT system achieve better translation quality for the CCG-augmented HPB system according to the manual analysis. By contrast, only 3% of the total sentence pairs which have a higher BLEU score for the CCG-augmented HPB system achieve better translation quality for the SAMT

Source: أين يمكنّي الحصول على عربة حقائب ؟
Reference: *where can i get a luggage cart ?*
SAMT: where can i get a bags ?
CCG: where can i get my baggage cart ?

Figure 3.21: An example of a better translation produced by the CCG-augmented HPB system compared to that of the SAMT system. The CCG-augmented HPB system has a lower BLEU score.

system according to the manual analysis. This could indicate a margin of improvement in the actual translation quality achieved by the CCG-augmented HPB system over the SAMT system although they have almost an equal BLEU score on the Arabic–English IWSLT test set (cf. Table 3.3). Moreover, this sheds light on the insufficiency of the BLEU metric, which is one of the most widely used automatic MT evaluation metrics, to evaluate the real translation quality of SMT systems. Figure 3.21 shows an example in which the CCG-augmented HPB system produces a better translation than the SAMT system despite having a lower BLEU score.

Source: أين يقع أقرب مطعم مكسيكي ؟
Reference: *where is the closest mexican restaurant ?*
SAMT: where is the nearest restaurant mexican ?
CCG: where 's the nearest mexican restaurant ?

Figure 3.22: An example of a wrong reordering of an adjective and a noun in the output of the SAMT system. The CCG-augmented HPB system captures the right order.

From Tables 3.12 and 3.13 we can also see that the SAMT and CCG-augmented HPB systems err almost equally in reordering adjectives and nouns, which have opposing order between Arabic and English. Figure 3.22 shows an example of a wrong reordering between an adjective and a noun in the output of the SAMT system. The CCG-augmented HPB system produces the correct reordering. Figures 3.23 and 3.24 show the derivation trees produced by the SAMT and CCG-augmented HPB systems for the same example, respectively. We can see that the wrong reordering of the adjective and noun in the output of the SAMT system is due to the use of a glue grammar rule to concatenate the adjective *mexican* and the noun *restaurant*.

As glue grammar rules do not perform any reordering, the order of the adjective and noun is wrong in the SAMT output. By contrast, we can see from Figure 3.24 that the use of the hierarchical rule $N \rightarrow <$ *X* مطعم , *N/N restaurant* $>$, which swaps the positions of the adjective and the noun, is what helped the CCG-augmented HPB system to produce the right translation.

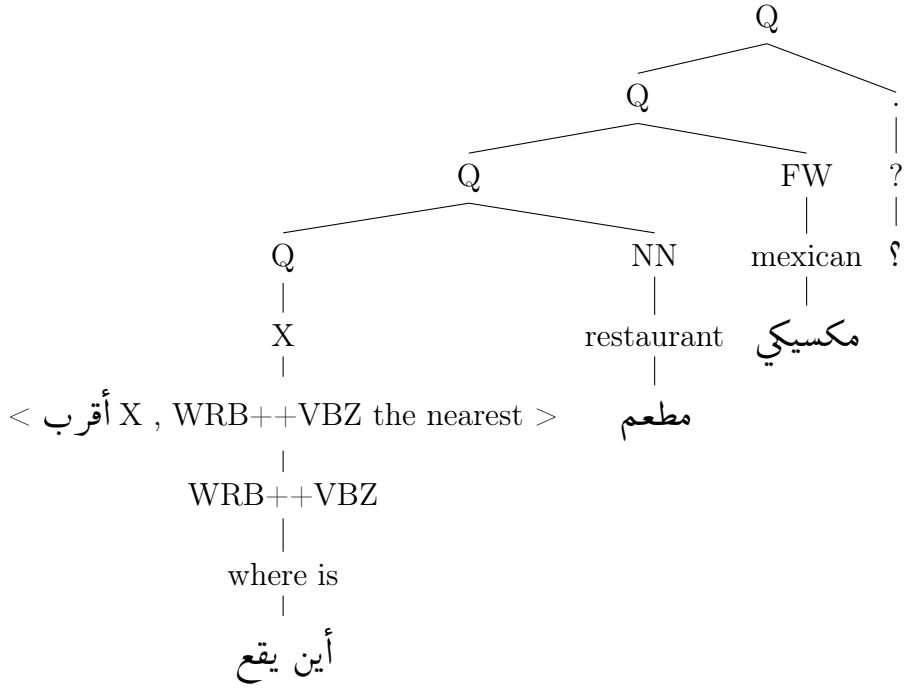

Figure 3.23: The derivation tree of the English translation produced by the SAMT system for the Arabic sentence ؟ أين يقع أقرب مطعم مكسيكي.

The CCG-augmented HPB System vs the HPB Baseline System Tables 3.14 and 3.15 show the results of the manual analysis conducted on sentence pairs from the output of the CCG-augmented HPB system and the HPB baseline system which have a BLEU score difference larger than a threshold equal to 0.2. We obtained 27 sentences for which the CCG-augmented HPB system outperformed the HPB baseline system and 26 sentences in the reverse case, which totally constitute 10.5% of the total sentences in the test set. Table 3.14 shows the percentage of different cases found for sentence pairs which have a higher BLEU score for the CCG-augmented HPB system compared with their counterparts from the HPB baseline

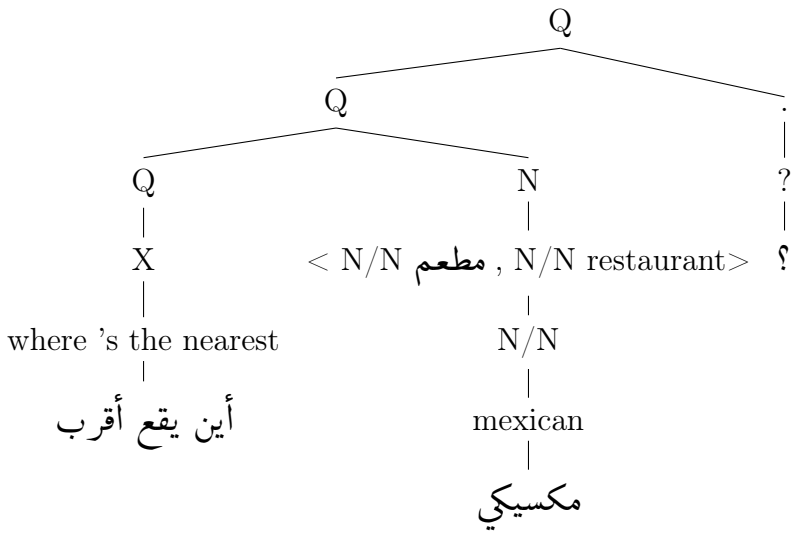

Figure 3.24: The derivation tree of the English translation produced by the CCG-augmented HPB system for the Arabic sentence ؟ أين يقع أقرب مطعم مكسيكي.

system. Table 3.15 shows the reverse case.

From Table 3.14 we can see that missing translation of a word/phrase and wrongly translating a word/phrase are the main faults committed by the HPB baseline system compared to the CCG-augmented HPB system, constituting 26% and 19% of the total cases, respectively.

Case	**Number**	**Percentage**
Missing verb	0	0%
Missing translation of word/phrase	7	26%
Wrong reordering of adjective and noun	1	4%
Wrong translation of word/phrase	5	19 %
Equally right	4	15%
Equally wrong	7	26%
Better quality	3	10 %
Total	27	

Table 3.14: Cases that result from manually comparing higher BLEU score sentences from the output of the CCG-augmented HPB system to their counterparts from the output of the HPB baseline system.

Table 3.15 shows that wrongly translating a word/phrase by the CCG-augmented HPB system is the main reason why the HPB baseline system produces better translation quality than the CCG-augmented HPB system, which constitutes 50%

Case	Number	Percentage
Missing verb	0	0%
Missing translation of word/phrase	0	0%
Wrong reordering of adjective and noun	1	4%
Wrong translation of word/phrase	13	50 %
Equally right	4	15 %
Equally wrong	7	27 %
Better quality	1	4%
Total	26	

Table 3.15: Cases that result from manually comparing higher BLEU score sentences from the output of the HPB baseline system to their counterparts from the output of the CCG-augmented HPB system.

of the total cases. Figure 3.25 shows an example of a wrong translation of a word produced by the CCG-augmented HPB system. In this example, the Arabic verb ترينا was wrongly translated as *get* in the output of the CCG-augmented HPB system, whereas it was correctly translated to *show us* in the output of the HPB baseline system.

Source: هل ترينا مقاعدنا ؟
Reference *could you show us to our seats ?*
CCG: do we get our seats ?
HPB: will you show us to our seats ?

Figure 3.25: An example of wrong translation of a word in the output of the CCG-augmented HPB system. The HPB baseline system produces the right translation.

Comparing Tables 3.14 and 3.15, we can see that while the HPB baseline system omits the translation of a word/phrase in 26% of the total cases compared to the CCG-augmented HPB system, the CCG-augmented HPB system does not commit this fault in any of the studied cases. This means that the CCG-augmented HPB system was able to avoid this weakness of the HPB baseline system. Figure 3.26 shows an example of a missing word *key* in the output of the HPB baseline system. The CCG-augmented HPB system succeeds in capturing this word in its translation output. As we saw when comparing the CCG-augmented HPB system with the SAMT system, unaligned words in addition to wrongly aligned words are the main reasons which cause erroneous word deletions in the translation output. However,

the syntactic constraints imposed by the CCG-augmented HPB systems help to recover from some of these errors as missing the translation of a word is likely to affect the grammaticality of the translation output. As it was found when manually analysing sentence pairs from the outputs of the CCG-augmented HPB system and the SAMT system, both the CCG-augmented HPB system and the HPB baseline system mistake equally in reordering adjectives and nouns. This emphasises the wrong reordering of adjectives and nouns between Arabic and English as a common weakness of the three studied systems.

Source: . ليس هذا هو المفتاح الصحيح
Reference: *this is the wrong key .*
HPB: this isn 't the right .
CCG: this isn 't the right key .

Figure 3.26: An example of a missing word in the output of the HPB baseline system. The CCG-augmented HPB system captures the word translation.

3.6 Conclusions

In this chapter, we described our approach to augmenting the HPB model with syntactic information extracted using CCG. We demonstrated the advantages of using CCG categories to label nonterminals in hierarchical rules over CF-PSG-based labels. CCG has a flexible structure that results from its ability to combine categories using a set of simple combinatory rules. This flexibility enabled our CCG-based labelling method to extract syntactic labels for many phrases which do not necessarily correspond to grammatical constituents in traditional terms. Our experiments demonstrated that our CCG-based labelling approach was able to provide a significantly better coverage for phrases than the SAMT labelling method. Furthermore, CCG-based labels were demonstrated to be less sparse than the SAMT labels, leading to a smaller translation model and more reliable rule probabilities. Most importantly, CCG category labels provide richer and more accurate syntactic information than SAMT labels, because CCG labels precisely describe the dependents and lo-

cal context of the words and phrases without being redundant. Our experiments also showed that our CCG-augmented HPB system outperformed the SAMT system in terms of BLEU score on all the Chinese-to-English and Arabic-to-English experiments, except for the Arabic-to-English speech expressions translation where the two systems performed similarly. However, our CCG-augmented HPB system was not able to outperform the HPB baseline system on most of the conducted Arabic-to-English and Chinese-to-English experiments. We believe this to be due to the sparseness of the CCG-based nonterminal labels which restrict the search space during translation, which in turn negatively affects the translation quality. We also conducted experiments which compare the performance of the systems which use CCG and CF-PSG labels extracted from the 1-best parse trees of the target side of the training data. Our experimental results demonstrated that the 1-best CCG system slightly outperformed the 1-best CF-PSG system for Arabic-to-English and Chinese-to-English translation. We believe that this is due to fact that CCG categories represent richer syntactic information than CF-PSG constituent labels, which helps to impose more discriminative syntactic constraints. However, the 1-best CCG was not able to outperform our CCG-augmented HPB system which uses CCG labels extracted from the parsing chart. This is because the 1-best CCG system does not take advantage of the flexibility of CCG structures, which enables to extract more annotated phrases and rules than the 1-best CCG system.

Our manual analysis gave insights into the strengths and weaknesses of our CCG-augmented HPB system compared with the HPB baseline and SAMT systems beyond what could be gleaned from automatic evaluation metrics. The manual analysis demonstrated that our CCG-augmented HPB system was better at inserting verbs in the output translation than the SAMT system. Furthermore, our CCG-augmented HPB system succeeded in avoiding a major weakness of the HPB SMT system, namely the failure to translate words/phrases. However, compared to the HPB baseline system, the main weakness of our CCG-augmented HPB system was its failure to translate some words/phrases correctly.

The research conducted in this chapter addresses the first research question (**RQ1**) by demonstrating that CCG is better than CF-PSG when these grammar formalisms are used to extract labels for nonterminals in the HPB SMT model. In the next chapter, we try to explore ways to enhance the performance of our CCG-augmented HPB system by investigating different CCG-based nonterminal labelling approaches which try to tackle the label sparsity problem by extracting less fine-grained CCG-based nonterminal labels.

Chapter 4

Simplifying CCG-based Nonterminal Labels

In the previous chapter, we demonstrated that using CCG categories to label nonterminals in the HPB SMT model helped to achieve better translation quality than the SAMT system. We also demonstrated that the CCG-based labels used in our CCG-augmented HPB system were less sparse, have better coverage and hold richer syntactic information than the SAMT labels. However, our CCG-augmented HPB system was not able to outperform the HPB baseline system in many of the experiments. We believe that this might be due to the sparsity of our CCG-based labels, which leads to rule probability fragmentation and imposes restrictions on the decoding search space. In this chapter, we try to tackle the label sparsity problem which affects the performance of our CCG-augmented HPB system. We present two approaches to extracting less sparse but still rich CCG-based nonterminal labels. Through the research conducted in this chapter we try to verify the effectiveness of our approaches on improving the performance of our CCG-augmented HPB system, which answers our second research question (**RQ2**).

Section 4.1 introduces our motivation to explore approaches towards the simplification of nonterminal labelling. Section 4.2 introduces related work. Section 4.3 presents our first label simplification approach which is based on extracting contex-

tual information presented in CCG categories to label nonterminals in hierarchical rules. In Section 4.4, we explain the second label simplification approach which removes features held by some CCG categories from the nonterminal label representation. We present our experiments in Section 4.5 which examine the effect on performance of the two simplification approaches both individually and combined under different factors: the language pair, the size of the training data, the size of the language model and the domain of the data. Section 4.6 presents the conclusions for this chapter.

4.1 Motivation

With the emergence of different approaches to incorporating syntax in hierarchical SMT systems, not only different grammar formalisms have been explored in the syntax augmentation process, but also different nonterminal labelling approaches of different degrees of granularity. The use of a small set of different nonterminal labels guarantees a manageable increase in the size of the translation model and thus a faster decoding process than in the case of a larger set of nonterminal labels. However, coarse nonterminal labels are not expressive enough to impose the syntactic restrictions required to prevent the production of ungrammatical translations. On the other hand, a large number of different nonterminal labels leads to the generation of large translation models with sparse rule probabilities, which slows down the decoding process and negatively affects translation quality. Furthermore, attaching different labels to the same phrase leads to spurious ambiguity, which means that different labellings of the same translation will compete with each other during decoding, which also affects translation quality in a negative way.

In Chapter 3 we demonstrated that using CCG categories as nonterminal labels is better than CF-PSG-based labels extracted in the SAMT system. We demonstrated that CCG categories provide richer and more accurate syntactic descriptions, which are able to annotate more phrases than the SAMT method while at the same time

being less sparse. Our experiments showed that our CCG-augmented HPB system was able to outperform the SAMT system in most of the experiments. However, our experiments showed that our CCG-augmented HPB system was not able to outperform the HPB baseline system. We believe that the richness of CCG-based labels can negatively affect performance, which results from imposing strict syntactic constraints on phrases replacing nonterminals during decoding. We propose a solution to this problem based on simplifying CCG categories used to label nonterminals in hierarchical rules by employing part of the information represented in CCG categories. This makes the labels less fine-grained, and loosens the strong syntactic constraints held by them. This softening method comes at the expense of the accuracy of the syntactic labels. Nevertheless, we want to examine the effect of the trade-off between the richness of the labels and the flexibility of the syntactic constraints they impose, which provides an answer to our second research question (**RQ2**).

We suggest two simplification schemes. The first scheme employs only the contextual information presented in a CCG category and ignores its resulting category in the representation of the syntactic label (cf. Section 4.3). The second scheme drops the features held by some of the CCG categories from the representation of the syntactic label (cf. Section 4.4). We present experiments which examine the effect of applying each of the two schemes both individually and combined under different factors. We use data sets from different domains and sizes for Arabic-to-English, Chinese-to-English and French-to-English translation. We also examine the effect of the size of the language model on the performance of our systems.

4.2 Related Work

Wang et al. (2010) argue that nonterminal labels which correspond to the Penn Treebank tag set are too coarse to represent accurate syntactic constraints imposed on phrases replacing nonterminals during decoding, which leads to the production

of ungrammatical translations. As a solution, they explore the effect of applying two relabelling approaches borrowed from monolingual parsing to split the Penn Treebank-based nonterminal labels into more fine-grained labels by representing more contextual information in them. The first relabelling approach is linguistically motivated (Johnson, 1998; Klein and Manning, 2003). It incorporates information about the parent, head word and siblings tags into the nonterminal label. The second relabelling approach is statistically motivated (Matsuzaki et al., 2005; Petrov et al., 2006). It performs automatic category splitting which learns to subcategorize nonterminals using Expectation Maximization. They demonstrate that both relabelling approaches help to improve performance over the baseline systems. Hanneman and Lavie (2011) propose an approach to extracting coarse-grained nonterminal labels. Their approach merges nonterminal labels using information extracted from source- and target-side parse trees. They use a metric to calculate the distance between two syntactic labels in one language based on the difference in their alignment probabilities with labels from the other language. They then apply a greedy label merging algorithm which merges two labels with the closest distance calculated according to the distance metric. Zollmann and Vogel (2011) label phrases and nonterminals in the HPB SMT model with labels extracted from clustering phrases according to their feature vectors. The feature vectors contain information about the lexical descriptors of the words within the phrase in addition to the lexical descriptors of the boundary words surrounding the phrase.

In contrast to the label splitting approach followed by Wang et al. (2010), we try to extract less fine-grained CCG-based nonterminal labels in order to reduce the sparsity of nonterminal labels based on CCG categories, which are much richer than the Penn Treebank tag set. While the label coarsening approach followed by Hanneman and Lavie (2011) depends on source- and target-side parse trees, we use CCG categories extracted form target-side parse trees of the training data as a basis to extract less sparse CCG-based labels. Zollmann and Vogel (2011) incorporate contextual information based on POS tags and word-based classes extracted via

unsupervised word clustering into nonterminal label extraction. In contrast, we use contextual information already presented in CCG categories to extract CCG-based nonterminal labels. CCG contextual categories present richer contextual information than POS tags. Furthermore, CCG contextual labels are syntax-based in contrast to the syntax-free word-based classes obtained via unsupervised learning.

4.3 CCG Contextual Labels

A CCG category describes rich syntactic information about the upper levels of the syntactic structure in which the word/phrase participates. Furthermore, the same word/phrase would have multiple CCG categories according to the different contexts within which the word/phrase might appear. That is the reason for the proliferation of different CCG categories. In our first label simplification approach, we try to employ the information about the syntactic context of the word/phrase reflected by its CCG category in a syntactic label, which we call the CCG contextual label, and ignore other information in the CCG category. A CCG category takes the form of:

$$C = (R \backslash A_l) / A_r$$

where A_l represents the left argument category, A_r represents the right argument category, and R represents the resulting category. In the contextual label representation, we include the left and right argument categories A_l and A_r of C and exclude the resulting category R from the label representation. Thus, a syntactic label of the form $A_l\_A_r$ is extracted from the category C. If any of the left or right argument categories in C does not exist, this argument is replaced with the X symbol in the CCG contextual label. The labels that consist of only the left and right arguments of CCG categories still express important syntactic information about words/phrases but are simpler and thus less sparse than complete CCG categories.

To extract hierarchical rules annotated with CCG contextual labels, we first

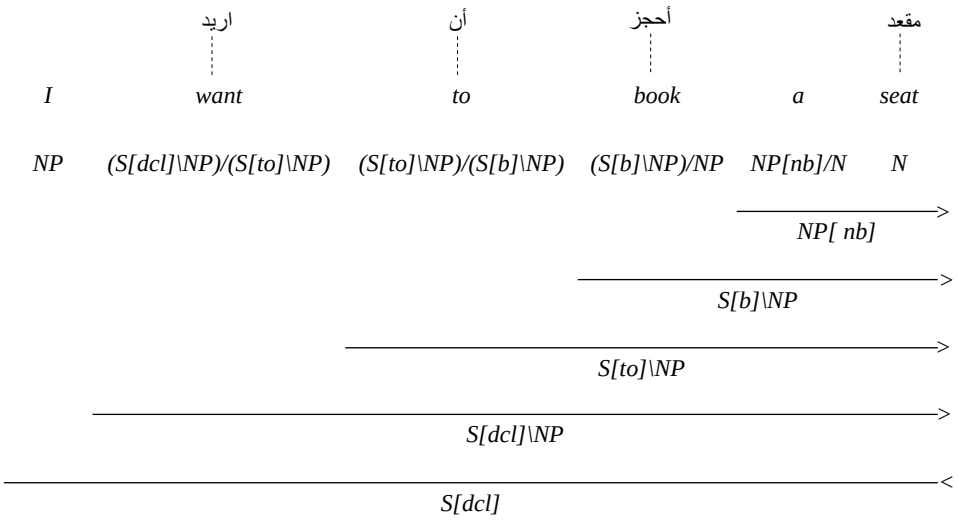

Figure 4.1: An Arabic source sentence and its aligned English translation along with its CCG parse tree.

annotate the target side of the training corpus with CCG categories as described in Seciton 3.3. Then, for each target-side phrase in the training corpus assigned C as its CCG category, we extract the contextual label corresponding to C and assign it to the phrase. Lastly, hierarchical rules annotated with CCG contextual labels are extracted from the training corpus according to the same basic hierarchical rule extraction algorithm (cf. Section 2.4.1 page 21). Figure 4.2 shows a set of phrases extracted from the Arabic–English sentence pair illustrated in Figure 4.1 along with the CCG category and CCG contextual labels assigned to them. For example, the CCG category assigned to the phrase *want to book* is $(S[dcl]\backslash NP)/NP$. This CCG category has $S[dcl]$ as a functor, NP as a right argument and NP as a left argument. Therefore, we use NP_NP as the CCG contextual label assigned to this phrase.

CCG Category Label	*CCG Context Label*	*α*	*β*
S[dcl]/(S[to]\NP)	*X_ S[to]\NP*	أريد	*I want*
(S[dcl]\NP)/(S[b]\NP)	*NP_S[b]\NP*	أريد أن	*want to*
NP[nb]	*X_X*	مقعد	*a seat*
S[b]\NP	*NP_X*	أحجز مقعد	*book a seat*
S[to]\NP	*NP_X*	أن أحجز مقعد	*to book a seat*
(S[dcl]\NP)/NP	*NP_NP*	أريد أن أحجز	*want to book*
S[dcl]	*X_X*	أريد أن أحجز مقعد	*I want to book a seat*

Figure 4.2: A set of phrases extracted from the Arabic–English sentence pair in Figure 4.1 along with the CCG category and CCG contextual labels assigned to them.

4.4 Feature-stripped CCG Labels

Some CCG categories bear features which describe certain syntactic information such as the type of the sentence and the verb phrase. For example the atomic category S might have a feature attached to it (described between two brackets after the category symbol) which distinguishes types of sentences such as declarative $S[dcl]$ or wh-question $S[wq]$. Verb phrases also might hold features such as to infinitival $S[to]\backslash NP$, bare infinitival $S[b]\backslash NP$, and past participle $S[pt]\backslash NP$. These features attached to some CCG categories are similar to latent variables attached to nonterminal symbols in the probabilistic CFG in what is called the probabilistic CFG with latent annotation (Matsuzaki et al., 2005; Petrov et al., 2006).

We examine the effect on performance of removing these features from CCG category and CCG contextual labels used to label nonterminals in the HPB model. By removing these features, we intend to further reduce the total number of different syntactic labels used in the CCG-augmented HPB systems, and accordingly the size of the extracted grammar. Figure 4.3 shows the feature-stripped CCG category and CCG contextual labels assigned to some phrase pairs extracted from the sentence

Feature-stripped CCG Category Label	*Feature-stripped CCG Context Label*	*α*	*β*
S/(S\NP)	*X_ S\NP*	أريد	*I want*
(S\NP)/(S\NP)	*NP_S\NP*	أريد أن	*want to*
NP	*X_X*	مقعد	*a seat*
S\NP	*NP_X*	أحجز مقعد	*book a seat*
S\NP	*NP_X*	أن أحجز مقعد	*to book a seat*
(S\NP)/NP	*NP_NP*	أريد أن أحجز	*want to book*
S	*X_X*	أريد أن أحجز مقعد	*I want to book a seat*

Figure 4.3: A set of phrases extracted from the Arabic–English sentence pair in Figure 4.1 along with the feature-stripped CCG category and CCG contextual labels assigned to them.

pair in Figure 4.1. The next section presents the experiments which examine the effect of applying this simplification method on each of CCG category and CCG contextual labels.

4.5 Experiments

In this section, we present our experiments which compare the performance of our CCG-augmented HPB systems which use different CCG-based labelling approaches with PB and HPB baseline systems on Arabic-to-English, Chinese-to-English and French-to-English translation. We examine the performance of the different labelling approaches using small and large data sets. We also explore the effect of using larger language models on the performance of the different systems. Furthermore, we examine how the different systems perform on data from different domains.

4.5.1 Data and Settings

For Arabic-to-English and Chinese-to-English small data set experiments, we use the same data we used in our experiments in Section 3.5.1 (page 61). Additionally, we examine the performance of the systems on a 60k sentence pair data set randomly selected from the Arabic–English UN corpus (Eisele and Chen, 2010) as an additional small training data set. For the Chinese-to-English large data set experiments, we use a training data of 223k sentence pairs randomly selected from the FBIS corpus. The development and test sets are of 1200 sentence pairs each. For Arabic-to-English large data set experiments, we use 223k sentence pairs randomly selected from the Arabic–English UN corpus to train our systems. The development and test sets are of 1000 sentence pairs each. For the French-to-English small data set experiments, we use 60k sentence pairs selected randomly from the Europarl corpus (Koehn, 2005). For the French-to-English large data set experiments, we use 223k sentence pairs selected randomly from the same corpus. The French–English development and test sets are of 1000 sentence pairs each. It is worth mentioning that the small Arabic–English, Chinese–English and French–English data sets are parts of their corresponding big data sets. Table 4.1 summarises data sets (apart from the sets used in Section 3.5.1) used in our experiments.

Corpus	**Training set**	**Development set**	**Test set**
AE $_{small\ data}$	60k	1000	1000
AE $_{large\ data}$	223k	1000	1000
CE $_{small\ data}$	51k	1200	1200
CE $_{large\ data}$	223k	1200	1200
FE $_{small\ data}$	60k	1000	1000
FE $_{large\ data}$	223k	1000	1000

Table 4.1: Data used in our experiments.

The experimental settings used in our experiments in this chapter are the same settings used in our experiments in Section 3.5.1 (page 61).

4.5.2 Baseline Systems

We build our HPB baseline using the Moses Chart Decoder[1] (Hoang et al., 2009b) with maximum phrase length and maximum rule span set to 12 words. Hierarchical rules extracted contain up to 2 nonterminals. Maximum chart span is set to 20 words. The cube pruning pop-up limit is set to 1000. The PB baseline system is built using the Moses Phrase-Based Decoder (Koehn et al., 2007) with maximum phrase length set to 12 words.

4.5.3 CCG-Augmented Systems

We build the following CCG-augmented HPB systems in our experiments:

- **CCG Context:** uses CCG contextual labels as nonterminals labels in hierarchical rules (cf. Section 4.3).
- **CCG:** uses CCG categories as nonterminals labels in hierarchical rules (cf. Section 3.3).
- **CCG Context (s):** uses feature-stripped CCG contextual labels as nonterminal labels in hierarchical rules (cf. Section 4.4).
- **CCG (s):** uses feature-stripped CCG categories as nonterminal labels in hierarchical rules (cf. Section 4.4).

We use the CCG parser from C&C tools[2] (Clark and Curran, 2007) to parse the English side of the training data from which we extract the CCG-augmented HPB models. We use the Moses Chart Decoder to build our CCG-augmented HPB systems with the same settings as the HPB baseline system.

[1] http://www.statmt.org/moses/?n=Moses.SyntaxTutorial
[2] http://svn.ask.it.usyd.edu.au/trac/candc/

System	BLEU	TER	METEOR
CCG Context (s)	**23.69**	**63.78**	55.88
PB	23.13	64.00	**57.11**
CCG	22.97	64.12	56.15
CCG Context	22.68	64.59	56.62
HPB	22.60	64.32	56.21
CCG (s)	20.98	64.84	55.58

Table 4.2: Experimental results for our CCG-augmented HPB systems and baseline systems on the Arabic–English small news data set. Systems are ordered according to their descending BLEU score.

4.5.4 Small Data Set Experiments

This section presents Arabic-to-English, Chinese-to-English and French-to-English small data set experiments. The Arabic-to-English and Chinese-to-English experiments are conducted on the news and travel speech expressions domains. The systems performance on the UN domain for Arabic-to-English translation is also explored. The language models are trained on the target side of the training data.

Arabic-to-English Experimental Results

Tables 4.2, 4.3 and 4.4 show the BLEU, METEOR and TER scores for our CCG-augmented HPB systems and the baseline systems on Arabic–English news, IWSLT and small UN data sets, respectively. From Table 4.2 we can see that the best-performing system in terms of BLEU and TER scores on the news data is the feature-stripped CCG contextual label system, beating the PB baseline by 0.56 absolute BLEU points, which corresponds to a 2.42% relative improvement. The result of paired bootstrap resmpling test shows that the feature-stripped CCG contextual label system is better than the PB system in 98% of the samples, which is statistically significant at p-level=0.05. However, the PB baseline system outperforms the feature-stripped CCG contextual label system by 1.23 absolute METEOR points, which corresponds to a 2.2% relative improvement. Using CCG contextual labels does not improve performance over CCG category labels. The CCG contextual label system is 0.29 BLEU points behind the CCG category system, which corresponds

System	BLEU	TER	METEOR
HPB	**53.20**	30.95	71.43
CCG Context(s)	53.14	31.04	**71.50**
CCG Context	52.86	31.13	71.33
CCG (s)	52.70	**30.85**	69.85
CCG	52.32	31.89	70.86
PB	52.31	32.42	70.80

Table 4.3: Experimental results for our CCG-augmented HPB systems and the baseline systems on the Arabic–English IWSLT data.

to a 1.2% relative BLEU decrease. However, the CCG contextual label system has a better METEOR score than the CCG category system. The experiments demonstrate that removing features from CCG contextual labels improves the performance of the CCG contextual label system by 1.01 absolute BLEU points, which corresponds to a 4.85% relative improvement, whereas removing features from CCG categories causes a decrease of 1.99 absolute BLEU points, which corresponds to a 9.48% relative performance decrease.

For Arabic-to-English experiments on the IWSLT data, Table 4.3 shows that the HPB system has the best BLEU score with just a small improvement (0.06 absolute BLEU points) over the second best-performing system, namely the feature-stripped CCG contextual label system. We can also see that all CCG-based systems perform better than the PB baseline. The experiments show that the CCG contextual label system performs better than the CCG category system by 0.54 absolute BLEU points, which corresponds to a 1.02% relative improvement. Removing features from CCG contextual labels and CCG categories helps to improve performance by 0.28 and 0.38 absolute BLEU points, which corresponds to relative improvements of 0.58% and 0.72%, respectively. The paired bootstrap resampling test demonstrates that non of these improvements are statistically significant at p-level=0.05.

Table 4.4 shows the BLEU, TER and METEOR scores for our CCG-augmented HPB systems and the baseline systems on the Arabic–English small UN data set. The table demonstrates that the CCG contextual label system is the best-performing system in terms of BLEU score, outperforming the HPB baseline system, which is

System	BLEU	TER	METEOR
CCG Context	**35.54**	50.27	63.96
HPB	35.40	**50.18**	**64.13**
CCG Context (s)	35.16	50.48	63.77
PB	34.79	50.99	64.05
CCG (s)	34.49	50.92	63.34
CCG	32.98	52.93	62.17

Table 4.4: Experimental results for our CCG-augmented HPB systems and the baseline systems on the Arabic–English small UN data set.

the best-performing system in terms of TER and METEOR scores, by 0.14 absolute BLEU points, which corresponds to a 0.4% relative improvement. The paired bootstrap resampling test shows that the CCG contextual label system outperforms the HPB baseline system in 73% of the samples, which is not statistically significant at p-level=0.05. The feature-stripped contextual labels system outperforms the PB baseline system by 0.37 absolute BLEU points, which corresponds to a 1% relative improvement. The paired bootstrap resampling test shows the feature-stripped CCG contextual label system outperforms the PB baseline system in 94% of the samples at p-level=0.05, which is not statistically significant. Removing features from the CCG contextual labels leads to performance degradation by 0.38 absolute BLEU points, which corresponds to a 1% relative decrease. By contrast, removing features from CCG categories improves performance by 0.51 absolute BLEU points, which corresponds to a 1.5% relative improvement. The paired bootstrap resampling test shows that the feature-stripped CCG category system outperforms the CCG category system in 100% of the samples, which is statistically significant at p-level=0.05.

In general, removing features from CCG contextual labels results in performance gain for both news and IWLST data but leads to performance degradation for the UN data. Removing features from CCG categories helps to improve performance for both the IWSLT and UN data but not for the news data.

System	BLEU	TER	METEOR
HPB	**22.26**	**68.31**	51.22
CCG	21.74	68.92	51.02
CCG (s)	21.66	68.75	51.21
CCG Context	21.44	68.71	**51.32**
CCG Context (s)	21.21	69.94	49.61
PB	21.11	68.90	51.02

Table 4.5: Experimental results for our CCG-augmented HPB systems and the baseline systems on the Chinese–English small news data set.

Chinese-to-English Experimental Results

Tables 4.5 and 4.6 show the BLEU, METEOR and TER scores for our CCG-augmented HPB systems and the baseline systems on the Chinese–English news and IWSLT data, respectively. For the news data, Table 4.5 shows that the HPB baseline system is the best-performing systems in terms of BLEU and TER, outperforming the CCG category system by 0.52 absolute BLEU points, which corresponds to a 0.24% relative improvement. Removing features from CCG category and CCG contextual labels decreases performance by 0.08 and 0.23 absolute BLEU points, which corresponds to relative decreases of 0.37% and 1%, respectively. It is worth noting that the CCG contextual label system achieves the best METEOR score among the systems. Although the CCG contextual label system achieves a lower BLEU score than the CCG category system, the CCG contextual label system outperforms the CCG category system in both TER and METEOR scores by 0.21 absolute TER points and 0.30 absolute METEOR points, which corresponds to 0.3% relative TER improvement and 0.6% relative METEOR improvement.

For the IWSLT data, Table 4.6 shows that the CCG contextual label system is the best-performing system in terms of BLEU and METEOR scores, beating the HPB baseline system by 0.53 absolute BLEU points, which corresponds to a 1% relative improvement. However, the paired bootstrap resampling test shows that this improvement is not statistically significant at p-level=0.05. Using CCG contextual labels improves performance over the CCG category system by 3.16 absolute

BLEU points, which corresponds to a 6.5% relative improvement. The paired bootstrap resampling test shows that the CCG contextual label system outperforms the CCG category system in 98% of the samples, which is statistically significant at p-level=0.05. We can see also from Table 4.6 that removing features from CCG contextual labels causes a performance decrease of 0.6 absolute BLEU points, which corresponds to a 1.17% relative performance decrease. It is worth noting that removing features from CCG contextual labels achieves a better TER score than the CCG contextual label system by 0.45 absolute TER points, which corresponds to a 1.63% relative improvement. In contrast, removing features from CCG categories improves performance by 0.6 absolute BLEU points, which corresponds to a 1.24% relative improvement. The paired bootstrap resampling test shows that the feature-stripped CCG category system outperforms the CCG category system in 71% of the samples, which is not statistically significant at p-level=0.05.

System	**BLEU**	**TER**	**METEOR**
CCG Context	**51.42**	33.00	**67.82**
HPB	50.89	**32.53**	66.81
CCG Context (s)	50.82	32.55	67.59
CCG (s)	48.86	36.12	65.92
CCG	48.26	36.17	64.88
PB	47.69	34.20	65.35

Table 4.6: Experimental results for our CCG-augmented HPB systems and the baseline systems on the Chinese–English IWSLT data.

In general, the Chinese-to-English experimental results for the IWSLT and small news data sets show that removing features from CCG contextual labels damages performance on both the news and IWSLT data, whereas removing features from CCG categories does not show a consistent effect on the performance of the CCG category system.

Compared to the Arabic-to-English IWSLT experimental results, the Chinese-to-English IWSLT experiments show that the CCG contextual label system outperforms the HPB baseline while none of the Arabic-to-English CCG-based systems is able to beat the HPB baseline on the IWSLT data. Bearing in mind that the size

System	BLEU	TER	METEOR
HPB	**38.96**	**40.07**	**60.05**
CCG Context	38.40	40.91	59.22
PB	38.37	39.92	59.48
CCG Context (s)	38.36	41.10	59.26
CCG (s)	38.12	40.36	59.38
CCG	36.12	42.17	57.98

Table 4.7: Experimental results for our CCG-augmented HPB systems and the baseline systems on 20k of the Chinese–English IWSLT data.

of the Chinese–English IWSLT training data is about three times the size of the Arabic–English IWSLT data, and in order to verify whether this result is related to the source language or the data size, we reduced the size of the Chinese–English IWSLT data to the size of the Arabic–English IWSLT training data and reran the same systems. Table 4.7 presents the results of the Chinese-to-English systems on the reduced data set. We see that the HPB baseline system has the best BLEU score, while other CCG-based systems maintain the same order relative to each other. Reducing data size results in an advantage for the HPB system over other CCG-based systems. This indicates that the size of the training data is an important factor affecting the performance of the CCG-based systems.

French-to-English Experimental Results

Table 4.8 shows the BLEU, METEOR and TER scores of our CCG-augmented HPB systems and the baseline systems on the French–English small data set. The table shows that the PB baseline system is the best-performing system in terms of BLEU and METEOR scores, outperforming the second best-performing system, namely the CCG category system, by 0.14 absolute BLEU points, which corresponds to a 0.5% relative improvement. The paired bootstrap resampling test shows that the PB baseline system outperforms the CCG category system in 73% of the samples, which is not statistically significant at p-level=0.05. The CCG category system outperforms the HPB baseline system by a small margin of just 0.01 absolute BLEU points. The table also shows that removing features from CCG categories decreases

System	BLEU	TER	METEOR
PB	**27.84**	56.47	**58.98**
CCG	27.70	56.56	58.84
HPB	27.69	56.60	58.65
CCG (s)	27.58	56.47	58.84
CCG Context (s)	27.57	**56.41**	58.87
CCG Context	27.44	56.59	58.84

Table 4.8: Experimental results for our CCG-augmented HPB systems and the baseline systems on the French–English small data set.

performance by 0.12 absolute BLEU points, which corresponds to a 0.43% relative decrease. However, removing features from CCG contextual labels leads to an increase of 0.13 absolute BLEU points, which corresponds to a 0.47% relative increase. However, this improvement is not statistically significant at p-level=0.05 according to the bootstrap resampling test. It is worth noting that the feature-stripped CCG contextual label system is the best-performing system in terms of TER score. In addition, we can see that the CCG category systems achieve better performance than the CCG contextual label systems.

4.5.5 Large Data Set Experiments

In this section, we present experiments which examine the effect of adding more training data on the performance of our CCG-augmented HPB systems. The Arabic–English, Chinese–English and French–English training data consists of 223k sentence pairs each, which is about four times the size of the training data used in the small data set experiments (cf. Section 4.5.1). The language models used by the systems in this section are built from the target side of the training data.

Arabic-to-English Experimental Results

Table 4.9 shows the experimental results of our CCG-augmented HPB systems and the baseline systems on the Arabic–English large data set. We can see that the HPB and PB baseline systems are the best-performing systems. The HPB baseline system is the best-performing system in terms of BLEU score, whereas the

System	BLEU	TER	METEOR
HPB	**43.35**	43.94	69.58
PB	43.34	**43.45**	**70.19**
CCG Context	43.06	44.04	69.45
CCG Context (s)	42.94	44.07	69.69
CCG	42.13	45.05	68.75
CCG (s)	41.62	44.83	68.84

Table 4.9: Experimental results for our CCG-augmented HPB systems and the baseline systems on the Arabic–English large data set.

PB baseline system is the best-performing system in terms of TER and METEOR scores. We can also see that the CCG contextual label systems outperform the CCG category systems for all metrics. The table also shows that removing features from CCG contextual labels and from CCG categories demonstrably decreases the performance of the corresponding systems by 0.12 and 0.51 absolute BLEU points, which corresponds to relative decreases of 0.28% and 1.21%, respectively.

Chinese-to-English Experimental Results

Table 4.10 shows the experimental results of our CCG-augmented HPB systems and the baseline systems on the Chinese–English large data set. The table demonstrates that the feature-stripped CCG contextual label system is the best-performing system in terms of BLEU score, outperforming the HPB baseline system by 0.17 absolute BLEU points, which corresponds to a 0.68% relative improvement. The paired bootstrap resampling test shows that the CCG contextual label system outperforms the HPB baseline system in 83% of the samples, which is not statistically significant at p-level=0.05. Using contextual labels causes a slight performance decrease of 0.03 absolute BLEU points compared with the CCG category system. We can see that removing features from CCG contextual labels and CCG categories results in increases of 0.8 and 0.47 absolute BLEU points, which corresponds to relative improvements of 3.2% and 1.92%, respectively. The paired bootstrap resampling test shows that these improvements are statistically significant at p-level=0.05.

System	BLEU	TER	METEOR
CCG Context (s)	**25.24**	64.76	54.11
HPB	25.07	**64.45**	54.21
CCG (s)	24.94	64.77	**54.55**
CCG	24.47	65.11	54.31
CCG Context	24.44	65.5	53.21
PB	24.39	65.56	54.10

Table 4.10: Experimental results for our CCG-augmented HPB systems and the baseline systems on the Chinese–English large data set.

French-to-English Experimental Results

Table 4.11 shows experimental results for our CCG-augmented HPB systems and the baseline systems on the French–English large data set. The table shows that the feature-stripped CCG category system achieves the best performance among the systems according to BLEU, TER and METEOR scores, outperforming the second best-performing system, namely the CCG category system, by 0.27 absolute BLEU points, which corresponds to a 1% relative improvement. The paired bootstrap resampling test shows that the feature-stripped CCG category system outperforms the CCG category system in 100% of the samples, which is statistically significant at p-level=0.05. The table also shows that the CCG category system outperforms the HPB baseline system by 0.08 absolute BLEU points, which corresponds to a 0.27% relative improvement. The paired bootstrap resampling test shows that the CCG category system outperforms the HPB baseline system in 100% of the samples, which is statistically significant at p-level=0.05. The CCG contextual label systems perform similarly, with a small difference of 0.01 absolute BLEU points between the two systems, and they both underperform compared to the CCG category system and the HPB baseline system in terms of BLEU, TER and METEOR.

4.5.6 Large Language Model Experiments

In this section, we present experiments conducted on the Arabic–English, Chinese–English and French–English small data sets using a large language model. The

System	BLEU	TER	METEOR
CCG (s)	**29.93**	**54.43**	**60.93**
CCG	29.76	54.68	60.89
HPB	29.68	54.69	60.84
CCG Context (s)	29.66	54.85	60.77
CCG Context	29.65	55.26	60.52
PB	29.31	55.13	60.45

Table 4.11: Experimental results for our CCG-augmented HPB systems and the baseline systems on the French–English large data set.

training data sets we use in these experiments are the same as those used in the small data sets experiments (cf. Section 4.5.4) but the language models used by the systems in this section are built from the target side of the large data sets (cf. Section 4.5.1).

Arabic-to-English Experimental Results

Table 4.12 shows the experimental results for our CCG-augmented HPB systems and the baseline systems on the Arabic–English small data set using a large language model. The table shows that the HPB baseline system is the best-performing system in terms of BLEU, TER and METEOR scores. We can also see that all the CCG-augmented HPB systems underperformed compared to the PB and HPB baseline systems in terms of BLEU score. The table also shows that the CCG contextual label system outperforms the CCG category system by 1.17 absolute BLEU points, which corresponds to a 3.12% relative improvement. The paired bootstrap resampling test shows that this improvement is statistically significant at p-level=0.05. We can also see that removing features from CCG contextual labels causes a performance decrease of 1.59 absolute BLEU points, which corresponds to a 4% relative decrease. By contrast, removing features from CCG categories leads to a slight increase of 0.09 absolute BLEU points, which corresponds to a 0.24% relative improvement. The paired bootstrap resampling test shows that this improvement is not statistically significant at p-level=0.05.

System	BLEU	TER	METEOR
HPB	**39.36**	**47.92**	**66.40**
PB	38.79	48.44	66.19
CCG Context	38.66	48.29	65.76
CCG (s)	37.58	49.69	64.78
CCG	37.49	49.42	64.97
CCG Context (s)	37.07	50.39	64.49

Table 4.12: Experimental results for our CCG-augmented HPB systems and the baseline systems on the Arabic–English small data set using a large language model.

Chinese-to-English Experimental Results

Table 4.13 demonstrates the experimental results for our CCG-augmented HPB systems and the baseline systems on the Chinese–English small data set using a large language model. From the table we can see that the HPB baseline system is the best-performing system in terms of BLEU, METEOR and TER scores, outperforming the second best-performing system, namely the CCG contextual label system by 0.6 absolute BLEU points, which corresponds to a 2.6% relative improvement. The paired bootstrap resampling test shows that the HPB baseline system outperforms the CCG contextual label system in 100% of the samples, which is statistically significant at p-level=0.05. We can also see that the CCG category system comes behind all the other CCG-augmented HPB systems, whereas it outperforms the PB baseline system by 0.29 absolute BLEU points, which corresponds to a 1.31% relative improvement. The table demonstrates that removing features from CCG categories improves performance by 0.39 absolute BLEU points, which corresponds to a 1.75% relative improvement. The paired bootstrap resampling test shows that the feature-stripped CCG category system outperforms the CCG contextual label system in 66% of the samples, which is not statistically significant at p-level=0.05. On the other hand, removing features from CCG contextual labels decreases performance by 1.01 absolute BLEU points, which corresponds to a 4.36% relative decrease.

System	BLEU	TER	METEOR
HPB	**23.77**	**66.37**	**52.79**
CCG Context	23.17	66.48	52.31
CCG (s)	22.70	66.84	52.28
CCG Context (s)	22.69	67.69	51.42
CCG	22.31	67.71	51.85
PB	22.02	68.57	51.17

Table 4.13: Experimental results for our CCG-augmented HPB systems and the baseline systems on the Chinese–English small data set using a large language model.

French-to-English Experimental Results

Table 4.14 demonstrates the experimental results for our CCG-augmented HPB systems and the baseline systems on the French–English small data set using a large language model. The table shows that the feature-stripped CCG category system is the best-performing system in terms of BLEU, METEOR and TER scores, outperforming the HPB baseline system by 0.26 absolute BLEU points, which corresponds to a 0.92% relative improvement. The paired bootstrap resampling test shows that the feature-stripped CCG category system outperforms the HPB baseline system in 87% of the samples, which is not statistically significant at p-level=0.05. We can also see that the CCG contextual label system outperforms the CCG category system by 0.57 absolute BLEU points, which corresponds to a 2% relative improvement. The table demonstrates that the feature-stripped CCG contextual label system lies behind the CCG contextual label system by 0.66 absolute BLEU points, which corresponds to a 2.4% relative decrease. In contrast, removing features from CCG categories improves performance by 1.04 absolute BLUE points, which corresponds to a 3.78% relative improvement.

4.5.7 Analysis

In this section, we provide an analysis of the experimental results obtained in Sections 4.5.4–4.5.6. We explore the sparsity of the simplified CCG labels. We also investigate the effect of the different factors on the performance of our CCG-augmented

System	BLEU	TER	METEOR
CCG (s)	**28.52**	**55.35**	**59.49**
HPB	28.26	55.66	59.35
PB	28.16	56.35	59.30
CCG Context	28.05	56.03	59.22
CCG	27.48	57.00	59.16
CCG Context (s)	27.39	57.02	58.95

Table 4.14: Experimental results for our CCG-augmented HPB systems and the baseline systems on the French–English small data set using a large language model.

HPB systems, namely the domain of the data, the size of the training data and the size of the language model.

System	Number of Labels
CCG	547
CCG (s)	221
CCG Context	251
CCG Context (s)	116

Table 4.15: Number of different labels used by our CCG-augmented HPB systems to annotate the target side of the Arabic–English small UN data set.

Label Sparsity

Without loss of generality, we study the sparsity of our simplified CCG labels on the Arabic–English UN small data set only, as we always use CCG labels on the English side, and due to the close similarities in the distribution of CCG category label frequency counts between Arabic–English and Chinese–English data sets illustrated in Figures 3.10 and 3.11 (pages 72 and 73). We first measure the number of different syntactic labels used by each CCG-augmented system to annotate the target side of the Arabic–English small UN data set as illustrated in Table 4.15. We can see that the feature-stripped CCG category system and the CCG contextual label system have similar number of labels, which constitutes less than a half of the number of labels used by the CCG category system. The feature-stripped CCG contextual label system has the least number of labels, which constitutes about 21% of the labels used by the CCG category system.

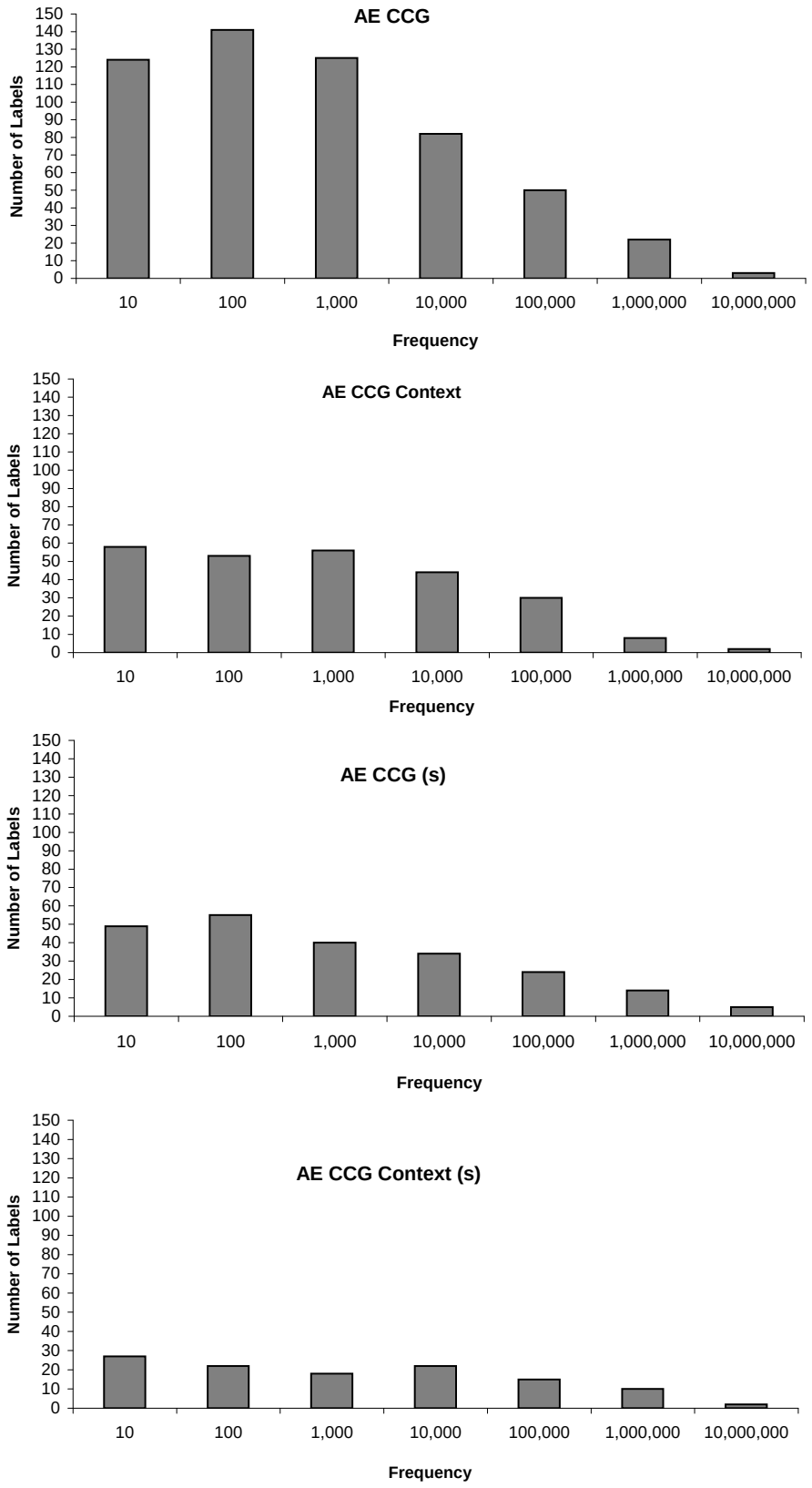

Figure 4.4: Label frequency counts for our CCG-augmented HPB systems built on the Arabic–English small UN data set.

Similar to our approach to illustrate label sparsity in Figures 3.10 and 3.11, Figure 4.4 demonstrates the distribution of label frequency counts for labels used by our CCG-augmented HPB systems to annotate the target side of the Arabic–English small UN data set. We can see that the simplified CCG labels have in general more label mass shifted towards higher frequencies compared with the CCG category labels, which indicates reduced label sparsity. The feature-stripped CCG contextual labels seem to have the least sparsity among other labels with no more than 42% of the labels occur less than or equal to 100 times in the training data. The feature-stripped CCG category labels are slightly less sparse (47% of the labels with frequencies less than or equal to 100) compared with the CCG category labels (48% of the labels with frequencies less than or equal to 100). The CCG contextual labels lie in the middle with 44% of the labels occur less than or equal to 100 times.

Arabic-to-English Experiments

Figure 4.5 shows the BLEU scores of our CCG-augmented HPB systems and baseline systems on each of the Arabic–English small UN data set, the large UN data set and the small UN data set with a large language model. We can see that removing features from CCG categories improves the BLEU score under both the small data set setting and the large language model setting. By contrast, removing features from CCG categories decreases the BLEU score under the large data setting. Furthermore, the figure demonstrates that the CCG contextual label system is the best-performing CCG-augmented system under all settings, outperforming the PB and HPB baseline systems on the small data set only. Simplifying CCG labels helps to improve the performance over the HPB and PB baseline systems on the small data set only, whereas it failed to improve performance over the baseline systems when using a larger language model or on the large data set. This might be because using a larger language model and larger data helps to improve the grammaticality of the translation, which minimizes the role of the syntactic labels in improving the grammaticality of the translation. The figure also shows that removing features

from CCG contextual labels causes performance to degrade under all settings.

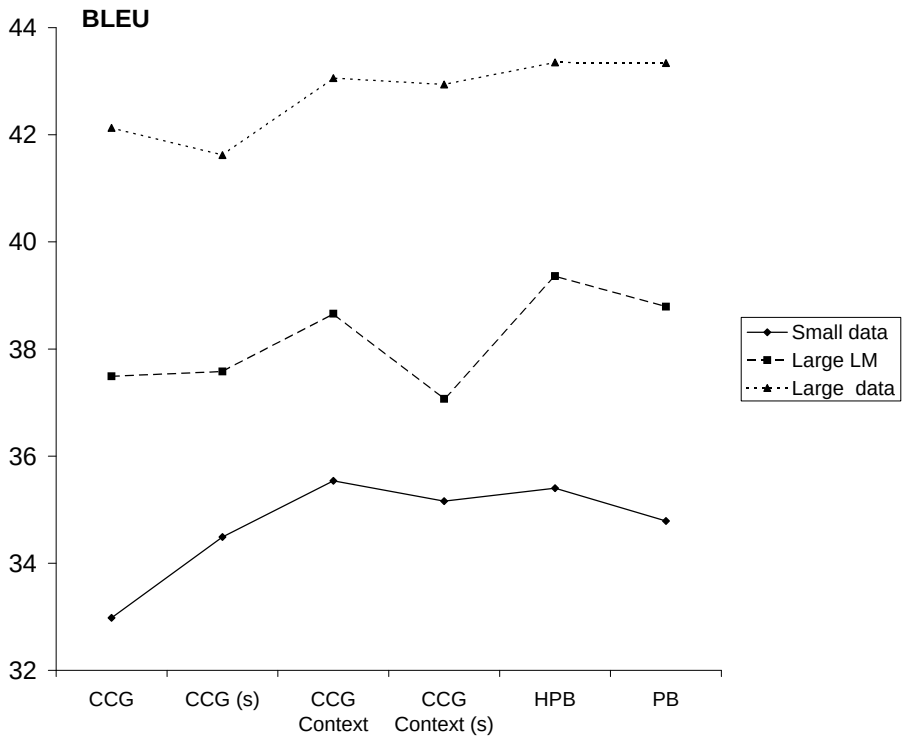

Figure 4.5: BLEU scores of our CCG-augmented HPB systems and baseline systems for Arabic-to-English translation on each of the small UN data set, the UN large data set and the small UN data set with a large language model.

Figure 4.6 shows the BLEU scores of our CCG-augmented HPB systems and the baseline systems on the small data sets from different domains: UN, news and IWSLT. The figure demonstrates that the CCG-augmented HPB systems show similar trends on the UN and IWSLT data. Removing features from CCG categories improves the BLEU score for both data sets. In addition, using CCG contextual labels achieves additional performance gains over the feature-stripped CCG categories labels for both data sets. Removing features from CCG contextual labels improves performance on the IWSLT data to the best BLEU score among the CCG-augmented HPB systems, whereas it damages performance on the UN data. The figure also demonstrates that the CCG-augmented HPB systems show different trends on the news data from these found on the UN and IWSLT data. We can see that removing features from CCG categories damages performance to a large extent on the news

data. Furthermore, the CCG contextual label system is not able to outperform the CCG category system on the news data. The figure also demonstrates that removing features from CCG contextual labels leads to the best performance among the CCG-augmented HPB systems and the baseline systems on the news data.

In general CCG contextual label systems perform better than CCG category systems for Arabic-to-English translation. This might be explained by some consistency exhibited when translating Arabic phrases into English phrases regarding the type of the resulting category in the CCG category of the target English phrase. Therefore, this consistency allows the resulting category to be removed from the CCG label representation without much affecting the grammaticality of the translation output while achieving a performance gain from using less sparse labels.

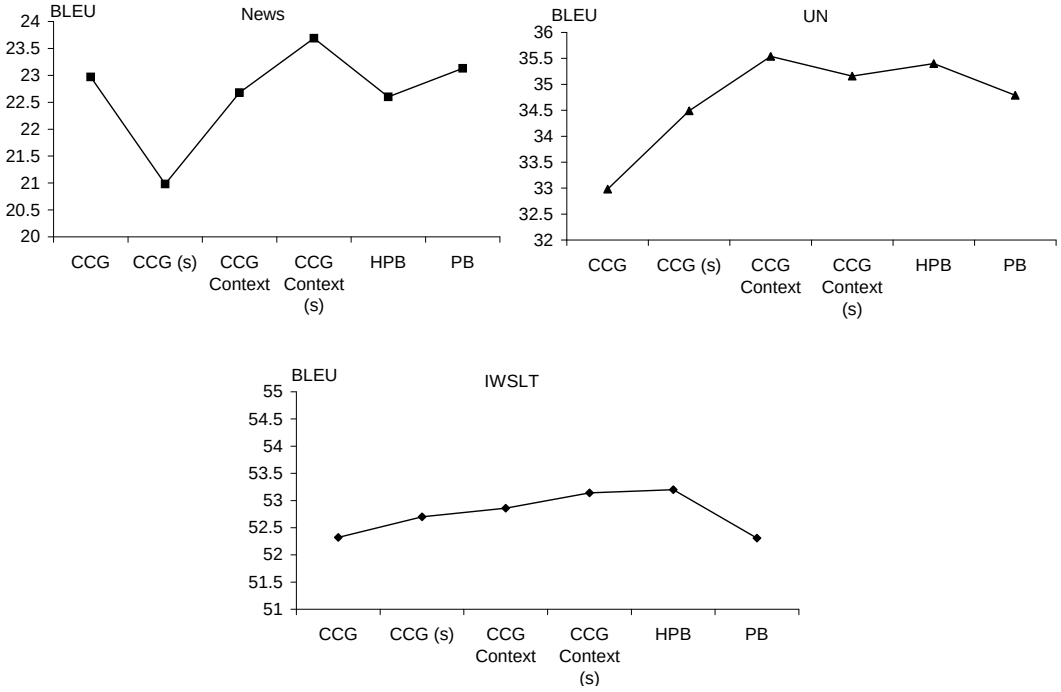

Figure 4.6: BLEU scores of our CCG-augmented HPB systems and the baseline systems for Arabic-to-English translation on the news, UN and IWSLT data sets.

Chinese-to-English Experiments

Figure 4.7 shows the performance of our CCG-augmented HPB systems and the baseline systems measured in terms of BLEU score for Chinese-to-English translation

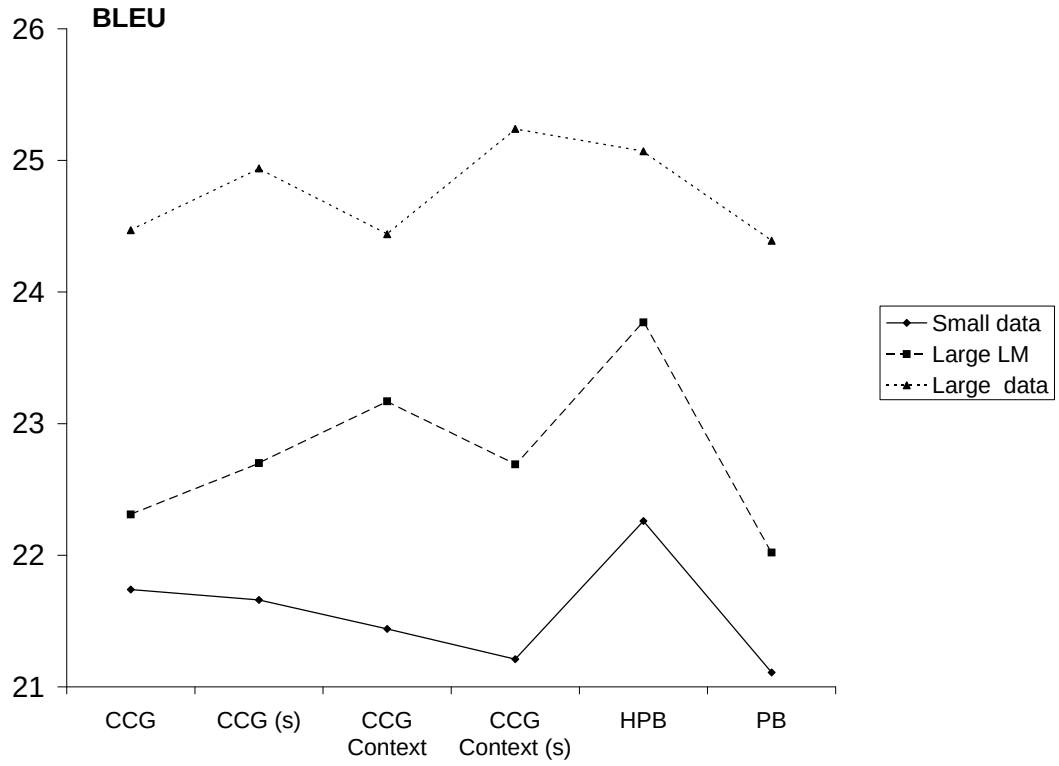

Figure 4.7: BLEU scores of our CCG-augmented HPB systems and the baseline systems for Chinese-to-English news translation on each of the small data set, the large data set and the small data set with a large language model.

on each of the small data set, the large data set and the small data set with a large language model in the news domain. We can see that the CCG-augmented HPB systems show reverse trends between the small data set setting and the large language model setting. Removing features from CCG categories leads performance to degrade under the small data set setting, whereas it improves performance under the large language model setting. Furthermore, the CCG contextual label system achieves a further increase in BLEU score over the feature-stripped CCG category system under the large language model setting. By contrast, the CCG contextual label system has a lower BLEU score than the feature-stripped CCG category system under the small data set setting. The figure also shows that for both the large language model setting and the small data set setting, removing features from CCG contextual labels decreases performance. Moreover, for both the large language model setting and the small data set setting, none of the CCG-augmented HPB

systems is able to outperform the HPB baseline system, whereas for the large data set setting, the feature-stripped CCG contextual label system, which is the best-performing system, outperforms the HPB baseline system. For the large data set setting, we can see that the systems show a different trend compared with the small data setting and the large language model setting. The figure shows that removing features from CCG categories helps to improve performance on the large data set. Furthermore, the CCG contextual label system has the worst BLEU score among the CCG-augmented HPB systems. However, removing features from CCG contextual labels helps to improve performance over the other CCG-augmented HPB systems and the baseline systems.

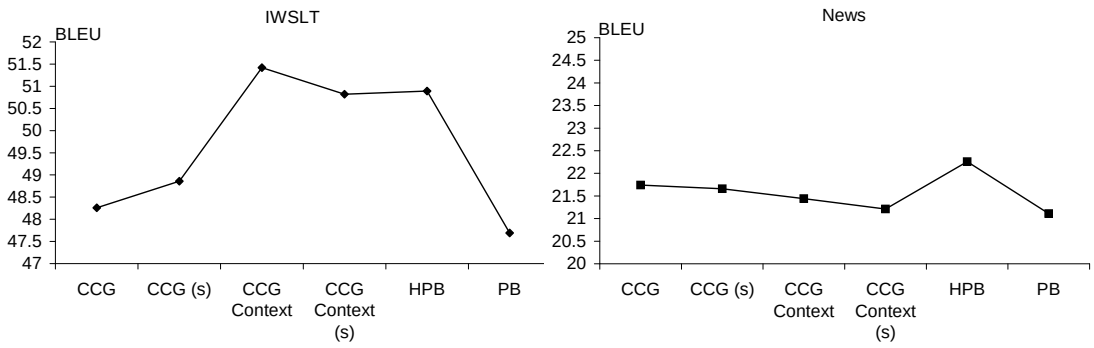

Figure 4.8: BLEU scores of our CCG-augmented HPB systems and the baseline systems for Chinese-to-English translation on the IWSLT data set and the news small data set.

Figure 4.8 shows the BLEU scores of our CCG-augmented HPB systems and the baseline systems for Chinese-to-English translation on data from two different domains: IWSLT and news. We can see the reverse trend demonstrated by the CCG-augmented HPB systems between the different domains. While performance increases when moving from the CCG category system, passing through the feature-stripped CCG category system to the CCG contextual label system on the IWSLT data, performance shows a downward trend for the same systems on the news data. The figure also shows that removing features from the CCG contextual label system causes performance to degrade on both the IWSLT and news data sets. The

CCG contextual label system achieves the best performance over the other CCG-augmented HPB systems and the baseline systems on the IWSLT data. In contrast, the CCG category system is the best-performing CCG-augmented HPB system on the news data but it is not able to outperform the HPB baseline system.

French-to-English Experiments

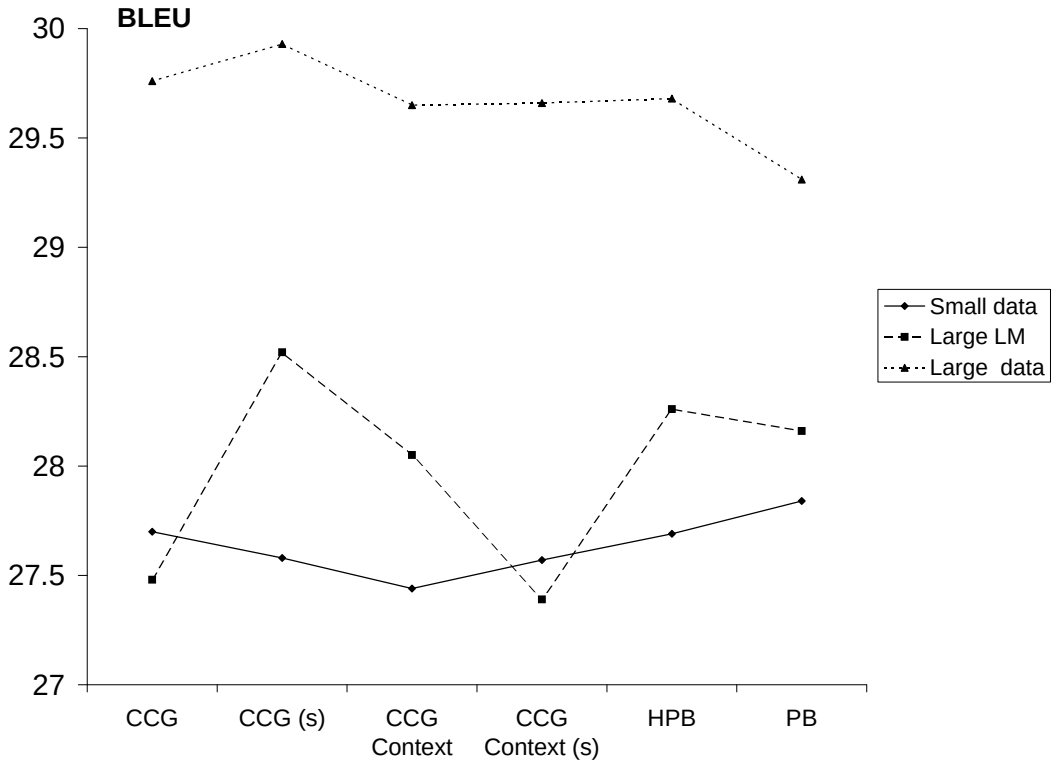

Figure 4.9: BLEU scores of our CCG-augmented HPB systems and the baseline systems for French-to-English translation on each of the small data set, the large data set and the small data set with a large language model.

Figure 4.9 shows the performance of our CCG-augmented HPB systems and baseline systems measured in terms of BLEU score for French-to-English translation on each of the small data set, the large data set and the small data set with a large language model. The figure demonstrates that removing features from CCG categories improves performance under the large data set and the large language model settings, whereas it decreases performance on the small data set. In addition, we can see that removing features from CCG contextual labels improves performance

on the small data set, whereas it causes performance to largely degrade under the large language model setting, and it causes a marginal increase on the large data set. The CCG contextual label system is able to outperform the CCG category system under the large language model setting only. The figure also demonstrates that the feature-stripped CCG category system is able to outperform the HPB and PB baseline systems under the large data and the large language model settings, achieving the best performance among all the systems under both settings. By contrast, the HPB baseline system and the CCG category system perform almost similarly on the small data set, outperforming the other CCG-augmented HPB systems. It is worth noting that the CCG category system and the feature-stripped CCG contextual label system achieve lower BLEU scores under the large language model setting than the BLEU scores they achieve on the small data set, which might happen when using a large out-of-domain language model because of the differences in the learnt weights of the language model feature between different systems during tuning.

In general, the feature-stripped CCG label system performs better than other CCG-augmented HPB systems for French-to-English translation. Similar to the Arabic-to-English experimental analysis, we might explain this by consistencies exhibited when translating French phrases into English in terms of the features held by CCG categories of the English phrases. This helps to guarantee the use of the English phrase with the correct CCG feature during translation without explicitly representing this feature in the label of the English phrase. This provides the ability to take advantage of using less sparse labels by stripping features from CCG categories while maintaining the same grammaticality level obtained using the more fine-grained CCG category labels which bear these syntactic features.

Conclusion

The detailed analysis of the experimental results obtained under different factors demonstrates that the performance of the different CCG-based labelling approaches is affected by the language pair, the data size, the language model size and the

System	Data	AE	CE	FE
CCG (s)	IWSLT	+	+	
	Small news	−		
	Small data	+	−	−
	Large LM	+	+	+*
	Large data	−	+	+*
CCG Context	IWSLT	+	+*	
	Small news	−*		
	Small data	+*	−	−
	Large LM	+	+	+
	Large data	+	−	−
CCG Context (s)	IWSLT	+	+	
	Small news	+*		
	Small data	+	−	−
	Large LM	−	+	−
	Large data	+	+*	−

Table 4.16: A summary of performance improvements (+) and degradations (–) obtained using our label simplification approaches under different factors compared with the CCG-augmented HPB system which uses CCG categories as nonterminal labels. The * symbol indicates that the system outperforms the HPB baseline system.

data domain. For Arabic-to-English translation on different domains, the feature-stripped CCG contextual label system was demonstrated to achieve the best performance among the other CCG-augmented HPB systems on the IWSLT and news data, outperforming the baseline systems on the news data only. For Arabic-to-English UN data translation under different settings, the CCG contextual label system was the best-performing CCG-augmented HPB system under the small data, the large data and the large language model settings, outperforming the baseline systems on the small data set. In general, CCG contextual label systems outperformed the CCG category systems for Arabic-to-English translation under different settings and in different domains. For Chinese-to-English news translation, the best-performing CCG-augmented HPB system varied under different settings. For the small news data set, no CCG label simplification approach helped to improve performance over the CCG category system. For the large language model setting, the CCG contextual label system was the best-performing CCG-augmented HPB sys-

tem, but it could not outperform the HPB baseline system either. For the large data set, the feature-stripped CCG contextual label system is the best-performing CCG-augmented HPB system and it outperforms the baseline systems. For Chinese-to-English translation on different domains, the CCG contextual label system achieved the best performance among all the examined systems on the IWSLT data, whereas the CCG category system was the best-performing CCG-augmented HPB system for the news domain. Generally, for Chinese-to-English translation, there is no clear trend whether the CCG contextual label systems perform better than CCG category systems. For French-to-English translation under different settings, the feature-stripped CCG category system was the best-performing system under the large language model and the large data set settings, outperforming the HPB and PB baseline systems under both settings, whereas no CCG label simplification approach helped to improve the performance over the CCG category system on the small data set. For French-to-English translation under different settings, the CCG category systems achieved better performance than the CCG contextual label systems in general. Figure 4.16 demonstrates a summary of performance gains and losses achieved by our label simplification approaches under different factors compared with the performance of the CCG-augmented HPB system which uses CCG categories as nonterminal labels.

4.6 Conclusions

In Chapter 3, we demonstrated that the HPB system with nonterminals augmented with CCG categories performed better than the SAMT system. However, it was not able to outperform the HPB baseline system in many experiments. We believe that this was due to the sparsity of CCG labels and the strict syntactic constraints imposed by them. In this chapter we tried to solve this problem by extracting less sparse CCG-based labels while at the same time maintaining rich syntactic information in the labels. We introduced two approaches to simplifying CCG-based nonter-

minal labels. The first approach uses contextual information extracted from CCG categories to label nonterminals in hierarchical rules. The second approach simplifies CCG categories by removing syntactic features held by some CCG categories. We also examined a third simplification approach which combines the previous approaches together.

We presented experiments which examine the effectiveness of our CCG-based label simplification approaches under different factors: the language pair, the corpus size, the language model size and the domain of the data. We conducted experiments on Arabic-to-English, Chinese-to-English and French-to-English translation using different training data and language model sizes and domains. Our experimental results demonstrated that our CCG-based label simplification approaches are promising. Our CCG-augmented HPB systems which use simplified CCG labels were able to outperform the CCG-augmented HPB system which uses CCG categories in most of the conducted experiments. Furthermore, CCG-based label simplification helped to improve performance over the HPB and PB baseline systems in some of the experiments. Our experimental analysis demonstrated that the performance of the different simplification approaches varied according to the language pair, the corpus size, the size of the language model used and the domain of the data.

To conclude, we demonstrated in this chapter that simplifying CCG category-based nonterminal labels helped to improve the performance of our CCG-augmented HPB system, which addresses our second research question (**RQ2**). In the next chapter, we will try to further extend the CCG-augmentation for the HPB SMT model by incorporating CCG combinatory rules in the HPB decoding process and increasing the coverage of the CCG-based syntactic constraints.

Chapter 5

Extending CCG-based Syntactic Constraints in Hierarchical Phrase-Based SMT

In the previous chapter, we tried to improve the performance of our CCG category HPB system by extracting less sparse CCG-based labels. In this chapter, we explore the factors which limit the coverage of the syntactic constraints applied in our CCG-augmented HPB system, which addresses our third research question (**RQ3**). Furthermore, we propose two approaches to extend the coverage of the syntactic constraints applied in our CCG-augmented HPB system and explore their effect on performance, which addresses our fourth research question (**RQ4**). Our first approach to extending the coverage of the syntactic constraints in our CCG-augmented HPB model tries to increase the coverage of CCG-based nonterminal labels. Our second extension approach extends the CCG-based syntactic constraints to include the whole HPB SMT grammar.

Section 5.1 discuses the limitations on the coverage of the syntactic constraints applied in our CCG-augmented HPB system and presents our motivation to extend these syntactic constraints. Section 5.2 reviews related work and provides an introduction to the Preference Grammars paradigm to apply soft syntactic constraints

in the HPB SMT model, according to which we apply soft syntactic constraints in our models. Section 5.3 introduces our first approach to extending the syntactic constraints in our CCG-augmented HPB system, namely the extended CCG labels. Section 5.4 presents our second approach to extending the syntactic constraints in our CCG-augmented HPB system, namely the CCG-augmented glue grammar. In Section 5.5, we present our experiments which explore the effect on performance of our extension approaches both individually and combined on data from different domains and of different sentence lengths. Finally, we provide our conclusions in Section 5.6.

5.1 Motivation

Providing SMT systems with target-side syntactic knowledge aims at guiding the translation process towards producing more grammatical translations. Theoretically, this should help to improve the performance of the syntax-augmented SMT systems over their syntax-free counterparts. However, syntax augmentation for SMT faces many hurdles which minimize the benefit gained from applying syntactic constraints by SMT systems. One of the main problems facing syntax-augmented SMT systems is the limited coverage of the syntactic constraints applied in them. This problem emerges from the inability of different grammar theories to express the syntactic constraints imposed on phrases and rules extracted by SMT systems via purely statistical methods without using any syntactic knowledge. Another problem affecting the performance of syntax-augmented SMT systems is the strictness of the syntactic constraints applied in them. Hard syntactic constraints restrict the decoding process by excluding all the translations which violate them. Some of these excluded translations are in fact grammatical translations, but they do not comply with the learnt syntactic constraints either because of parsing errors committed during syntactic annotation of the data, or because of the sparsity of the syntax-augmented translation model.

In order to examine the coverage of the syntactic constraints applied by our CCG-augmented HPB SMT systems, we look back to Table 3.8 (page 74) which shows the percentage of unlabelled phrases extracted by each of the SAMT and CCG-augmented HPB systems built in Section 3.5 (page 60). The table shows that the SAMT system fails to label 44% to 70% of the total phrase pairs whereas the CCG-augmented HPB system fails to label 30% to 46% of the total phrase pairs extracted from the Chinese–English and Arabic–English IWSLT and news training corpora. This indicates that at least third of the extracted phrases in our CCG-augmented HPB systems do not undergo any syntax control.

Another factor limiting the coverage of the syntactic constraints in syntax-augmented HPB SMT systems is that only part of the grammar, namely hierarchical rules, is syntactically augmented. Previous syntax augmentation approaches for HPB SMT (Zollmann and Venugopal, 2006; Venugopal et al., 2009; Chiang, 2010; Stein et al., 2010) ignore the other part of the grammar, namely glue grammar (cf. Section 2.4.1 page 21). We examine the importance of glue grammar in the HPB SMT model by measuring the percentage of glue grammar rules out of the total rules which participated in the derivations produced by each of the SAMT and CCG-augmented HPB systems on the Chinese–English and Arabic–English news and IWSLT data (cf. Section 3.5 page 60). Table 5.1 shows that glue grammar rules constitute about 30% to 57% of the total rules, which means that they play an important role in the translation process. Bearing in mind that the application of hierarchical rules has usually a limited span in order to reduce the complexity of chart decoding, the application of the syntactic constraints imposed by them is also limited for the same reason.

After identifying the factors which limit the coverage of the syntactic constraints in our CCG-augmented HPB system, which addresses our third research question (**RQ3**), we try to tackle these factors by expanding the scope of the CCG-based syntactic constraints in our CCG-augmented HPB system. To achieve this we follow a two-fold approach. First, we try to extend the notion of the syntactic label

System	CE news	AE news	CE IWSLT	AE IWSLT
CCG	29.87%	35.63%	56.10%	56.90%
SAMT	30.47%	34.35%	53.47%	50.26%

Table 5.1: Percentage of glue grammar rules out of the total rules participating in the derivations produced by each of the SAMT and CCG-augmented HPB systems built on the Chinese–English and Arabic–English news and IWSLT data (cf. Section 3.5 page 60).

attached to nonterminals and phrases with the aim of increasing label coverage (cf. Section 5.3). Secondly, we provide glue grammar with syntax by augmenting glue grammar rules with CCG combinatory rules (cf. Section 5.4). We also examine the combination of these two approaches. In order to avoid losing the performance gain that might be achieved by our enhancements because of the strictness of the syntactic constraints, we try to apply these enhancements in a soft way under the Preference Grammars paradigm for applying soft syntactic constraints in HPB SMT (Venugopal et al., 2009) (cf. Section 5.2.1).

5.2 Related Work

Several approaches have been proposed to increase the coverage of the syntactic constraints in syntax-augmented SMT systems. SAMT (Zollmann and Venugopal, 2006) increases the coverage of CF-PSG constituent labels using CCG-like slash operators in addition to the concatenation "plus" operator to combine constituent labels extracted from CF-PSG parse trees. Wang et al. (2010) try to solve the flat structures problem of the Penn Treebank-style trees, which decreases the coverage and reduces the generalization ability of the syntax-augmented translation rules. They explore the effect of various tree restructuring approaches. They try simple binarization approaches such as left, right and head binarization in addition to learning restructuring preferences (left or right) for each node in the tree using Expectation Maximization (Dempster et al., 1977).

In order to increase nonterminal label coverage but at the same time avoid the

proliferation of syntactic labels when extending them with binary operators as in SAMT, Stein et al. (2010) use only the Penn Treebank tag set to label nonterminals and phrases in the HPB SMT model. Phrases which are not spanned by a single node in the parse tree are labelled with the label of the closest node. The distance to a certain node is measured by the words which have to be inserted or deleted from the phrase in order to be exactly covered by the node. Zhang et al. (2008) use tree sequences to annotate phrases in a tree-to-tree translation model. A tree sequence is an ordered sequence of tree fragments covering a phrase, which helps to provide coverage for phrases which do not correspond to a single constituent.

Hassan et al. (2009) present an approach which builds a full CCG-based parse tree of the translation output during decoding. They propose a Dependency-based Direct Translation Model which integrates an incremental CCG dependency-based language model parser built on an incremental version of the CCGBank (Hockenmaier and Steedman, 2007) into the Direct Translation Model (Ittycheriah and Roukos, 2007). The parser builds a fully connected dependency-based structure for the translation output incrementally during decoding, which enabled to handle long range dependencies and prune ungrammatical translations.

Recently, applying syntactic constraints in syntax-augmented HPB SMT systems in a soft manner has been demonstrated to be more effective than hard syntactic constraints (Venugopal et al., 2009; Chiang, 2010). Applying soft syntactic constraints means that the derivations which violate the syntactic constraints imposed by the model are not prevented per se, but the system has the flexibility to decide when to apply these syntactic constraints. Marton and Resnik (2008) incorporate source-side soft constituency features into the HPB SMT model. The idea of a soft constituency feature was originally proposed by Chiang (2005) to encourage the HPB SMT system to favour phrases which correspond to syntactic constituents during translation. The feature awards the usage of a rule when its source phrase is covered by a source constituent. Marton and Resnik (2008) developed this feature by penalizing the usage of a rule when its source phrase violates phrase boundaries.

Furthermore, instead of using a single feature for all constituent types, they use finer grained features which distinguish between source constituent types, enabling the system to learn constituent-based preferences.

Zollmann and Vogel (2010) try to give the SAMT model the flexibility to violate the syntactic constraints imposed by nonterminal labels by incorporating both labelled and unlabelled rules in the translation model. A binary feature which distinguishes syntax-augmented rules from their unlabelled counterparts is added to each rule and the feature weight is tuned using MERT. They also try to handle low-probability syntax-augmented rules by incorporating probabilities calculated on the unlabelled version of the rules as features into the syntax-augmented rules. Venugopal et al. (2009) transform the syntactic constraints in the SAMT translation model into a syntactic feature integrated into the log-linear model. They use an unlabelled translation model during decoding. Another SAMT-based syntactic model, which measures the probability of different labellings of each hierarchical rule, is used to calculate the value of the syntactic feature at each nonterminal replacement during decoding.

Chiang (2010) incorporates soft source- and target-based syntactic constraints into a tree-to-tree translation model. Similar to the approach followed by Venugopal et al. (2009), the model uses rules without syntactic annotation during decoding while chart items bear the syntactic labels attached to partial translations. The syntactic constraints in this model are applied through a set of syntactic features in the log-linear model. These syntactic features are calculated for each unlabelled rule application used in the translation based on a syntactically annotated instance of the rule extracted from source- and target-side parse trees. In contrast to the approach proposed by Venugopal et al. (2009), Chiang (2010) softens the matching constraint performed during nonterminal replacement by allowing the model to learn preferences toward considering groups of different syntactic categories to be compatible with each other. Thus the mismatch between any two syntactic categories in the same compatibility group is not discouraged.

Huang et al. (2010) use soft syntactic features in the HPB SMT model based on latent syntactic category distributions learnt from source-side parse trees. These latent categories are extracted according to the hierarchy and the syntax of the phrases extracted from source-side parse trees. Then, real-valued feature vectors based on latent category distribution are assigned to phrases and nonterminals. These feature vectors help to convert the strict equality test between the categories of the nonterminal and the categories of the replacing phrase performed during translation into a similarity score calculated on the feature vectors of each of the nonterminal and the phrase. Moreover, latent syntactic categories help to exploit syntactic similarities between syntactic categories assigned to phrases extracted from source-side parse trees. Stein et al. (2010) use soft dependency-based features in String-to-Dependency HPB SMT model. The features, which are added to the log-linear model, soften the strict well-formedness constraint applied in the previous model of Shen et al. (2008) which allows only for well-formed dependency structures to participate in translation.

5.2.1 Preference Grammars

Preference Grammars is an approach proposed by Venugopal et al. (2009) to soften the syntactic constraints and reduce probability fragmentation in the original SAMT model (Zollmann and Venugopal, 2006). They use a syntactic feature p_{syn} in the log-linear model to score unlabelled derivations according to the degree to which they comply with the syntactic constraints learnt from the syntactically annotated training data. The use of unlabelled derivations helps to reduce probability fragmentation among different labellings of the same derivation and prevent them from competing with each other during translation. Furthermore, applying the syntactic constraints in a soft way loosens the strict constraints imposed on the decoding search space which limit the set of translations explored.

The Preference Grammars formalism uses an unlabelled translation model (the same translation model extracted by the HPB SMT system) to perform translation.

The formalism uses another syntactically annotated translation model (the same translation model extracted by the SAMT system) to calculate the syntactic feature p_{syn}. This model specifies the probability distribution of different labellings of each unlabelled rule r, which is called the preference distributions p_{pref}. A rule labelling is a vector $\overrightarrow{h} =< h_0, h_1, ..., h_k >$ which specifies the syntactic labels attached to the left-hand side (h_0) and the nonterminals on the right-hand side of the rule ($h_1, ..., h_k$). The syntactic feature p_{syn} is calculated for each derivation d as in (5.1):

$$p_{syn}(d) = \prod_{i=1}^{|d|} \phi_i \tag{5.1}$$

where ϕ_i is a factor calculated for each nonterminal n_i replacement in the derivation d, and $i \in 1..|d|$ is the index of the nonterminal in the derivation. The calculation of ϕ_i depends on the left-hand-side preference distribution u_i. u_i specifies the probability of each possible label assigned to the left-hand side of a subtranslation to replace the nonterminal n_i. Thus, u is added as an extra information to chart items.

During translation, when a rule r is applied on a chart item X, which has a left-hand-side preference distribution u, the factor ϕ is calculated as in (5.2):

$$\phi = \sum_{h \in hargs(n)} u(h) \tag{5.2}$$

where $hargs(n)$ specifies the set of different possible labels for the nonterminal n according to the rule r. The value of $u(h)$ when the rule does not have any nonterminal on the right-hand side is set to $p_{pref}(h|r)$. Then, the left-hand-side preference distribution v of the chart item, which results from applying the rule r on the chart item X with left-hand-side preference distribution u, is calculated as in (5.3):

$$v(h) = \sum_{\overrightarrow{h'} = <h, h'_1, ..., h'_k> \in hargs(r)} p_{pref}(\overrightarrow{h'}|r) \prod_{i=1}^{k} u(h'_i) \tag{5.3}$$

where $hargs(r)$ is the set of all different labellings of the rule r. Finally, the left-hand-side preference distribution v is renormalized as in (5.4):

$$v'(h) = \frac{v(h)}{\sum_{h'} v(h')} \tag{5.4}$$

5.3 Extended CCG-based Syntactic Labels

In Chapter 3, we extracted nonterminal labels which consist of a single CCG category. We demonstrated that CCG flexible structures allow a better label coverage than SAMT labels (cf. Section 3.5.5 page 68). However, Table 3.8 (page 74) showed that single-category CCG labels do not cover about one third of the total extracted phrases. In order to increase label coverage, we extend the definition of the nonterminal label to be composed of more than one CCG category. Therefore, if there is no CCG category in the CCG parsing chart cell which covers a phrase, the highest-scoring sequence of CCG categories with a minimum number of CCG categories is extracted from the CCG parsing chart cells covering the phrase and used as the phrase label. This is similar to the "plus" operator used to combine the constituent labels in SAMT, with the difference is that we always keep the minimum number of CCG categories per label using CCG combinatory rules, which helps to prevent label redundancy by maintaining a minimum set of extended CCG labels and thus avoid label proliferation.

Figure 5.2 shows a set of phrases along with their extended CCG labels extracted from the sentence illustrated in Figure 5.1. Extended CCG labels are extracted from the forest trees enclosed in the parsing chart built for the sentence. However, for the sake of clarity, we only illustrated the best sequence of CCG supertags assigned to the words of the sentence in Figure 5.1. In this example, the phrase *contributes 600* has an extended CCG label composed of two categories: $(S[dcl]\backslash)/NP$ and N/N, which are the categories of the words *contributes* and *600*, respectively. These categories cannot be further combined with each other using any of the CCG combinatory rules, that is why we formed a composed label combining the two categories. Another example is the phrase *Germany contributes 600 million to finance SFD*, which is

assigned $S[dcl] + N/N$ as an extended CCG label. This label is composed of two categories: $S[dcl]$ which results from combining the categories of the words of the phrase *Germany contributes 600 million to finance*, and N/N which is the category of the word *SFD*. Thus, the extended CCG label $S[dcl] + N/N$ is assigned to this phrase, which is the shortest sequence of CCG categories with the highest score covering the phrase.

Germany	**contributes**	**600**	**million**	**to**	**finance**	**SFD**	**projects**
NP	(S[dcl]\NP)/NP	N/N	N	(S[to]\NP)/(S[b]\NP)	(S[b]\NP)/NP	N/N	N

Figure 5.1: An English sentence along with the best sequence of CCG supertags assigned to its words.

We define the **degree** of the extended label to be the number of CCG categories in the label. In our previous example, the extended CCG label N of the phrase *SFD projects* is of degree one, and the extended CCG label $(S[b]\backslash NP)/NP + N/N$ of the phrase *finance SFD* is of degree two. The degree of the system which uses extended CCG labels is defined to be the maximum degree of the labels used in the model.

Phrase	**Extended CCG label**
Germany contributes	S[dcl]/NP
contributes 600	(S[dcl]\NP)/NP+N/N
finance SFD	(S[b]\NP)/NP+N/N
SFD projects	N
contributes 600 million	S[dcl]\NP
million to finance	N+(S[to]\NP)/NP
contributes 600 million to finance SFD	S[dcl]\NP+N/N
Germany contributes 600 million to finance SFD	S[dcl]+N/N

Figure 5.2: A set of phrases extracted from the sentence illustrated in Figure 5.1 along with the corresponding extended CCG labels.

5.4 CCG-augmented Glue Grammar

Figure 5.3 shows the derivation tree of the English translation produced by our CCG-augmented HPB system which uses the syntax-free glue grammar for the Arabic

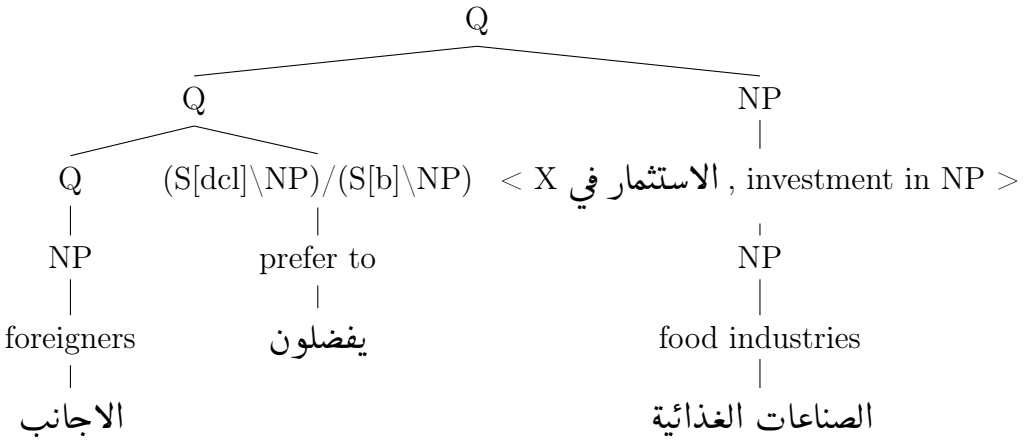

Figure 5.3: The derivation tree of the English translation produced by the CCG-augmented HPB system which uses the syntax-free glue grammar for the Arabic sentence الاجانب يفضلون الاستثمار في الصناعات الغذائية.

sentence الاجانب يفضلون الاستثمار في الصناعات الغذائية.[1] A glue grammar rule is used to concatenate the phrase *foreigners prefer to* with the phrase *investment in food industries*, which apparently leads to a grammatically wrong translation. If glue grammar rules perform grammatical checking on the phrases being concatenated, this translation would be prevented. Thus, the lack of syntax awareness in glue grammar rules limits the scope of the syntactic constraints applied in the CCG-augmented HPB system, which results in ungrammatical translations being produced.

Instead of concatenating phrases during glue grammar rule application without applying any syntactic constraint, we try to provide glue grammar rules with syntactic knowledge by augmenting them with CCG combinatory rules. CCG combinatory rules are binary rules, which makes them suitable to be applied on glue grammar rules which are also binary rules. First, we change the definition of the glue grammar rule in (2.8) (cf. Section 2.4.1 page 24) to the one in (5.5):

$$X \rightarrow < \ X_1 \ X_2 \ , \ X_1 \ X_2 \ > \tag{5.5}$$

[1] Q in the derivation trees replaces the S symbol in the original glue grammar rules (cf. glue grammar rules (2.8) and (2.9) page 24) to avoid confusion with the S symbol used in syntactic constituents.

This removes the left-balance constraint from the construction of glue grammar rule application, which allows more flexibility in applying glue grammar rules. Additionally, this rule allows the application of glue grammar rules and hierarchical rules to alternate, which gives further flexibility. Secondly, we build a metric which judges the grammaticality of concatenating two phrases at each glue grammar rule application based on their extended CCG labels. The calculation of this grammaticality metric $m_{CCGglue}$ is based on an extended CCG label model. This model is extracted using relative frequency counts from the target side of the training corpus which is annotated with extended CCG labels for each subphrase in each sentence. Thus, the probability of a label l of degree n is calculated according to the extended CCG label model as in (5.6):

$$p_{extCCG}(l) = \frac{count(l)}{\sum_{l_i \in L_n} count(l_i)} \tag{5.6}$$

where L_n is the set of extended CCG labels with degree n.

Whenever two phrases with left-hand-side preference distributions u_1 and u_2 are concatenated using a glue grammar rule, the value of the syntactic feature p_{syn} is calculated according to the algorithm illustrated in Figure 5.4. The algorithm also returns the left-hand-side preference distribution v' of the resulting phrase. The algorithm iterates in lines 4 to 13 on each label pair l_i and l_j from u_1 and u_2, respectively. Firstly, the function ***parse*** in line 5 applies all possible CCG combinatory rules on the extended CCG label $l_i + l_j$ and returns the extended CCG label l_r with the minimum number of CCG categories. Then, the function ***numOfCats*** in line 6 calculates the number of CCG categories in the resulting label l_r. If l_r is composed of one CCG category, the two phrases are likely to constitute a grammatical phrase and the grammaticality metric $m_{CCGglue}$ is set to 1 in line 7. Otherwise, the grammaticality metric $m_{CCGglue}$ is set to the probability of l_r according to the extended CCG label model $p_{extCCG}(l_r)$ in line 9. The values of v and p_{syn} are updated in lines 11 and 12. After the iteration ends, v is renormalized in lines

15-21. Finally the value of the syntactic feature p_{syn} for the current glue grammar rule application is returned along with the left-hand-side preference distribution v' for the resulting phrase.

Algorithm 1 Calculate p_{syn} for CCG-augmented Glue Grammar Rule Application

Input: u_1, u_2: the left-hand-side preference distributions for the first and second phrase, respectively.

1: $v \leftarrow$ assign 0 to all units
2: $p_{syn} \leftarrow 0$
3: $m_{CCGglue} \leftarrow 0$
4: **for all** l_i **in** u_1 **and** l_j **in** u_2 **do**
5: $\quad l_r \leftarrow parse(l_i + l_j)$
6: $\quad$ **if** $numOfCats(l_r) = 1$ **then**
7: $\quad\quad m_{CCGglue} \leftarrow 1$
8: $\quad$ **else**
9: $\quad\quad m_{CCGglue} \leftarrow p_{extCCG}(l_r)$
10: $\quad$ **end if**
11: $\quad v(l_r) \leftarrow v(l_r) + \big(u_1(l_i) * u_2(l_j)\big)$
12: $\quad p_{syn} \leftarrow p_{syn} + \big(u_1(l_i) * u_2(l_j) * m_{CCGglue}\big)$
13: **end for**
14: $sum_v \leftarrow 0$
15: **for** $i = 1...|v|$ **do**
16: $\quad sum_v \leftarrow sum_v + v[i]$
17: **end for**
18: $v' \leftarrow$ assign 0 to all units
19: **for** $i = 1...|v|$ **do**
20: $\quad v'[i] \leftarrow v[i]/sum_v$
21: **end for**
22: **return** p_{syn}, v'

Figure 5.4: The algorithm for calculating the syntactic feature p_{syn} during glue grammar rule application.

Augmenting glue grammar rules with CCG combinatory rules enables a full parse tree of the translation output to be built, which helps to extend the scope of the syntactic constraints to cover the whole translation output. The grammaticality feature p_{syn} calculated during glue grammar and hierarchical rule application helps to guide the decoding process towards applying the CCG-based syntactic constraints without restraining it with hard syntactic constraints.

Figure 5.5 shows the derivation tree of the English translation produced by the

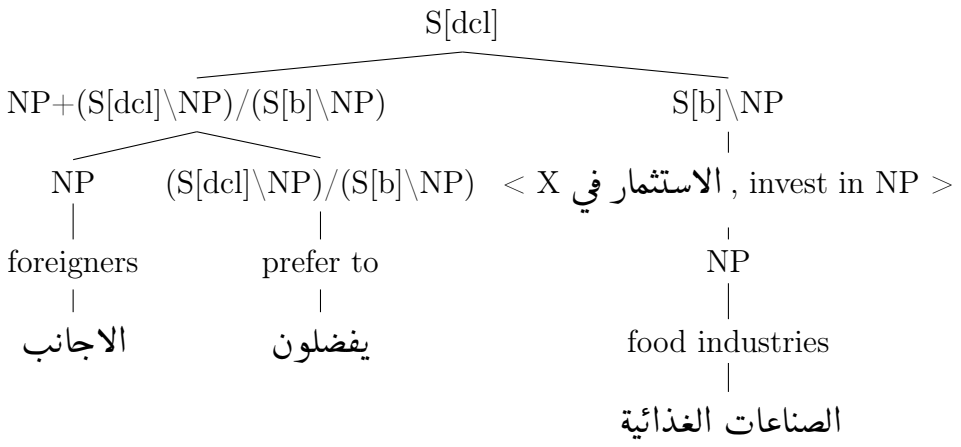

Figure 5.5: The derivation tree of the English translation produced by the CCG-augmented HPB system for the Arabic sentence الاجانب يفضلون الاستثمار في الصناعات الغذائية using the CCG-augmented glue grammar.

CCG-augmented HPB system using the CCG-augmented glue grammar for the same Arabic sentence in Figure 5.3. We can see that in contrast to the ungrammatical translation produced in the case of the non-syntactically aware glue grammar in Figure 5.3, the CCG-augmented glue grammar enables our CCG-augmented HPB system to produce a grammatical translation. Furthermore, we can see that the derivation tree in Figure 5.5 represents a full parse tree covering the whole translation output with the sentence category $S[dcl]$ at the root of the translation. In this example, when the phrases: *foreigners prefer to* and *invest in food industries* are concatenated via a glue grammar rule, CCG combinatory rules are applied on the categories of their extended CCG labels NP+(S[dcl]\NP)/(S[b]\NP and S[b]\NP, which results in the sentence category $S[dcl]$ at the root of the derivation tree. Thus, applying CCG combinatory rules during glue grammar rule application helps to enforce the grammaticality of the concatenation of two phrases by making sure that the categories of the concatenated phrases are compatible with each other. Hence, the phrase *invest in food industries* which starts with the infinitive verb *invest* and bears the category S[b]\NP is chosen to come after the phrase *prefer to* which bears the category (S[dcl]\NP)/(S[b]\NP) that apparently needs the category S[b]\NP as an argument to its right. By contrast, in Figure 5.3, we can see that syntactic

constraints are only applied during the application of the hierarchical rule which produces the phrase *investment in food industries*. Glue grammar rule application between the phrases *foreigners prefer to* and *investment in food industries* does not impose any syntactic constraint, which causes a noun phrase after to erroneously produce the phrase *prefer to*.

5.5 Experiments

In our experiments, we try to explore the effect of each approach to extending the syntactic constraints in our CCG-augmented HPB SMT system presented in Sections 5.3 and 5.4. We examine the performance of our extended CCG labels from different degrees under soft and hard syntactic constraints. We also conduct experiments which examine the combination of the extended CCG labels from different degrees with the CCG-augmented glue grammar. Thus, the CCG-augmented HPB systems in our experiments are divided into three groups:

- *CCG*: the systems which use extended CCG labels of different degrees using hard syntactic constraints with a syntax free glue grammar.
- CCG_{pref}: the systems which use extended CCG labels of different degrees using soft syntactic constraints with a syntax-free glue grammar.
- CCG_{glue}: the systems which use extended CCG labels of different degrees using soft syntactic constraints with the CCG-augmented glue grammar.

We also try to build systems which use extended CCG labels of different degrees using hard syntactic constraints with the CCG-augmented glue grammar. However, empty translations are produced for some sentences, because of the sparseness of the syntax-augmented rules which are sometimes unable to build a full translation for the input sentence under hard syntactic constraints. That is why we included these systems in the coverage analysis only (cf. Tables 5.9 and 5.10).

Furthermore, we try to test the performance of the different systems varying one aspect of translation difficulty, namely the source sentence length. Shorter sentences are generally easier to translate than longer sentences. Thus, for the Arabic-to-English news data experiments, we built three versions of each system: the first version is tuned and tested on short sentences of length ranging from one word to eleven words. The second version is tuned and tested on long sentences of more than eleven words of length. The third version is tuned and tested on a combination of short and long sentences.

5.5.1 Data and Settings

We tested our approach on the IWSLT Chinese–English and Arabic–English data sets described in Section 3.5.1 (page 61). In addition, we tested our approach on a corpus comprised of 40k sentence pairs selected randomly from the Arabic News corpus from LDC.[2] The long and short development and test sets contain 500 and 452 sentence pairs, respectively. The combined development and test sets result from combining the short and long development and test sets, respectively.

The experimental settings used in our experiments in this chapter are the same settings used in our experiments in Section 3.5.1 (page 61).

We build our HPB baseline using the Moses Chart Decoder[3] (Hoang et al., 2009b) with maximum phrase length and maximum rule span set to 12 words.[4] Hierarchical rules extracted contain up to 2 nonterminals. Maximum chart span is set to 20 words. Cube pruning pop-up limit is set to 1000.

We use the CCG parser from C&C tools[5] (Clark and Curran, 2007) to parse the English side of the training data from which we extract the CCG-augmented HPB models. The CCG-augmented HPB model which specifies different labellings of

[2]http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2004T17

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

[4]The version of the Moses Chart Decoder used to build the systems in this chapter is newer than the one used in the previous chapters, that is why a difference in the scores of the CCG-augmented HPB system and the HPB baseline system between this chapter and the previous chapters might be noticed.

[5]http://svn.ask.it.usyd.edu.au/trac/candc/

each unlabelled hierarchical rule is the same as the one built in Section 3.5 (page 60) but excluding the rules which use the X label. We annotate the target side of the training data with extended CCG labels extracted from the parsing chart built by the C&C parser for target-side sentences in the training data. We modified the parser code to add the functionality to find the highest-scoring minimum sequence of CCG categories out of a sequence of CCG categories. We then extract the CCG-augmented HPB models and the extended CCG labels model from the annotated training data. We also use the C&C parser to perform category combination during the CCG-augmented glue grammar rule application. We use the Moses Chart Decoder to build our CCG-augmented HPB systems with the same settings as the HPB baseline system.

For our CCG-augmented HPB systems which apply soft syntactic constraints according to the Preference Grammars paradigm, we modified the code of the Moses Chart Decoder to calculate the syntactic feature p_{syn} as part of the log-linear model and incorporate the preference grammar distribution into chart items. The number of labels in the preference grammar distribution for each chart item is limited to the five most probable labels. The number of different labellings extracted for each hierarchical rule is limited to the five most probable labellings. For the CCG-augmented HPB system which uses the CCG-augmented glue grammar, we modified the code of the Moses Chart Decoder so that it interacts with the C&C parser to perform category combination during glue grammar rule application.

5.5.2 Experimental Results on IWSLT Data

In this section we examine the effect of our approaches to extending the CCG-based syntactic constraints on Arabic-to-English and Chinese-to-English translation in the travel speech expressions domain using the IWSLT data. We try out extended labels of degree ranging from one to five using hard and soft syntactic constraints. We also examine the effect of using the CCG-augmented glue grammar with extended labels from different degrees. Tables 5.2 and 5.3 show the BLEU, TER and METEOR

scores of our CCG-augmented HPB systems and the HPB baseline system for Arabic-to-English and Chinese-to-English translation on the IWSLT data, respectively.

System	BLEU	TER	METEOR
HPB	52.90	31.06	**71.51**
CCG_1	51.54	32.32	70.33
CCG_2	51.63	32.27	71.21*
CCG_3	52.32*	32.13	70.63
CCG_4	51.11	33.20	70.87
CCG_5	51.93	32.04*	70.01
CCG_{pref1}	52.83	31.13	70.77
CCG_{pref2}	53.38	30.92	70.60
CCG_{pref3}	53.10	30.76*	70.77
CCG_{pref4}	53.09	30.76*	70.62
CCG_{pref5}	**53.76***	30.76*	71.05*
CCG_{glue1}	53.06	30.95	70.63
CCG_{glue2}	52.84	30.97	70.54
CCG_{glue3}	52.66	31.04	69.60
CCG_{glue4}	53.17	30.90	70.88*
CCG_{glue5}	53.51*	**30.38***	70.81

Table 5.2: Experimental results for our CCG-augmented HPB systems which use extended syntactic constraints and the HPB baseline system on the Arabic–English IWSLT data. CCG_{pref} refers to the CCG-augmented HPB systems which use soft syntactic constraints. CCG_{glue} refers to the systems which use the CCG-augmented glue grammar. The subscripted number at the end of the name of each system indicates its degree. The asterisk marks the best score achieved in each system group.

Arabic-to-English Experimental Results

Table 5.2 demonstrates that for the hard syntactic constraints setting, the 3-category CCG-augmented HPB SMT system is the best-performing system in terms of BLEU score. The table also shows that none of the hard constraint CCG-augmented HPB systems was able to outperform the HPB baseline system. Increasing the degree of the extended CCG labels from one to three leads to increases of 0.09 and 0.69 absolute BLEU points, which corresponds to relative improvements of 0.17% and 1.34%, respectively. However, the BLEU score decreases when the degree of the extended CCG labels increases from three to four by 1.21 absolute BLEU points, which cor-

responds to a 2.31% relative decrease. Increasing the degree of the extended CCG labels from four to five leads to an improvement of 0.82 absolute BLEU points, which corresponds to a 1.6% relative improvement.

For the soft syntactic constraint setting, we can see that the 5-category CCG-augmented HPB system is the best-performing system in terms of BLEU, TER and METEOR scores, outperforming the HPB baseline system by 0.86 absolute BLEU points, which corresponds to a 1.63% relative improvement. The result of the paired bootstrap resampling test demonstrates that this improvement is statistically significant at p-level=0.05. Table 5.2 also shows that using soft syntactic constraints leads to significant improvements over the systems which use hard syntactic constraints. Moreover, the CCG-augmented HPB systems from different degrees which use soft constraints outperform the HPB baseline system in terms of BLEU score except for the 1-category CCG-augmented HPB system. The table also demonstrates that increasing the degree of the extended CCG labels does not show a consistent improvement for the systems in this group.

As for the CCG-augmented HPB systems which use the CCG-augmented glue grammar, Table 5.2 demonstrates that the 5-category CCG-augmented HPB system achieves the best BLEU and TER scores. Using the CCG-augmented glue grammar helps to improve the BLEU score of each of the 1-category and 4-category CCG-augmented HPB systems by 0.23 and 0.08 absolute BLEU points, which corresponds to relative improvements of 0.43% and 0.15%, respectively. Although using the CCG-augmented glue grammar does not help to improve the BLEU score of the 5-category CCG-augmented HPB system, it helps to improve its TER score, which is the best TER score among all the examined systems.

In general, our Arabic-to-English experimental results on the IWSLT data demonstrate that the CCG-augmented HPB systems which use soft syntactic constraints achieve the best performance among other CCG-augmented HPB systems, significantly outperforming the HPB baseline system and the hard constraint CCG-augmented HPB systems. Moreover, using the CCG-augmented glue grammar with

System	BLEU	TER	METEOR
HPB	48.29	35.28	65.85
CCG_1	46.01	34.86*	63.01
CCG_2	46.16	35.51	62.84
CCG_3	44.15	37.14	62.30
CCG_4	45.61	35.67	63.13
CCG_5	46.28*	35.12	63.19*
CCG_{pref1}	49.73	34.04	66.66*
CCG_{pref2}	48.19	35.46	64.67
CCG_{pref3}	49.94*	34.02*	66.29
CCG_{pref4}	48.32	34.54	65.07
CCG_{pref5}	49.44	34.10	65.76
CCG_{glue1}	49.43	34.96	66.16
CCG_{glue2}	49.26	34.44	65.49
CCG_{glue3}	**50.73***	**33.50***	**66.67***
CCG_{glue4}	48.58	34.31	65.42
CCG_{glue5}	49.00	34.28	65.61

Table 5.3: Experimental results for our CCG-augmented HPB systems which use extended syntactic constraints and the HPB baseline system on the Chinese–English IWSLT data.

extended CCG labels from different degrees does not always help to improve the performance of the corresponding systems. We also notice that increasing the degree of the extended CCG labels causes fluctuation in performance under both soft and hard syntactic constraints and using the CCG-augmented glue grammar.

Chinese-to-English Experimental Results

Table 5.3 shows that using hard syntactic constraints, the CCG-augmented HPB system which uses extended CCG labels of degree five is the best performing system in terms of BLEU and METEOR scores. We can also see that none of the CCG-augmented HPB systems in this group outperforms the HPB baseline system except for the 1-category CCG-augmented HPB system which outperforms the HPB baseline system by 0.42 absolute TER points, which corresponds to a 1.2% relative improvement. Increasing the degree of the extended CCG labels for the systems in this group does not demonstrate a consistent improvement.

For the soft syntactic constraints setting, Table 5.3 shows that the 3-category

CCG-augmented HPB system is the best-performing system in terms of BLEU and TER scores. Moreover, the 3-category CCG-augmented HPB system outperforms the HPB baseline systems by 1.65 absolute BLEU points, which corresponds to a 3.4% relative improvement. The paired bootstrap resampling test demonstrates that this improvement is statistically significant at p-level=0.05. Similar to our Arabic-to-English experiments, using soft syntactic constraints helps to achieve significant improvements over the hard constraint CCG-augmented HPB systems. Furthermore, using soft syntactic constraints helps to improve performance over the HPB baseline system except for the 2-category CCG-augmented HPB system. Similar to the hard syntactic constraints setting, increasing the degree of the extended CCG labels for the soft constraint systems does not demonstrate a consistent improvement.

Table 5.3 demonstrates that using the CCG-augmented glue grammar with extended CCG labels increases the performance of the systems of degrees from two to four by 1.07, 0.79 and 0.26 absolute BLEU points, which corresponds to relative improvements of 2.22%, 1.58% and 0.54%, respectively. However, the paired bootstrap resampling test shows that these improvements are not statistically significant at p-level=0.05. We can see also that using the CCG-augmented glue grammar causes the performance of the 1-category and 5-category CCG-augmented HPB systems to degrade. It is worth noting that the 3-category CCG-augmented HPB system which uses the CCG-augmented glue grammar achieves the best performance measured in BLEU, TER and METEOR scores among all the examined systems.

In general, our Chinese-to-English experimental results on the IWSLT data show that using soft syntactic constraints helps to significantly improve the performance of the CCG-augmented HPB systems from different degrees over the HPB baseline system and the hard constraint CCG-augmented HPB systems. Although using the CCG-augmented glue grammar helps to achieve the best performance among the examined systems, it does not always help to improve performance. Similar to the Arabic-to-English IWSLT experimental results, increasing the degree of the

extended CCG labels does not demonstrate a consistent improvement across the examined systems.

System	Labels	Rule Table Size	%X
CCG_1	518	2895825	28.13%
CCG_2	8063	3147353	6.34%
CCG_3	18220	3175309	1.28%
CCG_4	22582	3179796	0.26%
CCG_5	23709	3180365	0.048%

Table 5.4: Number of different labels, rule table size and percentage of unlabelled nonterminals in the rule table of the CCG-augmented HPB systems of degrees from one to five built on the Arabic–English IWSLT data.

Label Coverage

We try to examine the effect of increasing the degree of the CCG-augmented HPB systems on label coverage, the number of different labels and the size of the translation model for Arabic-to-English and Chinese-to-English translation on the IWSLT data. Therefore, we measure label coverage in terms of the percentage of unlabelled nonterminals in the rule tables of the CCG-augmented HPB systems of different degrees. We also measure the number of different labels used to annotate the training data by these systems in addition to the size of their translation models in terms of number of rules. Tables 5.4 and 5.5 show that increasing the degree of the extended CCG labels from one to three significantly increases the number of different labels and the rule table size, and significantly decreases the percentage of unlabelled nonterminals. The tables also show that the pace of the growth in the rule table size and labels number slows down when increasing the degree of the extended CCG labels above three. In general, increasing the degree of the extended CCG labels increases the number of different labels and the rule table size, and decreases the number of unlabelled nonterminals. We can also see that the number of different labels in the Arabic-to-English systems of degrees from two to five is significantly smaller than their Chinese-to-English counterparts. Bearing in mind that the size of the Chinese–English IWSLT data is about 3 times the size of the Arabic–English

IWSLT data, this highlights the effect of the size of the training data on the number of different labels.

System	Labels	Rule Table Size	%X
CCG_1	584	4983289	31%
CCG_2	12040	5334511	7.6%
CCG_3	30632	5379606	1.71%
CCG_4	40493	5386289	0.38%
CCG_5	43636	5387167	0.086%

Table 5.5: Number of different labels, rule table size and percentage of unlabelled nonterminals in the rule table of the CCG-augmented HPB systems of degrees from one to five built on the Chinese–English IWSLT data.

5.5.3 Experimental Results on News Data

In this section, we examine the performance of our approaches to extending the syntactic constraints on the Arabic-to-English translation in the news domain. We examine the performance of our CCG-augmented HPB systems which use extended CCG labels of degrees from one to three under soft and hard syntactic constraints. We also explore the effect of using the CCG-augmented glue grammar on the performance of the different systems. Furthermore, we try to examine the effect of the source sentence length on the performance of the different systems. We first examine the performance of our systems on short sentences (less than twelve words) and then on long sentences (equal to or more than twelve words). Finally, we test our systems on a combination of the short and long sentences.

Short Sentences Translation

Table 5.6 shows the experimental results for our CCG-augmented HPB systems which use extended syntactic constraints and the HPB baseline system on Arabic–English short sentences from the news data. For the CCG-augmented HPB systems which use hard syntactic constraints, we can see that increasing the degree of the extended CCG labels helps to improve performance in terms of BLEU, METEOR

System	BLEU	TER	METEOR
HPB	41.96	49.39	68.77
CCG_1	41.05	48.97	69.09
CCG_2	42.29	48.25*	69.59
CCG_3	**43.18***	48.25*	**69.78***
CCG_{pref1}	41.39	49.81	68.78
CCG_{pref2}	41.05	50.47	67.30
CCG_{pref3}	42.97*	**47.56***	69.23*
CCG_{glue1}	41.47	49.53	68.70
CCG_{glue2}	43.17*	48.22*	69.48*
CCG_{glue3}	42.97	48.58	69.41

Table 5.6: Experimental results for our CCG-augmented HPB systems which use extended syntactic constraints and the HPB baseline system on Arabic–English short sentences from the news data.

and TER scores. Increasing the degree of the extended CCG labels from one to two helps to improve performance by 1.24 absolute BLEU points, which corresponds to a 3% relative improvement. The paired bootstrap resampling test shows that this improvement is statistically significant at p-level=0.05. Increasing the degree of the extended CCG labels from two to three improves performance by 0.89 absolute BLEU points, which corresponds to a 2.1% relative improvement. The paired bootstrap resampling test demonstrates that this improvement is statistically significant at p-level=0.05. Table 5.6 also shows that both the 2-category and 3-category CCG-augmented HPB systems outperform the HPB baseline system by 0.33 and 1.22 absolute BLEU points, which corresponds to relative improvements of 0.79% and 2.9%, respectively. The improvement in the case of the 3-category CCG-augmented HPB system is statistically significant whereas it is statistically insignificant in the case of the 2-category CCG-augmented HPB system at p-level=0.05. These improvements in BLEU score achieved over the HPB baseline system are corroborated by improvements with respect to TER and METEOR.

Table 5.6 also demonstrates that using soft syntactic constraints helps to improve performance only in the case of the extended CCG labels of degree one by 0.34 absolute BLEU points, which corresponds to a 0.83% relative improvement.

This improvement is demonstrated to be statistically insignificant by the paired bootstrapping test at p-level=0.05. The best-performing system in this group is the 3-category CCG-augmented HPB system, which achieves the best TER score among all the examined systems. We can also see that increasing the degree of the extended CCG labels for the soft constraint systems does not show a consistent improvement.

Table 5.6 shows that using the CCG-augmented glue grammar helps to improve the performance of the 1-category and 2-category CCG-augmented HPB systems by 0.08 and 2.12 absolute BLEU points, which corresponds to relative improvements of 0.19% and 5.16%, respectively. However, using the CCG-augmented glue grammar does not affect the BLEU score of the 3-category CCG-augmented HPB system. Table 5.6 also shows that increasing the degree of the extended CCG labels for the systems which use the CCG-augmented glue grammar does not show a consistent improvement.

In general, our Arabic-to-English experimental results on translation of short sentences from the news domain demonstrate that the CCG-augmented glue grammar helps to improve the performance over both the hard and soft syntactic constraint systems except for the extended CCG labels of degree three. The 3-category CCG-augmented HPB system which uses hard syntactic constraints is the best-performing system in terms of BLEU and METEOR scores. Neither using the soft syntactic constraints nor the CCG-augmented glue grammar is able to improve performance over the 3-category CCG-augmented HPB system which uses the hard syntactic constraints. In addition, increasing the degree of the extended CCG labels under hard syntactic constraints setting helps to improve performance, whereas it does not show a consistent improvement under the soft syntactic constraints setting or using the CCG-augmented glue grammar.

Long Sentences Translation

Table 5.7 demonstrates the experimental results for our CCG-augmented HPB systems which use extended syntactic constraints and the HPB baseline system on

System	BLEU	TER	METEOR
HPB	36.04	54.82	63.27
CCG_1	35.77	54.93	63.29
CCG_2	36.94*	53.68*	63.92*
CCG_3	36.08	55.17	62.92
CCG_{pref1}	36.35	53.91	63.69
CCG_{pref2}	37.05	53.33	63.95
CCG_{pref3}	**37.62***	**52.87***	**64.50***
CCG_{glue1}	35.67	55.00	62.78
CCG_{glue2}	36.18	54.91	62.74
CCG_{glue3}	36.95*	53.87*	63.47*

Table 5.7: Experimental results for our CCG-augmented HPB systems which use extended syntactic constraints and the HPB baseline system on Arabic–English long sentences from the news data.

Arabic–English long sentences from the news data. Under the hard syntactic constraints setting, the table shows that increasing the degree of the extended CCG labels from one to two increases performance in terms of all the metrics. The 2-category CCG-augmented HPB system outperforms the 1-category system by 1.17 absolute BLEU points, which corresponds to a 3.27% relative improvement. The paired bootstrap resampling test demonstrates that this improvement is statistically significant at p-level=0.05. The 2-category CCG-augmented HPB system also outperforms the HPB baseline system in terms of BLEU, METEOR and TER scores. The 2-category CCG-augmented HPB system achieves a higher BLEU score than the HPB baseline system by 0.9 absolute BLEU points, which corresponds to a 2.5% relative improvement. This improvement is statistically significant at p-level=0.05 according to the paired bootstrap resampling test. Increasing the degree of the extended CCG labels from two to three decreases performance by 0.86 absolute BLEU points, which corresponds to a 2.32% relative decrease.

For the systems which use soft syntactic constraints, Table 5.7 shows that using soft syntactic constraints helps to improve performance over the hard syntactic constraint systems in terms of all the metrics. Using soft syntactic constraints with the extended CCG labels of degrees one, two and three increases the BLEU score

over the corresponding hard syntactic constraint systems by 0.58, 0.11 and 1.54 absolute BLEU points, which corresponds to relative improvements of 1.62%, 0.3% and 4.27%, respectively. Among these improvements, the improvement achieved over the hard constraint 3-category system is the only statistically significant improvement at p-level=0.05 according to the bootstrap resampling test. We can also see that increasing the degree of the extended CCG labels under the soft syntactic constraints setting helps to improve performance in terms BLEU, METEOR and TER scores. Increasing the degree of the extended CCG labels from one to two improves the BLEU score by 0.7 absolute BLEU points, which corresponds to a 1.9% relative improvement. The paired bootstrap resampling test demonstrates that this improvement is not statistically significant at p-level=0.05. Increasing the degree of the extended CCG labels from two to three improves the BLEU score by 0.57 absolute BLEU points, which corresponds to a 1.54% relative improvement. This improvement is demonstrated by the paired bootstrap resampling test to be statistically insignificant at p-level=0.05.

Generally, our experimental results on Arabic-to-English translation of long sentences from the news domain demonstrate that using the CCG-augmented glue grammar does not achieve any improvement over the soft constraint systems. Furthermore, soft constraint CCG-augmented HPB systems achieve better performance than the hard constraint systems. The experimental results also show that increasing the degree of the extended CCG labels under soft syntactic constraints improves performance, which leads the 3-category CCG-augmented HPB system which uses soft syntactic constraints to be the best-performing system among all the systems. However, increasing the degree of the extended CCG labels under the hard syntactic constraints does not show a consistent improvement.

Translation of a Combination of Long and Short Sentences

Table 5.8 demonstrates the experimental results for our CCG-augmented HPB systems which use extended syntactic constraints and the HPB baseline system on

System	BLEU	TER	METEOR
HPB	38.06	52.46	65.47
CCG_1	37.96	52.39*	**65.67***
CCG_2	38.39*	52.43	65.00
CCG_3	38.28	52.50	65.5
CCG_{pref1}	37.19	53.46	64.41
CCG_{pref2}	38.36	**52.32***	65.05*
CCG_{pref3}	38.39*	52.33	64.87
CCG_{glue1}	37.93	52.88	64.98
CCG_{glue2}	**38.64***	52.44	64.65
CCG_{glue3}	38.35	52.43*	65.44*

Table 5.8: Experimental results for our CCG-augmented HPB systems which use extended syntactic constraints and the HPB baseline system on the Arabic–English long and short sentences from the news data.

the Arabic–English long and short sentences from the news data. For the CCG-augmented HPB systems which use hard syntactic constraints, increasing the degree of the extended CCG labels from one to two improves performance by 0.43 absolute BLEU points, which corresponds to a 1.13% relative improvement. The 2-category CCG-augmented HPB system outperforms the HPB baseline system by 0.33 absolute BLEU points, which corresponds to a 0.87% relative improvement. The paired bootstrap resampling test demonstrates that this improvement is not statistically significant at p-level=0.05. Increasing the degree of the extended CCG labels from two to three causes a slight decrease of 0.11 absolute BLEU points.

Table 5.8 demonstrates that using soft syntactic constraints causes the performance of the 1-category CCG-augmented HPB system to degrade by 0.77 absolute BLEU points, which corresponds to a 2% relative decrease. The 2-category CCG-augmented HPB system which uses soft syntactic constraints achieves almost the same performance as the corresponding system which uses hard syntactic constraints with a slight difference of 0.03 absolute BLEU points. Using soft syntactic constraints with the CCG-augmented HPB system of degree three improves performance slightly by 0.11 absolute BLEU points, which corresponds to a 0.29% relative improvement.

Table 5.8 also shows that using the CCG-augmented glue grammar with extended CCG labels of degrees one and two increases performance over the corresponding soft constraint systems by 0.74 and 0.28 absolute BLEU points, which corresponds to relative improvements of 2% and 0.73%, respectively. The paired bootstrap resampling test demonstrates that these improvements are not statistically significant at p-level=0.05. The table also shows that using the CCG-augmented glue grammar with the extended CCG labels of degree three causes a slight decrease of 0.04 absolute BLEU points in comparison with the corresponding soft constraint system, which corresponds to a 0.1% relative decrease.

In general, our experimental results for Arabic-to-English translation on a combination of the short and long sentences in the news domain demonstrate that neither using soft syntactic constraints nor combining it with the CCG-augmented glue grammar helps to improve performance over the hard constraint systems except for the 2-category CCG-augmented HPB system which uses the CCG-augmented glue grammar. The 2-category CCG-augmented HPB system which uses the CCG-augmented glue grammar achieves the best performance among all the examined systems, outperforming the HPB baseline system by 0.58 absolute BLEU points, which corresponds to a 1.5% relative improvement. The paired bootstrap resampling test demonstrates that this improvement is not statistically significant at p-level=0.05. Moreover, the experimental results demonstrate that increasing the degree of the extended CCG labels for the soft constraint systems demonstrates a consistent improvement. However, increasing the degree of the extended CCG labels for the strong constraint systems which use a syntax-free glue grammar and the systems which use the CCG-augmented glue grammar does not show a consistent improvement .

Coverage of Syntactic Constraints

We try to explore the effect of our extension approaches on the coverage the syntactic constraints applied by our CCG-augmented HPB systems during translation. We

	Degree 1	Degree 2	Degree 3
CCG			
Glue	116	227	140
Hier	179	18	0
Total	295	245	140
CCG_{pref}			
Glue	71	179	64
Hier	126	47	280
Total	197	226	344
CCG_{glue}			
Glue	2	54	53
Hier	266	52	136
Total	268	106	189
$CCG_{glue(hard)}$			
Glue	111	92	127
Hier	35	32	32
Total	146	124	159

Table 5.9: Number of incomplete trees in the translation output produced by our CCG-augmented HPB systems which use extended syntactic constraints of different degrees for the short sentence test set. Hier denotes the number of incomplete trees because of a hierarchical rule whereas Glue denotes the number of incomplete trees because of a glue grammar rule.

accomplish this by measuring the number of sentences in the translation output which are produced by derivation trees that do not form complete syntactic trees. We define the complete syntactic tree to be the derivation tree which has a single CCG category at its root. In the case of the CCG-augmented HPB systems which use soft syntactic constraints, the complete syntactic tree has at least one single-category CCG label among the root labels. Tables 5.9 and 5.10 show the number of incomplete trees produced by our CCG-augmented HPB systems from different degrees for the short and long test sets, respectively. The tables illustrate the number of incomplete trees in two categories according to the reason which prevented the building of a complete tree. The first category contains the trees which are incomplete because of a hierarchical rule which violated all the syntactic constraints imposed by the model. This category also includes the trees which have at its root a hierarchical rule that does not have any single-category CCG label on its left-hand side. The second category contains the trees which are incomplete because of a glue grammar rule.

For the CCG-augmented HPB systems which use a syntax-free glue grammar, this category contains the trees which use at least a glue grammar rule to concatenate two partial translations. In the case of the CCG-augmented HPB systems which use the CCG-augmented glue grammar, the second category contains the trees which are incomplete because of a glue grammar rule which has an empty left-hand side, or because of a glue grammar rule at the root which does not have any single-category CCG label on its left-hand side.

Table 5.9 shows that increasing the degree of the extended CCG labels used by the CCG-augmented HPB systems which use hard syntactic constraints with the syntax-free glue grammar helps to significantly decrease the number of incomplete trees in the translation output. We can see that this decrease is mainly due to the significant decrease in the number of incomplete trees because of a hierarchical rule. Furthermore, the BLEU scores of the CCG-augmented HPB systems which use hard syntactic constraints show improvements which correlate with this improvement in the coverage of the syntactic constraints (cf. Table 5.6). For the CCG-augmented HPB systems which use soft syntactic constraints, increasing the degree of the extended CCG labels does not help to decrease the number of incomplete trees. This is because the coverage of the syntactic constraints in the soft constraint systems is mainly affected by the learnt weight of the preference grammar feature p_{syn}, which controls how strict the syntactic constraints are applied. This also applies to the CCG-augmented HPB systems which use the CCG-augmented glue grammar. It is worth noting that the CCG-augmented HPB systems which use the CCG-augmented glue grammar demonstrate improvements in BLEU score which correlate with improvements achieved in coverage (cf. Table 5.6). For the systems which use the CCG-augmented glue grammar with the hard syntactic constraints $CCG_{glue(hard)}$, the table shows that increasing the degree of the extended CCG labels from one to two decreases the number of incomplete trees. However, increasing the degree of the extended CCG labels from two to three has an opposite effect. In fact, all the incomplete trees because of a hierarchical rule for these systems produce an empty

	Degree 1	Degree 2	Degree 3
CCG			
Glue	33	277	482
Hier	406	159	18
Total	439	436	500
CCG_{pref}			
Glue	453	463	481
Hier	5	30	0
Total	458	493	481
CCG_{glue}			
Glue	106	85	102
Hier	259	235	232
Total	365	320	334
$CCG_{glue(hard)}$			
Glue	89	83	82
Hier	276	227	248
Total	365	310	330

Table 5.10: Number of incomplete trees in the translation output produced by our CCG-augmented HPB systems which use extended syntactic constraints of different degrees for the long sentence test set.

translation (i.e. they do not cover all the source sentence). The table also shows that using the CCG-augmented glue grammar under hard syntactic constraints produces fewer incomplete trees than under the soft syntactic constraints except for extended CCG labels of degree two.

Table 5.10 demonstrates that the coverage of the syntactic constraints for long sentence translation is more limited than in the case of short sentence translation. This is clearly because longer sentences are more difficult to translate than shorter sentences, requiring more syntactic constraints to be satisfied. Furthermore, some long sentences exceed the 20-word maximum application span we put for hierarchical rules, above which it is necessary to use glue grammar rules to perform translation. This is one of the reasons why the number of incomplete trees in the case of the systems which use the syntax-free glue grammar is larger than in the case of the systems which use the CCG-augmented glue grammar. Table 5.10 shows that increasing the degree of the extended CCG labels helps to improve the coverage of hierarchical rules for the systems which use hard syntactic constraints. However,

Source: و أشار الى أن هذا قد يساعد في رفع أصوات أوروبا و آسيا
Reference: *he pointed out that this would help in raising voices in europe and asia*
CCG_3**:** he pointed out that such ever helping raising votes in europe , asia
CCG_{pref3}**:** he pointed out that such had would help in raising votes in europe , asia
CCG_{glue3}**:** he pointed out that this would help in raising votes in europe , asia

Figure 5.6: Translations produced by our 3-category CCG-augmented HPB systems for the Arabic sentence و أشار الى أن هذا قد يساعد في رفع أصوات أوروبا و آسيا .

this does not help to achieve an overall better coverage. The CCG-augmented HPB systems which use soft syntactic constraints does not show a better coverage either. Using the CCG-augmented glue grammar helps to significantly decrease the number of incomplete trees in comparison with the systems which use the syntax-free glue grammar. However, this increase in coverage is not corroborated by an improvement in BLEU score (cf. Table 5.7). The table also shows that similar to short sentences, the systems which use the CCG-augmented glue grammar with the hard syntactic constraints achieve the best coverage when using extended CCG labels of degree two. Furthermore, all the incomplete trees because of a hierarchical rule for these systems produce an empty translation. The table also demonstrates that using the CCG-augmented glue grammar under hard syntactic constraints achieves similar coverage in comparison with the systems which use the CCG-augmented glue grammar with soft syntactic constraints.

Figure 5.6 demonstrates the translations produced by our 3-category CCG-augmented HPB systems for the Arabic sentence و أشار الى أن هذا قد يساعد في رفع أصوات أوروبا و آسيا. We can see that the translations produced by the CCG-augmented HPB systems which use the syntax-free glue grammar under hard and soft syntactic constraints are ungrammatical. Figures 5.7 and 5.8 demonstrate

the derivation trees of the translations produced by the CCG-augmented HPB systems which use hard and soft syntactic constraints, respectively. We can see that the use of non-syntactically aware glue grammar is the main reason for producing ungrammatical translations. By contrast, the CCG-augmented HPB system which uses the CCG-augmented glue grammar succeeds in building a full derivation tree which produces a grammatically correct translation. It is worth noting that the systems which use soft syntactic constraints produce a set of syntactic labels for the left-hand side of each partial translation, but for the seek of clarity, we only illustrate the labels which participate in building the most probable syntactic tree interpreting the translation output.

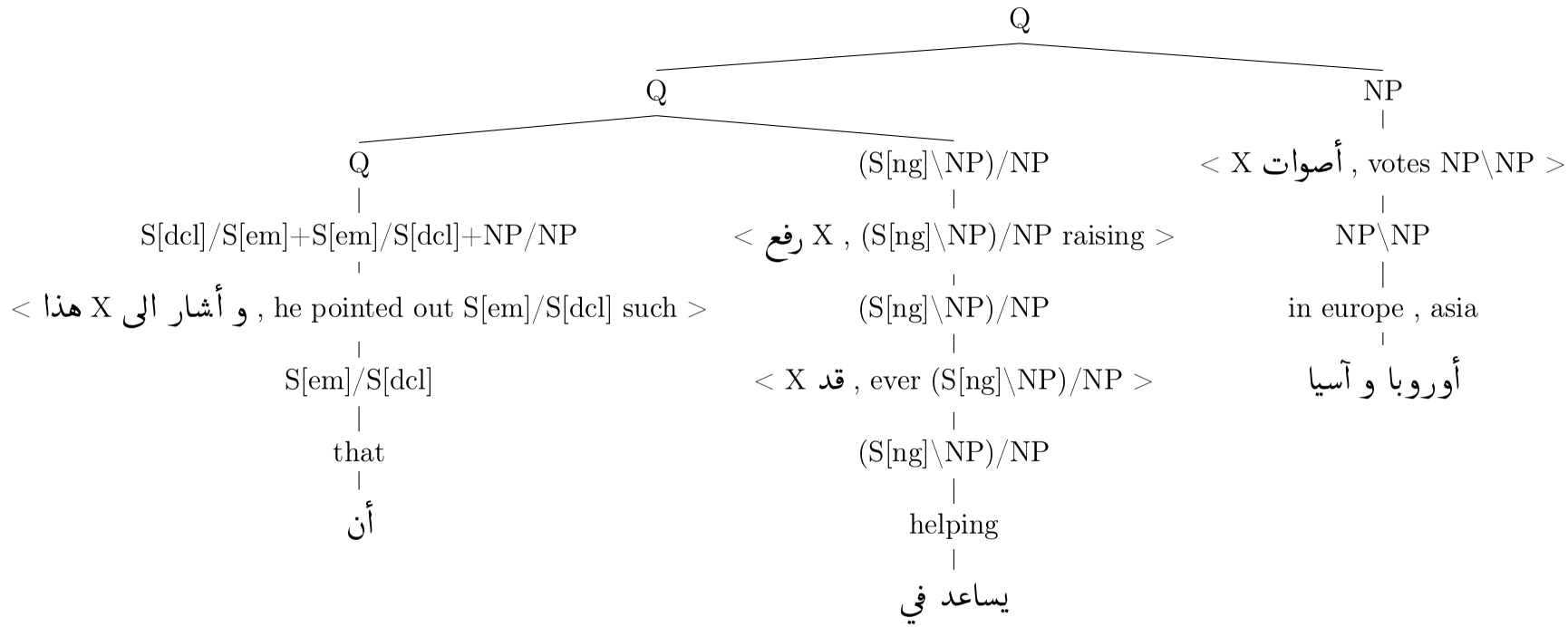

Figure 5.7: The derivation tree of the English translation produced by the 3-category CCG-augmented HPB system which uses hard syntactic constraints for the Arabic sentence و أشار الى أن هذا قد يساعد في رفع أصوات أوروبا و آسيا.

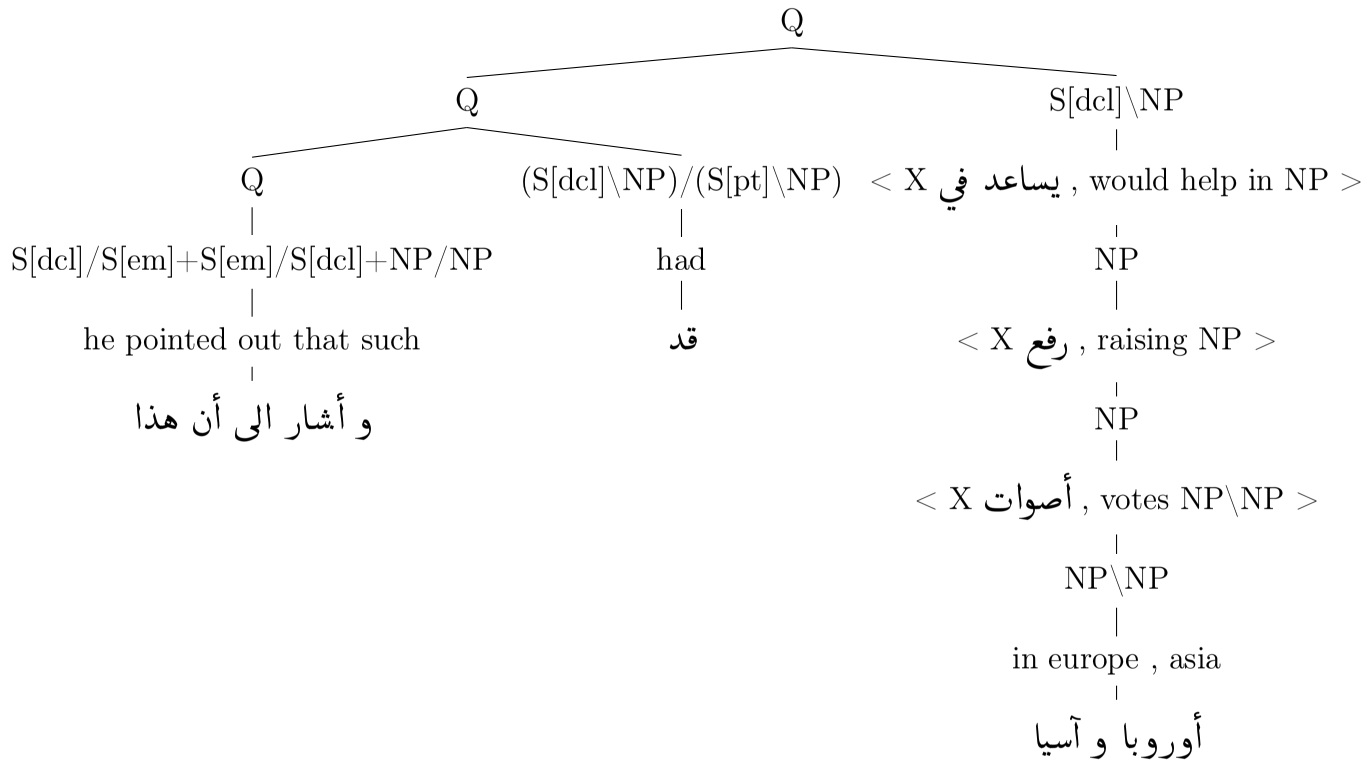

Figure 5.8: The derivation tree of the English translation produced by the 3-category CCG-augmented HPB system which uses soft syntactic constraints for the Arabic sentence و أشار الى أن هذا قد يساعد في رفع أصوات أوروبا و آسيا.

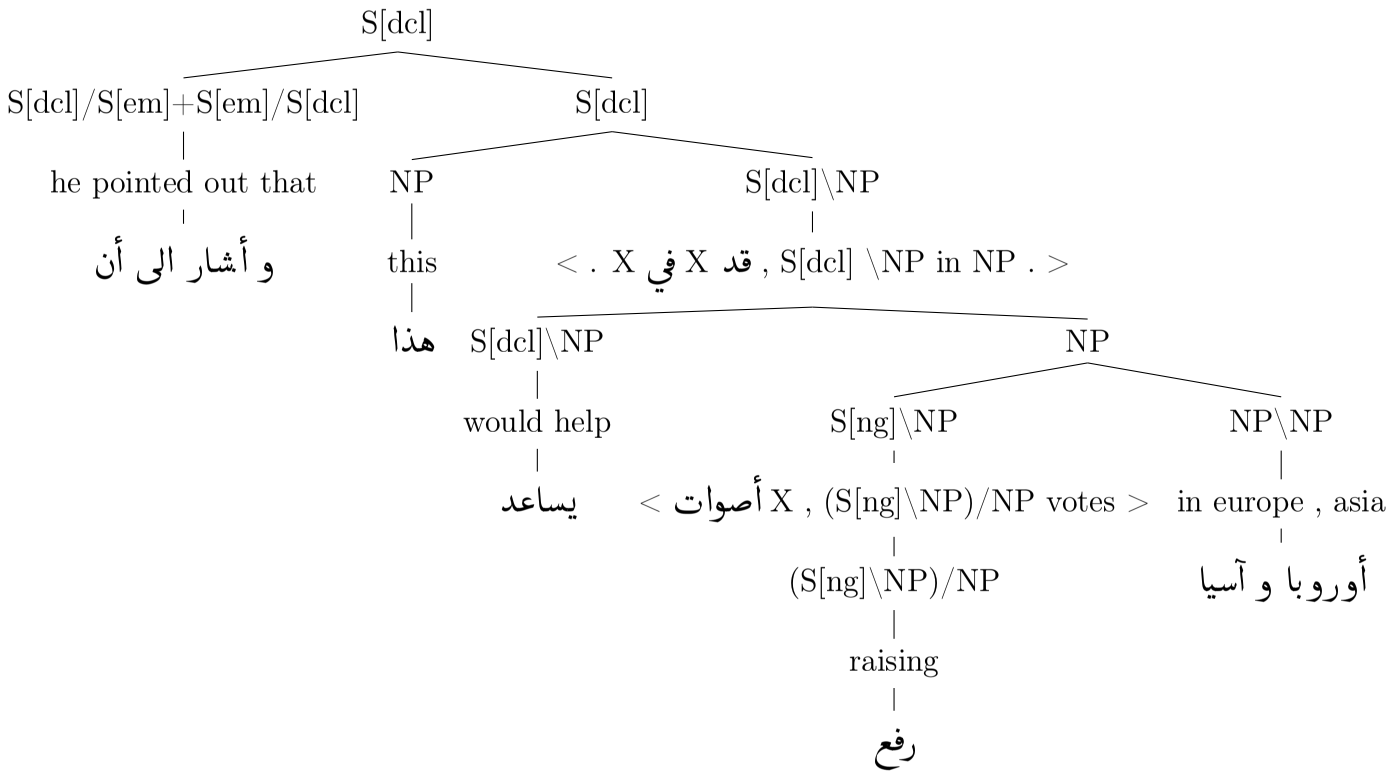

Figure 5.9: The derivation tree of the English translation produced by the 3-category CCG-augmented HPB system which uses the CCG-augmented glue grammar for the Arabic sentence و أشار الى أن هذا قد يساعد في رفع أصوات أوروبا و آسيا.

Figure 5.9 shows the derivation tree of the English translation produced by the 3-category CCG-augmented HPB system which uses the CCG-augmented glue grammar for the same Arabic sentence. We can see how the CCG-augmented hierarchical rules collaborate with the CCG-augmented glue grammar rules to build a complete tree which produces a grammatical translation. It is worth noting that our approach to measuring the coverage of the syntactic constraints is not totally precise. We can easily find many translations which are grammatically correct in spite of being produced by the syntax-free glue grammar rules. Moreover, some derivation trees, though complete, produce ungrammatical translations.

5.5.4 Analysis

Our experimental results demonstrated that using extended CCG-based syntactic constraints helps to significantly improve the performance of our CCG-augmented HPB systems over the HPB baseline system. However, the effect on performance of the different extension approaches varied according to the domain of the data between the IWSLT data and the news data. Moreover, the extension approaches showed different performance trends on the translation of short sentences, long sentences and the combination of long and short sentences.

For the IWSLT data, the CCG-augmented HPB systems which use extended CCG labels with soft syntactic constraints significantly outperformed the CCG-augmented HPB systems which use hard syntactic constraints on both Arabic-to-English and Chinese-to-English translation. The CCG-augmented glue grammar helped to improve the performance of most of the CCG-augmented HPB systems from different degrees on the Chinese–English IWSLT data, leading to the best performance among all the examined systems. However, for the Arabic–English IWSLT data, the CCG-augmented glue grammar was not able to improve the performance of most of the systems.

For the translation of short sentences from the news data, the use of extended CCG labels with soft syntactic constraints did not in general improve performance

over the systems which use hard syntactic constraints. Moreover, the CCG-augmented glue grammar helped to improve performance over the CCG-augmented HPB systems which use soft syntactic constraints. For the translation of long sentences from the news data, using extended CCG labels with soft syntactic constraints helped to improve performance over the systems which use hard syntactic constraints. However, using the CCG-augmented glue grammar did not further improve the performance of the CCG-augmented HPB systems which use soft syntactic constraints. When combining the short and long sentences from the news data, neither using soft syntactic constraints nor combining it with the CCG-augmented glue grammar improved performance over the systems which use hard syntactic constraints, except for the 2-category CCG-augmented HPB system which uses the CCG-augmented glue grammar.

The experiments also demonstrated that increasing the degree of the extended CCG labels helped to significantly increase label coverage (cf. Tables 5.4 and 5.5). Moreover, increasing the degree of the extended CCG labels above one improved performance on the Arabic-to-English news translation in general and helped to outperform the HPB baseline system (cf. Tables 5.6, 5.7 and 5.8). However, increasing the degree of the extended CCG labels did not show a consistent improvement.

One factor which might cause inconsistencies in the improvements obtained using our extension approaches on different data sets, between soft and hard syntactic constraints, using test sets of different sentence lengths, and using different degrees for the extended CCG labels, is the instability of the optimizer, namely MERT (Och, 2003) in our experiments. MERT uses a stochastic approach to optimize model parameters, which makes it prone to noisy parameter estimates (Clark et al., 2011). Clark et al. (2011) study the effect of optimizer instability on the experimental results. They demonstrate that optimizer instability can significantly affect translation quality. They warn that if optimizer instability is not controlled, it might lead to draw incorrect conclusions about the effectiveness of a specific MT approach. While we did not do this in this thesis, it is clearly an avenue to investigate further

in future work.

If we want to explain why the soft syntactic constraint systems achieved better performance than the hard syntactic constraint systems for long sentences whereas the opposite is true for short sentences, then we should bear in mind that long sentences are in general harder to translate, requiring more syntactic constraints to be satisfied than shorter sentences. This makes tree building for long sentences more prone to errors than for short sentences. Hard syntactic constraints are not fault-tolerant, which means that any violation of the syntactic constraints imposed by the model will break the tree and necessitate the use of glue grammar rules to join the partial translations. Hierarchical rules play a more essential role in HPB SMT than glue grammar rules, because glue grammar rules do not perform any reordering or impose any syntactic constraint. Thus using glue grammar rules rather than hierarchical rules would negatively affect performance. By contrast, soft syntactic constraints are more fault-tolerant, as they allow violations of the syntactic constraints imposed by the model. Thus, the use of hierarchical rules is continued even after such violations, what probably helped our soft constraint systems to achieve better performance for long sentence translation.

Trying to understand why our CCG-augmented glue grammar did not help to significantly improve the coverage of the syntactic constraints or performance over the systems which use the syntax-free glue grammar, we must take into consideration the following factors which cause errors or even failure in tree building. Firstly, parsing errors committed during annotation of training data cause some words/phrases to be wrongly tagged. Secondly, the training data does not contain all the contexts within which the word/phrase occurs. This might lead to the wrong CCG label of the word/phrase which does not match its appropriate context to be used during translation. The two previous reasons might also prevent complete trees from being built even for grammatical translations. Thirdly, tree building in our CCG-augmented HPB systems which use the CCG-augmented glue grammar is performed through a collaboration between the CCG-annotated hierarchical rules

and CCG parsing performed by the CCG-augmented glue grammar. However, this process of tree building lacks an important stage for CCG parsing, namely supertagging. The accuracy of supertagging is essential for CCG parsing accuracy (Clark and Curran, 2004). Accurately assigning supertags to the words of a sentence should take into consideration the context within which the word occurs, such as the surrounding words and their POS tags (Clark, 2002b). Phrases and words in our approach are assigned CCG-based labels along with their co-occurrence probabilities, which are learnt from the annotated training data. Thus, the most frequent label for a word/phrase is more likely to be used in the tree building process regardless of the context within which it occurs, which might lead to errors in tree building during translation. Finally, a problem which affects the accuracy of CCG parsing in general and thus affects our CCG-based tree building is the over-generation problem. One cause of over-generation in CCG parsing is the use of a set of rules called type-changing rules. Type-changing rules are unary rules used to deal with the category proliferation problem (Hockenmaier and Steedman, 2002). The use of these rules when the training data is parsed prior to rule extraction in our CCG-augmented HPB systems transmits the over-generation problem to our CCG-augmented chart decoding, which might generate ungrammatical translations.

Figure 5.10 shows the derivation of the English translation produced by the 3-category CCG-augmented HPB system which uses CCG-augmented glue grammar for the Arabic sentence قال الوزير أن هذه المحطات تم اجراء التجارب عليها وقد دخلت التشغيل بالفعل . Although the tree is complete, having the *S[dcl]* category at the root, the translation it produces is not grammatical. We can see that there is a type-changing rule at the root of the partial translation *conducting experiments for it joined the operation*, which changes the category at the root of the subtree

covering this partial translation from *S[dcl]* to *NP\NP*. This enables this subtree to join the subtree covering the previous partial translation *the minister said that these stations*. Finally, the resulting subtree joins the phrase *was re-conducted*, to form a complete tree. We can see from this example how type-changing rules led to the production of an ungrammatical translation despite being produced by a complete tree.

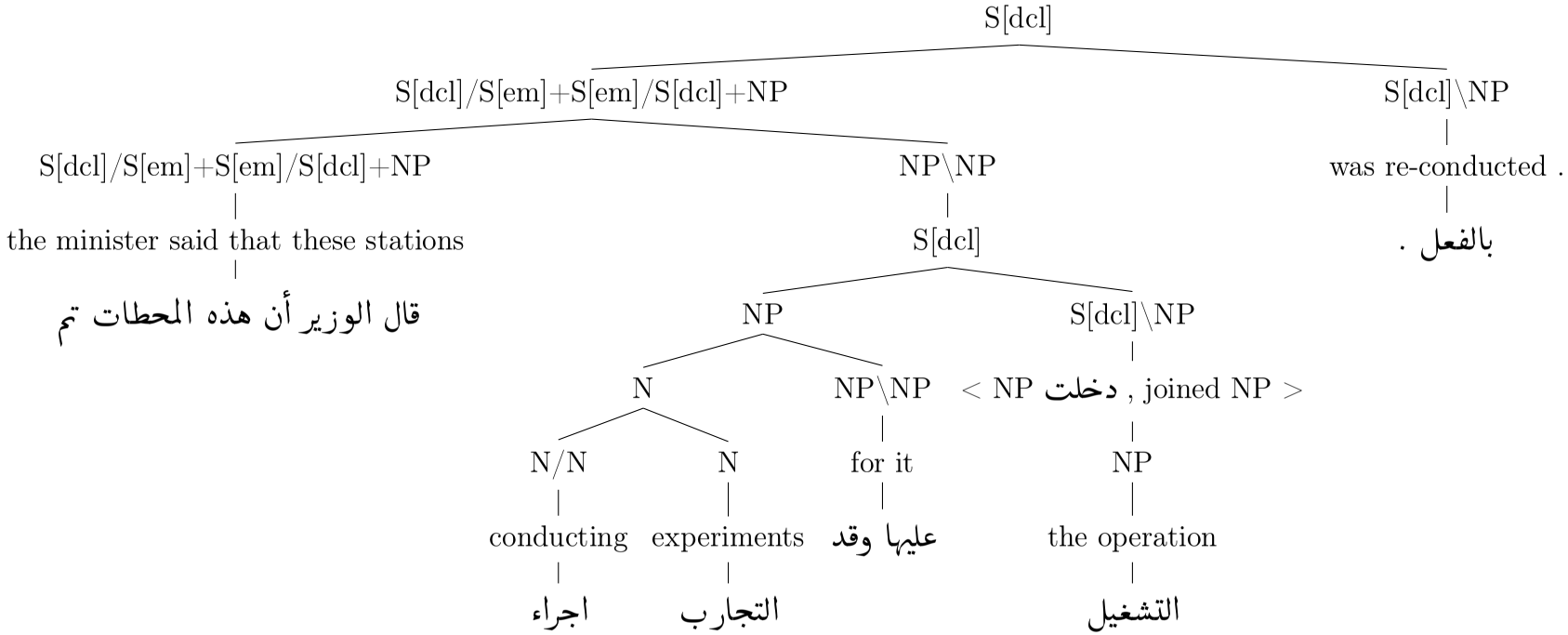

Figure 5.10: The derivation tree of the English translation produced by the 3-category CCG-augmented HPB system which uses CCG-augmented glue grammar for the Arabic sentence . قال الوزير أن هذه المحطات تم اجراء التجارب عليها وقد دخلت التشغيل بالفعل.

5.6 Conclusions

In this chapter, we tried to answer our third research question (**RQ3**), by addressing the factors which limit the coverage of the syntactic constraints applied in our CCG-augmented HPB system, which are the syntactically unannotated phrases and the syntax-free glue grammar. We argued that increasing the coverage of the syntactic constraints in our CCG-augmented HPB system helps to improve performance. Therefore, we tried to extend the coverage of the syntactic constraints in our CCG-augmented HPB system following two approaches. The first approach provides coverage for a larger proportion of the extracted phrases in the HPB SMT model. This is accomplished by extending the definition of the nonterminal label to include more than one CCG category. The second approach tries to provide a full syntax augmentation for the HPB SMT grammar by augmenting glue grammar rules with CCG combinatory rules.

We presented experiments examining the application of these two approaches both individually and combined on Arabic-to-English and Chinese-to-English translation in the travel speech expressions domain, and on Arabic-to-English translation in the news domain. We examined the performance of our extension approaches under both hard and soft syntactic constraints and on short and long sentences. Our experimental results showed that extending the syntactic constraints in our CCG-augmented HPB system helped to significantly improve its performance over the HPB baseline system, which answers our fourth research question (**RQ4**). Our different approaches to extending the CCG-based syntactic constraints demonstrated different performance trends on different data sets, between soft and hard syntactic constraints and between short and long sentences. We provided an in-depth analysis of our experimental results, in which we tried to explain these different trends and discuss problems facing our approaches.

Chapter 6

Conclusions

In this thesis, we presented approaches to incorporating syntactic information extracted using CCG into the HPB SMT model. Motivated by the previous research which shed light on the advantages which CCG provides for SMT over other grammar formalisms (Hassan et al., 2007, 2009), we used CCG to apply rich, precise and wide-coverage syntactic constraints in the HPB SMT model. We also tried to solve common problems affecting the performance of previous syntax-augmented HPB SMT systems, namely the strictness, sparsity and the limited coverage of the syntactic constraints, taking advantage of the richness and flexibility of CCG structures.

Following previous research on incorporating syntax in the HPB SMT model by annotating phrases and nonterminals with CF-PSG-based syntactic labels (Zollmann and Venugopal, 2006), we used CCG categories to attach syntactic labels to nonterminals and phrases in the HPB SMT model. We believe that CCG categories are richer, more accurate and more able to express the syntactic function of non-standard constituents such as SMT phrases than CF-PSG-based labels. This motivated us to argue that CCG is better than CF-PSG when they are used to label nonterminals and phrases in the HPB SMT model, which was the subject of our first research question:

RQ1: *Is CCG better than CF-PSG when using it to annotate nonterminals and*

phrases in the HPB SMT model?

To verify our argument, we carried out experiments which compared the performance of our CCG category HPB system with the SAMT system on Arabic-to-English and Chinese-to-English translation on data from the news and speech expressions domains. Our experimental results demonstrated that our CCG category HPB system outperformed the SAMT system in most of the conducted experiments. We also conducted a comparison between our CCG category HPB system and the SAMT system in terms of label coverage, translation model size, and the sparsity of the labels. The results of the comparison demonstrated that the CCG category labels have better coverage, are less sparse and lead to the production of smaller translation models than the SAMT labels. We also conducted an in-depth manual analysis comparing the translation output of our CCG-augmented HPB model with the SAMT and HPB baseline systems. Our manual analysis demonstrated that our CCG-augmented HPB system is better than the SAMT at inserting verbs. The analysis also showed that our CCG-augmented HPB system is better than both the SAMT and HPB system at capturing the translation of words/phrases missed by other systems. However, our analysis demonstrated one major weakness of our CCG-augmented HPB system, namely the failure to correctly translate some words/phrases. Our experimental results in addition to our manual analysis clearly demonstrated that CCG is better than CF-PSG when using it to annotate nonterminals and phrases in the HPB SMT model, which answers our first research question (**RQ1**).

Although our experimental results demonstrated that our CCG-augmented HPB system was able to outperform the SAMT system, our CCG-augmented HPB system underperformed compared to the HPB baseline system in most of the experiments. We believe that the sparsity of our CCG category labels which restricted the decoding search space is what caused its performance to deteriorate in comparison with the HPB baseline system. As a solution, we tried to see whether softening the syntactic constraints imposed by CCG category labels by making them more coarse-

grained would improve the performance of our CCG-augmented HPB system, which led us to our second research question:

RQ2: *Does softening the syntactic constraints imposed by CCG category nonterminal labels by simplifying them help to improve performance?*

We proposed two approaches to extract less-fine grained CCG-based labels from CCG categories. The first approach extracts CCG contextual information from CCG categories and employs them in the nonterminal label representation. The second approach removes syntactic features held by some CCG categories from the nonterminal label representation. Although these approaches reduce the richness of CCG categories, the simplified CCG labels still reflect important syntactic information about the syntactic function of the phrase and its context. We conducted experiments which explore the trade-off between the richness of the syntactic constraints and their flexibility under different factors: the language pair, the size and domain of the training data, and the size of the language model. Our experimental results demonstrated that our label simplification approaches helped to significantly improve performance over the CCG category HPB system. Furthermore, our experiments emphasised the effects of the language pair, the domain and size of the training data and the size of the language model on the performance of the different label simplification approaches.

Although we demonstrated that CCG categories have better coverage than SAMT labels, both our CCG-augmented HPB sysem and the SAMT system failed to label at least 30% of the total extracted phrases in the HPB SMT model. We believe that the limited coverage of the syntactic constraints in our CCG-augmented HPB model is a reason for the production of ungrammatical translations. Therefore, we wanted to explore the factors which limit the coverage of the syntactic constraints in our CCG-augmented HPB model in order to tackle them. This gave rise to our third research question:

RQ3: *Why is the coverage of the syntactic constraints limited in our CCG-augmented HPB system ?*

The fact that a large proportion of phrases and nonterminals in our CCG-augmented HPB model is syntactically unannotated is one reason for the limited coverage of the syntactic constraints in our CCG-augmented HPB system. Another important reason which limits the coverage of the syntactic constraints applied during translation is that part of the HPB SMT grammar, namely the glue grammar, is non-syntactically aware. We demonstrated that glue grammar rules constitute a large proportion of the derivations produced by our CCG-augmented HPB and the SAMT systems. We believe that the syntax-free glue grammar is one important reason why ungrammatical translations are produced. After identifying the factors which limit the coverage of the syntactic constraints in our CCG-augmented HPB model, we attempted to address these factors by extending the coverage of the syntactic constraints applied in our CCG-augmented HPB system in the aim of improving its performance. This led to our fourth research question:

RQ4: *Does extending the syntactic constraints in our CCG-augmented HPB system help to improve performance?*

We presented two approaches to increase the coverage of the syntactic constraints in our CCG-augmented HPB system. The first approach addresses the limited coverage of the nonterminal labels by extending the notion of the nonterminal label to include more than one CCG category. The second approach augments glue grammar rules with CCG combinatory rules, which extends our CCG-based syntax augmentation to include the whole HPB SMT grammar. We tried to apply our extension approaches using soft syntactic constraints using the Preference Grammars paradigm (Venugopal et al., 2009) so that performance is not hindered by the strictness of syntactic constraints. We carried out experiments which examined the effect on performance of applying these extension approaches both individually and combined on Arabic-to-English and Chinese-to-English translation in the speech expressions domain. We also conducted experiments which examine our extension approaches on Arabic-to-English translation on data from the news domain with different sentence lengths. Our experiments demonstrated that our extension ap-

proaches helped to significantly increase the coverage of the syntactic constraints in our CCG-augmented HPB system. Our experimental results also demonstrated that our extension approaches enabled our CCG-augmented HPB system to significantly outperform the HPB baseline system. Furthermore, our experiments showed that our different extension approaches exhibited different performance trends under the various factors examined, namely the strictness of the syntactic constraints, the sentence length, and the domain of the data. We conducted a detailed analysis of our experimental results, discussing the different performance trends of our extension approaches and shedding light on the problems that affected the performance of our CCG-augmented HPB systems which used extended syntactic constraints.

6.1 Future Work

The current CCG-augmented HPB SMT model can be improved in many ways, which opens up many avenues for future work. In our current model, only a single extended CCG category label is extracted from the categories of two extended CCG labels combined during glue grammar rule application. In fact, a sequence of CCG categories can be combined in many ways, resulting in a set of different CCG categories. Furthermore, models which score category combination similar to the ones used in CCG parsing (Clark and Curran, 2007) can be incorporated into scoring different extended CCG labels which result from combining CCG categories during glue grammar rule application. This helps to model the possible syntactic functions of a partial translation more accurately, which in turn helps to build more correct derivation trees.

Throughout our research, we demonstrated the shortcomings of using BLEU to evaluate the improvements on grammaticality of translation achieved by our CCG-augmented HPB systems. Furthermore, the insufficient sensitivity of the BLEU score to the grammaticality of the translation output might mislead feature weight optimization (Och et al., 2004). Thus, we believe that using a CCG-based or

a syntax-based evaluation metric (such as the one proposed by Owczarzak et al. (2007)) instead of BLEU in tuning and testing our CCG-augmented HPB systems would better estimate the grammaticality of the translation output and give insights into the improvements achieved by our CCG-augmented HPB systems. A simple CCG-based evaluation metric can be built on sequences of CCG supertags similar to the language model. More complex CCG-based evaluation metrics which use CCG dependencies can be also explored. Another interesting strand of research could also explore the correlation between the BLEU score and good parse trees produced for the translation output.

When measuring the coverage of the syntactic constraints in our CCG-augmented HPB model which uses the CCG-augmented glue grammar, we used a simple metric which counts the trees with a single CCG category at its root. A more comprehensive metric which more precisely estimates the coverage of the syntactic constraints beyond this simple metric can be investigated.

A problem we encountered when evaluating our CCG-augmented HPB systems is the inconsistency of improvements obtained under the different factors examined. We noted that this might be due to the instability of the optimizer. Thus, in order to account for optimizer instability and draw more reliable conclusions about the effectiveness of our different CCG-augmentation approaches for HPB SMT, the optimization and test set evaluation should be replicated at least three times for each system according to the recommendation of Clark et al. (2011).

Finally, the CCG-based syntactic information incorporated into the CCG-augmented derivation trees produced by our CCG-augmented HPB system can be employed in syntax-based postprocessing. The postprocessing stage can exploit the richness of CCG categories, which reflect valency and directionality, to manipulate the derivation trees based on their CCG annotation in the aim of correcting grammatical mistakes and improving translation quality.

Bibliography

Ajdukiewicz, K. (1935). Die syntaktische konnexität. *Studia Philosophica*, 1:1–27.

Almaghout, H., Jiang, J., and Way, A. (2010a). CCG augmented hierarchical phrase-based machine translation. In *Proceedings of the 7th International Workshop on Spoken Language Translation*, pages 211–218, Paris, France.

Almaghout, H., Jiang, J., and Way, A. (2010b). The DCU machine translation systems for IWSLT 2010. In *Proceedings of the 7th International Workshop on Spoken Language Translation*, pages 37–44, Paris, France.

Almaghout, H., Jiang, J., and Way, A. (2011). CCG contextual labels in hierarchical phrase-based SMT. In *Proceedings of the 15th conference of the European Association for Machine Translation*, pages 281–288, Leuven, Belgium.

Almaghout, H., Jiang, J., and Way, A. (2012). Extending CCG-based syntactic constraints in hierarchical phrase-based SMT. In *Proceedings of the 16th conference of the European Association for Machine Translation*, pages 28–30, Trento, Italy.

Auli, M., Lopez, A., Hoang, H., and Koehn, P. (2009). A systematic analysis of translation model search spaces. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, StatMT '09, pages 224–232, Stroudsburg, PA, USA. Association for Computational Linguistics.

Baker, K., Bloodgood, M., Callison-Burch, C., Dorr, B. J., Filardo, N. W., Levin, L., Miller, S., and Piatko, C. (2010). Semantically-informed machine translation:

A tree-grafting approach. In *Proceedings of The Ninth Biennial Conference of the Association for Machine Translation in the Americas*, Denver, Colorado.

Banerjee, P., Almaghout, H., Naskar, S., Roturier, J., Jiang, J., Way, A., and van Genabith, J. (2011). The DCU machine translation systems for IWSLT 2011. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 41–48, San Francisco.

Bangalore, S. and Joshi, A. (1999). Supertagging: An approach to almost parsing. *Computational Linguistics*, 25(2):237–265.

Bar-Hillel, Y. (1953). A quasi-arithmetical notation for syntactic description. *Language*, 29:47–58. Reprinted in Y. Bar-Hillel. (1964). *Language and Information: Selected Essays on their Theory and Application*, Addison-Wesley 1964, 61–74.

Birch, A., Osborne, M., and Koehn., P. (2007). CCG supertags in factored statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 9–16, Prague, Czech Republic.

Boxwell, S. A. and Brew, C. (2010). A Pilot Arabic CCGbank. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Brown, P., Cocke, J., Pietra, S. D., Pietra, V. D., Jelinek, F., Mercer, R., and Roossin, P. (1988). A statistical approach to language translation. In *Proceedings of the 12th conference on Computational linguistics - Volume 1*, COLING '88, pages 71–76, Stroudsburg, PA, USA. Association for Computational Linguistics.

Brown, P. F., Cocke, J., Pietra, S. A. D., Pietra, V. J. D., Jelinek, F., Lafferty, J. D., Mercer, R. L., and Roossin, P. S. (1990). A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85.

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., and Mercer, R. L. (1993). The math-

ematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19(2):263–311.

Cai, S., Lu, Y., and Liu, Q. (2009). Improved reordering rules for hierarchical phrase-based translation. In *Proceedings of the 2009 International Conference on Asian Language Processing*, IALP '09, pages 65–70, Washington, DC, USA. IEEE Computer Society.

Carpuat, M. and Wu, D. (2007). Improving statistical machine translation using word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 61–72, Prague, Czech Republic.

Chappelier, J.-C. and Rajman, M. (1998). A generalized CYK algorithm for parsing stochastic CFG. In *Proc. of 1st Workshop on Tabulation in Parsing and Deduction (TAPD'98)*, pages 133–137, Paris (France).

Charniak, E. (2001). Immediate-head parsing for language models. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL '01, pages 124–131, Toulouse, France. Association for Computational Linguistics.

Charniak, E., Knight, K., and Yamada, K. (2003). Syntax-based language models for statistical machine translation. In *MT Summit IX*, pages 40–46, New Orleans, USA.

Chiang, D. (2005). A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual meeting of the Association for Computational Linguistics*, ACL '05, pages 263–270, Ann Arbor, MI.

Chiang, D. (2007). Hierarchical phrase-based translation. *Comput. Linguist.*, 33(2):201–228.

Chiang, D. (2010). Learning to translate with source and target syntax. In *Proceed-*

ings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL '10, pages 1443–1452, Uppsala, Sweden.

Chomsky, N. (1957). *Syntactic Structures*. Mouton, The Hague.

Clark, J. H., Dyer, C., Lavie, A., and Smith, N. A. (2011). Better hypothesis testing for statistical machine translation: controlling for optimizer instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2*, HLT '11, pages 176–181, Stroudsburg, PA, USA. Association for Computational Linguistics.

Clark, S. (2002a). A Supertagger for Combinatory Categorial Grammar. In *Proceedings of the TAG+ Workshop*, pages 19–24, Venice.

Clark, S. (2002b). A supertagger for Combinatory Categorial Grammar. In *Proceedings of the TAG+ Workshop*, page 19–24, Venice, Italy.

Clark, S. and Curran, J. (2004). The Importance of Supertagging for Wide-Coverage CCG Parsing. In *Proceedings of COLING-04*, pages 282–288, Geneva, Switzerland.

Clark, S. and Curran, J. R. (2007). Wide-Coverage Efficient Statistical Parsing with CCG and Log-Linear Models. *Computational Linguistics*, 33(4):493–552.

Curran, J. R. and Clark, S. (2003). Investigating GIS and smoothing for maximum entropy taggers. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics - Volume 1*, EACL '03, pages 91–98, Budapest, Hungary. Association for Computational Linguistics.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal Statistical Society, Series B*, 39(1):1–38.

Earley, J. (1970). An efficient context-free parsing algorithm. *Commun. ACM*, 13:94–102.

Eisele, A. and Chen, Y. (2010). Multiun: A multilingual corpus from united nation documents. In Chair), N. C. C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., and Tapias, D., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Eisner, J. (2003). Learning non-isomorphic tree mappings for machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 2*, ACL '03, pages 205–208, Sapporo, Japan. Association for Computational Linguistics.

Galley, M., Graehl, J., Knight, K., Marcu, D., DeNeefe, S., Wang, W., and Thayer, I. (2006). Scalable inference and training of context-rich syntactic translation models. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 961–968, Sydney, Australia. Association for Computational Linguistics.

Galley, M., Hopkins, M., Knight, K., and Marcu, D. (2004). What's in a translation rule? In Susan Dumais, D. M. and Roukos, S., editors, *HLT-NAACL 2004: Main Proceedings*, pages 273–280, Boston, Massachusetts, USA. Association for Computational Linguistics.

Gao, Q. and Vogel, S. (2011). Utilizing target-side semantic role labels to assist hierarchical phrase-based machine translation. In *Proceedings of Fifth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 107–115, Portland, Oregon, USA. Association for Computational Linguistics.

Goutte, C., Cancedda, N., Dymetman, M., and Foster, G., editors (2009). *Learning Machine Translation*. Neural Information Processing. MIT Press, Cambridge.

Habash, N., Rambow, O., and Roth, R. (2009). MADA+TOKAN: A Toolkit for Arabic Tokenization, Diacritization, Morphological Disambiguation, POS Tagging,

Stemming and Lemmatization. In *Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR)*, pages 242–245, Cairo, Egypt.

Hanneman, G. and Lavie, A. (2011). Automatic category label coarsening for syntax-based machine translation. In *Proceedings of the Fifth Workshop on Syntax, Semantics and Structure in Statistical Translation*, SSST-5, pages 98–106, Portland, Oregon. Association for Computational Linguistics.

Haque, R., Kumar Naskar, S., van den Bosch, A., and Way, A. (2010). Supertags as source language context in hierarchical phrase-based SMT. In *Proceedings of AMTA 2010: The Ninth Conference of the Association for Machine Translation in the Americas*, pages 210–219, Denver, CO, USA.

Haque, R., Naskar, S., Ma, Y., and Way, A. (2009). Using supertags as source language context in SMT. In *Proceedings of the 13th Annual Meeting of the European Association for Machine Translation (EAMT 2009)*, pages 234–241, Barcelona, Spain.

Hassan, H. (2009). *Lexical syntax for statistical machine translation*. PhD thesis, Dublin City University.

Hassan, H., Sima'an, K., , and Way, A. (2007). Supertagged phrase-based statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, ACL '07, pages 288–295, Prague, Czech.

Hassan, H., Sima'an, K., and Way, A. (2009). A syntactified direct translation model with linear-time decoding. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, EMNLP '09, pages 1182–1191, Singapore.

He, Y. and Way, A. (2010). Metric and reference factors in minimum error rate training. *Machine Translation*, 24(1):27–38.

Hoang, H., Koehn, P., and Lopez, A. (2009a). A Unified Framework for Phrase-Based, Hierarchical, and Syntax-Based Statistical Machine Translation. In *Proc. of the International Workshop on Spoken Language Translation*, pages 152–159, Tokyo, Japan.

Hoang, H., Koehn, P., and Lopez, A. (2009b). A Unified Framework for Phrase-Based, Hierarchical, and Syntax-Based Statistical Machine Translation. In *Proceedings of the International Workshop on Spoken Language Translation*, pages 152–159, Tokyo, Japan.

Hockenmaier, J. (2003). *Data and Models for Statistical Parsing with Combinatory Categorial Grammar*. PhD thesis, University of Edinburgh.

Hockenmaier, J. (2006). Creating a CCGbank and a wide-coverage CCG lexicon for German. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, ACL-44, pages 505–512, Stroudsburg, PA, USA. Association for Computational Linguistics.

Hockenmaier, J. and Steedman, M. (2002). Acquiring compact lexicalized grammars from a cleaner treebank. In *Proceedings of Third International Conference on Language Resources and Evaluation*, page 1974–1981, Las Palmas, Spain.

Hockenmaier, J. and Steedman, M. (2007). CCGbank: A Corpus of CCG Derivations and Dependency Structures Extracted from the Penn Treebank. *Computational Linguistics*, 33:355–396.

Huang, L., Knight, K., and Joshi, A. (2006). A syntax-directed translator with extended domain of locality. In *Proceedings of the Workshop on Computationally Hard Problems and Joint Inference in Speech and Language Processing*, CHSLP '06, pages 1–8, New York City, New York. Association for Computational Linguistics.

Huang, Z., Čmejrek, M., and Zhou, B. (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*, EMNLP '10, pages 138–147, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ittycheriah, A. and Roukos, S. (2007). Direct translation model 2. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 57–64, Rochester, NY.

Jelinek, F. (1997). *Statistical methods for speech recognition*. MIT Press, Cambridge, MA, USA.

Johnson, M. (1998). Pcfg models of linguistic tree representations. *Comput. Linguist.*, 24(4):613–632.

Johnson, M., Geman, S., Canon, S., Chi, Z., and Riezler, S. (1999). Estimators for stochastic "unification-based" grammars. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 535–541, College Park, Maryland. Association for Computational Linguistics.

Joshi, A. and Schabes, Y. (1991). *Tree-adjoining Grammars and Lexicalized Grammars*. Technical report. University of Pennsylvania, School of Engineering and Applied Science, Department of Computer and Information Science.

Joshi, A. K. (1985). *How Much Context-Sensitivity is Necessary for Characterizing Structural Descriptions — Tree Adjoining Grammars*, pages 206–250. Cambridge University Press.

Kay, M. (1980). *Algorithm schemata and data structures in syntactic processing*, pages 35–70.

Klein, D. and Manning, C. D. (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 423–430, Sapporo, Japan. Association for Computational Linguistics.

Kneser, R. and Ney, H. (1995). Improved backing-off for m-gram language modeling. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 181–184, Detroit, Michigan, USA.

Knight, K. (1999). Decoding complexity in word-replacement translation models. *Comput. Linguist.*, 25:607–615.

Knight, K. and Graehl, J. (2005). An Overview of Probabilistic Tree Transducers for Natural Language Processing. In *Sixth International Conference on Intelligent Text Processing and Computational Linguistics (CICLing'2005)*, pages 1–25. Springer-Verlag.

Koehn, P. (2004a). Pharaoh: a beam search decoder for phrase-based statistical machine translation models. In *Machine translation: from real users to research: 6th conference of the Association for Machine Translation in the Americas, AMTA 2004*, pages 115–124, Washington, DC.

Koehn, P. (2004b). Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference of Empirical Methods in Natural Language Processing (EMNLP-2004)*, pages 388–395, Barcelona, Spain.

Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the tenth Machine Translation Summit*, pages 79–86, Phuket, Thailand. AAMT, AAMT.

Koehn, P. (January 2010). *Statistical Machine Translation*. Cambridge University Press.

Koehn, P., Axelrod, A., Mayne, A. B., Osborne, C. C.-B. M., and Talbot, D. (2005). Edinburgh system description for the 2005 IWSLT speech translation evaluation. In *International Workshop on Spoken Language Translation: Evaluation Campaign on Spoken Language Translation [IWSLT 2005]*, page 8pp, Pittsburgh, PA, USA.

Koehn, P. and Hoang, H. (2007). Factored translation models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL-2007)*, pages 868–876, Prague, Czech Republic.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (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 (ACL-2007)*, pages 177–180, Prague, Czech Republic.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In *HLT-NAACL 2003: conference combining Human Language Technology conference series and the North American Chapter of the Association for Computational Linguistics conference series*, pages 48–54, Edmonton, Canada.

Lavie, A. and Abhaya, A. (2007). Meteor: an automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 228–231, Prague, Czech Republic. Association for Computational Linguistics.

Lewis, II, P. M. and Stearns, R. E. (1968). Syntax-directed transduction. *J. ACM*, 15(3):465–488.

Li, J., Tu, Z., Zhou, G., and van Genabith, J. (2012). Using syntactic head information in hierarchical phrase-based translation. In *Proceedings of the Seventh*

Workshop on Statistical Machine Translation, pages 232–242, Montréal, Canada. Association for Computational Linguistics.

Li, Z., Callison-Burch, C., Dyer, C., Ganitkevitch, J., Khudanpur, S., Schwartz, L., Thornton, W. N. G., Weese, J., and Zaidan, O. F. (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, Stroudsburg, PA, USA. Association for Computational Linguistics.

Liu, Y., Huang, Y., Liu, Q., and Lin, S. (2007). Forest-to-string statistical translation rules. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 704–711, Prague, Czech Republic. Association for Computational Linguistics.

Liu, Y., Liu, Q., and Lin, S. (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*, ACL-44, pages 609–616, Sydney, Australia. Association for Computational Linguistics.

Liu, Y., Lü, Y., and Liu, Q. (2009). Improving tree-to-tree translation with packed forests. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 558–566, Suntec, Singapore. Association for Computational Linguistics.

Lopez, A. (2008). Tera-scale translation models via pattern matching. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 505–512, Manchester, United Kingdom. Association for Computational Linguistics.

MacQueen, J. B. (1967). Some Methods for Classification and Analysis of Multi-Variate Observations. 1.

Marcu, D., Wang, W., Echihabi, A., and Knight, K. (2006). SPMT: statistical machine translation with syntactified target language phrases. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 44–52, Sydney, Australia. Association for Computational Linguistics.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of English: the penn treebank. *Computational Linguistics*, 19:313–330.

Marton, Y. and Resnik, P. (2008). Soft syntactic constraints for hierarchical phrase-based translation. In *Proceedings of the 2008 Annual Meeting of the Association for Computational Linguistics (ACL-HLT)*, pages 1003–1011.

Matsuzaki, T., Miyao, Y., and Tsujii, J. (2005). Probabilistic CFG with latent annotations. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 75–82, Ann Arbor, Michigan. Association for Computational Linguistics.

Mehay, D. N. and Brew, C. H. (2012). CCG Syntactic Reordering Models for Phrase-based Machine Translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 210–221, Montréal, Canada. Association for Computational Linguistics.

Mylonakis, M. and Sima'an, K. (2011). Learning hierarchical translation structure with linguistic annotations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 642–652, Portland, Oregon. Association for Computational Linguistics.

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, pages 160–167, Sapporo Convention Center, Japan.

Och, F. J., Gildea, D., Khudanpur, S., Sarkar, A., Yamada, K., Fraser, A., Kumar, S., Shen, L., Smith, D., Eng, K., Jain, V., Jin, Z., and Radev, D. (2004). A smorgasbord of features for statistical machine translation. In Susan Dumais, D. M. and Roukos, S., editors, *HLT-NAACL 2004: Main Proceedings*, pages 161–168, Boston, Massachusetts, USA. Association for Computational Linguistics.

Och, F. J. and Ney, H. (2002). Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 295–302, Stroudsburg, PA, USA. Association for Computational Linguistics.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29:19–51.

Och, F. J. and Ney, H. (2004). The alignment template approach to statistical machine translation. *Comput. Linguist.*, 30:417–449.

Owczarzak, K., van Genabith, J., and Way, A. (2007). Labelled dependencies in machine translation evaluation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 104–111, Stroudsburg, PA, USA. Association for Computational Linguistics.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, (ACL 2002)*, pages 311–318, Philadelphia, Pennsylvania.

Peter, J.-T., Huck, M., Ney, H., and Stein, D. (2011). Soft string-to-dependency hierarchical machine translation. In *International Workshop on Spoken Language Translation*, pages 246–253, San Francisco, California, USA.

Petrov, S., Barrett, L., Thibaux, R., and Klein, D. (2006). Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of*

the Association for Computational Linguistics, ACL-44, pages 433–440, Sydney, Australia. Association for Computational Linguistics.

Petrov, S. and Klein, D. (2007). Improved inference for unlexicalized parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 404–411, Rochester, New York. Association for Computational Linguistics.

Pollard, C. and Sag, I. (1994). *Head-Driven Phrase Structure Grammar*. Studies in Contemporary Linguistics. University of Chicago Press.

Quirk, C., Menezes, A., and Cherry, C. (2005). Dependency treelet translation: syntactically informed phrasal smt. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 271–279, Ann Arbor, Michigan. Association for Computational Linguistics.

Ratnaparkhi, A. (1996). A maximum entropy part-of-speech tagger. In *Proceedings of the Empirical Methods in Natural Language Processing Conference*, University of Pennsylvania.

Sadat, F. and Habash, N. (2006). Combination of Arabic preprocessing schemes for statistical machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 1–8, Sydney, Australia.

Schwartz, L., Callison-Burch, C., Schuler, W., and Wu, S. (2011). Incremental syntactic language models for phrase-based translation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 620–631, Portland, Oregon, USA. Association for Computational Linguistics.

Shen, L., Xu, J., and Weischedel, R. (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. Association for Computational Linguistics.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In *Proceedings of Association for Machine Translation in the Americas (AMTA-2006)*, pages 223–231, Cambridge, MA, USA.

Steedman, M. (2000). *The syntactic process*. MIT Press, Cambridge, MA, USA.

Stein, D., Peitz, S., Vilar, D., and Ney, H. (2010). A cocktail of deep syntactic features for hierarchical machine translation. In *Conference of the Association for Machine Translation in the Americas 2010*, number 9, page 9.

Stein, D., Vilar, D., Peitz, S., Freitag, M., Huck, M., and Ney, H. (2011). A guide to jane, an open source hierarchical translation toolkit. *Prague Bull. Math. Linguistics*, 95:5–18.

Stolcke, A. (2002). Srilm—an extensible language modeling toolkit. In *ICSLP 2002, Interspeech 2002: 7th International Conference on Spoken Language Processing*, pages 901–904, Denver, Colorado, USA.

Stroppa, N., van den Bosch, A., and Way, A. (2007). Exploiting Source Similarity for SMT using Context-Informed Features. In *Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI'07)*, page 231–240, Skövde, Sweden.

Tillmann, C. (2004). A unigram orientation model for statistical machine translation. In *Proceedings of HLT-NAACL 2004: Short Papers*, HLT-NAACL-Short '04, pages 101–104, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tse, D. and Curran, J. R. (2010). Chinese CCGbank: extracting CCG derivations from the Penn Chinese Treebank. In *Proceedings of the 23rd International Confer-*

ence on Computational Linguistics, COLING '10, pages 1083–1091, Stroudsburg, PA, USA. Association for Computational Linguistics.

Venugopal, A., Zollmann, A., N.A.Smith, and Vogel, S. (2009). Preference grammars: softening syntactic constraints to improve statistical machine translation. In *Proceedings of Human Language Technologies: the 2009 annual conference of the North American Chapter of the ACL*, pages 37–45, Boulder, Colorado.

Vilar, D., Stein, D., and Ney, H. (2008). Analysing soft syntax features and heuristics for hierarchical phrase based machine translation. In *International Workshop on Spoken Language Translation*, pages 190–197, Honolulu, Hawaii.

Vilar, D., Stein, D., Peitz, S., and Ney, H. (2010). If i only had a parser: Poor man's syntax for hierarchical machine translation. In *International Workshop on Spoken Language Translation*, pages 345–352, Paris, France.

Wang, W., May, J., Knight, K., and Marcu, D. (2010). Re-structuring, re-labeling, and re-aligning for syntax-based machine translation. *Computational Linguistics*, 36:247–277.

Watanabe, T., Tsukada, H., and Isozaki, H. (2006). Left-to-right target generation for hierarchical phrase-based translation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, ACL-44, pages 777–784, Stroudsburg, PA, USA. Association for Computational Linguistics.

Weese, J., Callison-Burch, C., and Lopez, A. (2012). Using categorial grammar to label translation rules. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 222–231, Montréal, Canada. Association for Computational Linguistics.

Wu, D. (1997). Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Comput. Linguist.*, 23:377–403.

Yamada, K. and Knight, K. (2001). A syntax-based statistical translation model. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL '01, pages 523–530, Toulouse, France. Association for Computational Linguistics.

Yamada, K. and Knight, K. (2002). A decoder for syntax-based statistical mt. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 303–310, Philadelphia, Pennsylvania. Association for Computational Linguistics.

Zhang, H.-P., Yu, H.-K., Xiong, D.-Y., and Liu, Q. (2003). HHMM-based Chinese lexical analyzer ICTCLAS. In *Proceedings of the second SIGHAN workshop on Chinese language processing - Volume 17*, SIGHAN '03, pages 184–187, Stroudsburg, PA, USA. Association for Computational Linguistics.

Zhang, M., Jiang, H., Aw, A., Li, H., Tan, C. L., and Li, S. (2008). A tree sequence alignment-based tree-to-tree translation model. In *ACL-08: HLT. 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Proceedings of the conference*, pages 559–567, The Ohio State University, Columbus, Ohio, USA. The Association for Computer Linguistics.

Zollmann, A. and Venugopal, A. (2006). Syntax augmented machine translation via chart parsing. In *Proceedings of the Workshop on Statistical Machine Translation, HLT/NAACL*, pages 138–141, Stroudsburg, PA, USA. Association for Computational Linguistics.

Zollmann, A., Venugopal, A., Och, F., and Ponte, J. (2008). A systematic comparison of phrase-based, hierarchical and syntax-augmented statistical MT. In *Proceedings of the 22nd International Conference on Computational Linguistics*, pages 1145–1152, Manchester, UK.

Zollmann, A. and Vogel, S. (2010). New parameterizations and features for PSCFG-

based machine translation. In *Proceedings of the 4th Workshop on Syntax and Structure in Statistical Translation (SSST)*, Beijing, China.

Zollmann, A. and Vogel, S. (2011). A word-class approach to labeling pscfg rules for machine translation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1–11, Stroudsburg, PA, USA. Association for Computational Linguistics.