Integrating Source-Language Context into Log-Linear Models of Statistical Machine Translation

Rejwanul Haque

B-Tech, M-Tech

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

June 2011

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.: 58123385

Date:

Abstract

The translation features typically used in state-of-the-art statistical machine translation (SMT) model dependencies between the source and target phrases, but not among the phrases in the source language themselves. A swathe of research has demonstrated that integrating source context modelling directly into log-linear phrase-based SMT (PB-SMT) and hierarchical PB-SMT (HPB-SMT), and can positively influence the weighting and selection of target phrases, and thus improve translation quality. In this thesis we present novel approaches to incorporate source-language contextual modelling into the state-of-the-art SMT models in order to enhance the quality of lexical selection. We investigate the effectiveness of use of a range of contextual features, including lexical features of neighbouring words, part-of-speech tags, supertags, sentence-similarity features, dependency information, and semantic roles. We explored a series of language pairs featuring typologically different languages, and examined the scalability of our research to larger amounts of training data.

While our results are mixed across feature selections, language pairs, and learning curves, we observe that including contextual features of the source sentence in general produces improvements. The most significant improvements involve the integration of long-distance contextual features, such as dependency relations in combination with part-of-speech tags in Dutch-to-English subtitle translation, the combination of dependency parse and semantic role information in English-to-Dutch parliamentary debate translation, supertag features in English-to-Chinese translation, or combination of supertag and lexical features in English-to-Dutch subtitle translation. Furthermore, we investigate the applicability of our lexical contextual model in another closely related NLP problem, namely machine transliteration.

Acknowledgments

First of all, I would like to express my deepest gratitude to my supervisor Andy Way for his support, guidance, and encouragement throughout the research. He always inspires me to explore my research in various directions, offers ample suggestions and experiences, and keeps me on the right track. Under his guidance I have grown up both as a researcher and a professional person. Thanks to Andy.

I would also like to express my sincere gratitude to Antal van den Bosch for the fruitful and insightful discussions, with whom we have carried on constant collaborations. I am grateful to him for his suggestions, cooperation, support throughout my work, and providing computing infrastructure.

Special thanks to my advisor, Sudip Kumar Naskar, for examining my research topics and having intensive discussions. I also would like to thank my Master thesis supervisor, Sivaji Bandyopadhyay, for his cooperation and motivation to initiate work in the area of Natural Language Processing. I am grateful to Mikel L. Forcada and Markus Helfert for providing valuable comments on my transfer paper. I am thankful to Marta Ruiz and Rafael E. Banchs for a fruitful collaboration. I would like to thank both present and past members of the NCLT/CNGL for their support and interest in my work, including Yifan, Sergio, Jie, Jinhua, Yanjun, Josef, Ankit, Pratyush, Hala, Tsuyoshi, Sandipan, Joachim, Javed, Eithne, Ríona and Debasis.

I wish to acknowledge Science Foundation Ireland[1] for funding my research (under grant no. 07/CE/I1142) and the Irish Centre for High-End Computing[2] for the use of their resources.

Last but not the least, I would like to convey my thanks to my mother Aktar Khanam, my father Samsul Haque and all my family members for their moral supports throughout the course. Finally and most importantly, thanks to my soul mate, my wife Benojir, for abundant support throughout my work.

[1] http://www.sfi.ie
[2] http://www.ichec.ie

Contents

List of Figures

List of Tables

Acronyms

NLP	Natural Language Processing
MT	Machine Translation
SMT	Statistical Machine Translation
PB-SMT	Phrase-Based Statistical Machine Translation
HPB-SMT	Hierarchical Phrase-Based Statistical Machine Translation
RBMT	Rule-Based Machine Translation
EBMT	Example-Based Machine Translation
CBMT	Corpus-Based Machine Translation
WB-SMT	Word-Based Statistical Machine Translation
TM	Translation Model
LM	Language Model
MERT	Minimum Error-Rate Training
HMM	Hidden Markov Model
AER	Alignment Error Rate
SVM	Support Vector Machine
MaxEnt	Maximum Entropy
LDC	Linguistic Data Consortium
BTEC	Basic Travel Expression Corpus
CPH	Canadian Parliament Hansards
IWSLT	International Workshop on Spoken Language Translation
PSCFG	Probabilistic synchronous context-free grammar
POS	Part of Speech
NT	Non Terminal
VP	Verb Phrase
NP	Noun Phrase
ADJP	Adjectival Phrase
N	Noun

Europarl	European Parliamnetary
EILMT	English to Indian Language Machine Translation
WMT	Workshop on Statistical Machine Translation
MetricsMATR	Metrics for Machine Translation
NIST	National Institute of Standards and Technology
BLEU	BiLingual Evaluation Understudy
METEOR	Metric for Evaluation of Translation with Explicit ORdering
TER	Translation Edit Rate
WER	Word Error Rate
PER	Position-Independent Word Error Rate
WSD	Word-Sense Disambiguation
CI	Contextual Information
CCG	Combinatory Categorial Grammar
LTAG	Lexicalized Tree Adjoining Grammar
k-NN	k-Nearest Neighbour
TRIBL	Tree-based approximation of Instance-Based Learning
MBL	Memory-Based Learning
TiMBL	Tilburg Memory-Based Learner
MaTrEx	Machine Translation Using Examples
EM	Expectation Maximization
MBR	Minimum Bayes Risk
SRILM	Stanford Research Institute Language Modeling
CYK	Cocke Younger Kasami
IG	Information Gain
GR	Gain Ratio
IB	Instance-Based
MIRA	Margin Infused Relaxed Algorithm

FST	Finite State Transducer
DTM2	Direct Translation Model 2
UN	United Nations
FBIS	Foreign Broadcast Information Service

Notations

$P(.\|.)$	Conditional probability
e_1^I	Target sentence
f_1^I	Source sentence
$\hat{e}_k$	k^{th} target phrase
$\hat{f}_k$	k^{th} source phrase
e_i	i^{th} target word
f_j	j^{th} source word
s_1^K	Segmentation of the source and target sentences
s_k	k^{th} segmentation
h / ϕ	Feature function in log-linear SMT framework
λ	Feature weight
X	Nonterminal
α	Hiero source Phrase
γ	Hiero target Phrase
$\sim$	A one-to-one correspondence between nonterminals in source and target phrases
D(G)	A set of PSCFG derivations
e(d)	Target-side yield of derivation d
$\hat{h}_{\text{mbl}}$ / ϕ_{mbl}	Memory-based feature function
$\hat{h}_{\text{best}}$ / ϕ_{best}	Memory-based binary feature function

k	Distances in instance-based classifier
n	Switching point in the feature ordering from IGTree to normal k-NN classification
N	Total number of named entities
r_{ij}	j^{th} reference transliteration for ith name
c_{ik}	k^{th} candidate transliteration for ith name
$\mathrm{CI}_{\mathrm{lex}}$	Lexical contextual information
$\mathrm{CI}_{\mathrm{pos}}$	Contextual information with POS features
$\mathrm{CI}_{\mathrm{st}}$	Contextual information with supertag features
$\mathrm{CI}_{\mathrm{di}}$	Contextual information with dependency features
$\mathrm{CI}_{\mathrm{si}}$	Contextual information with semantic features

Chapter 1

Introduction

The current state-of-the-art Statistical Machine Translation (SMT) models (Koehn et al., 2003; Chiang, 2007) can be viewed as log-linear combinations of features (Och and Ney, 2002) that usually comprise translational features and language models. The translational features involved in these models typically express dependencies between the source and target phrases, but not dependencies between the phrases in the source language themselves, i.e. they do not take into account the contexts of those phrases.

Stroppa et al. (2007) observed that incorporating source-language context using neighbouring words and part-of-speech tags had the potential to improve translation quality. This has led to a whole tranche of research, of which we provide an overview in Chapter 2, which has shown that integrating source context modelling into SMT models can positively influence the weighting and selection of target phrases, and thus improve translation quality.

Approaches to include source-language context to help select more appropriate target phrases have partly been inspired by methods used in word-sense disambiguation (WSD), where rich contextual features are employed to determine the most likely sense of a polysemous word given that context. These contextual features may include lexical features, i.e. words appearing in the immediate context (Giménez and Màrquez, 2007; Stroppa et al., 2007), shallow and deep syntactic fea-

tures of the sentential context (Gimpel et al., 2008), full sentential context (Carpuat and Wu, 2007), and sentence-similarity features (Costa-Jussà and Banchs, 2010). Studies in which syntactic features are employed have made use of part-of-speech taggers (Stroppa et al., 2007), supertaggers (Haque et al., 2009a, 2010a), and shallow and deep syntactic parsers (Gimpel et al., 2008; Haque et al., 2009b).

In this thesis, we show that exploring local sentential context information in the form of both supertags (Haque et al., 2009a, 2010a) and syntactic dependencies (Haque et al., 2009b) can be integrated into a Phrase-Based SMT (PB-SMT) (Koehn et al., 2003) and a Hierarchical Phrase-Based SMT (HPB-SMT) (Chiang, 2007) models successfully to improve target phrase selection. In addition to supertags and dependency relations, we introduce semantic roles as a new contextual feature in the PB-SMT model. We compare the effectiveness of these rich and complex knowledge sources as source-language context features with that of the basic contextual features (i.e. words and POS tags) in state-of-the-art SMT models. Furthermore, we explore various similarity features (Haque et al., 2010b) by measuring the similarity between a source sentence to be translated with the source side of the parallel training sentences, the usefulness of which are examined by integrating them into a PB-SMT model, both individually and in collaboration with the supertag-based features.

We examine the scalability of our research to larger amounts of training data, and explore a range of language pairs featuring typologically different languages. Our results allow us to conclude that incorporating source-language contextual features benefits a range of different language pairs, both with English as source language (translating to Dutch, Chinese, Japanese, Hindi, Spanish, and Czech) and target language (from Dutch), on different types of data such as news articles and commentary, parliamentary debates, patents, and subtitles, according to a range of automatic evaluation measures.

Furthermore, we apply our context-informed PB-SMT model to a different NLP task, namely machine transliteration (Haque et al., 2009c). We showed that our

context-sensitive transliteration models significantly outperform the respective baseline models in terms of transliteration accuracy.

1.1 Research Questions

Target phrase selection, and for that matter, target sentence generation in SMT are mostly driven by the n-gram target language model (LM). Target phrase selection to generate a target sentence mostly depends on the n-gram matches in the LM. The generation of a target sentence with respect to an n-gram language model can thus be viewed as a measure of similarity between the sentence under generation and the sentences of the corpus on which the LM is built. In other words, the use of LM makes the resulting translation similar to previously seen sentences.

Translation of a source sentence in example-based machine translation (EBMT) (Nagao, 1984) can be seen as a process consisting of three consecutive steps: (i) retrieval, i.e. look for the source sentences in the bilingual corpus that are similar to the sentence to be translated, (ii) matching, i.e. find useful fragments in these sentences, and (iii) recombination, i.e. adapt the translations of these fragments (Nagao, 1984; Somers et al., 1994; Somers, 1999; Carl and Way, 2003). Consequently, EBMT crucially relies on the retrieval of source sentences from the bilingual corpus, which are similar to the input sentence to be translated; in other words, EBMT is *source-similarity-oriented*. On the other hand, the word sense disambiguation, a task intricately related to machine translation (MT), typically employs rich context-sensitive features to determine contextually the most likely sense of a polysemous word. Can we exploit the idea of EBMT's source-similarity in SMT by making use of source-language context which could play an important role in finding the most suitable translation of a polysemous SMT phrase? A variety of research papers have exploited source-language context in SMT in order to improve lexical selection (e.g. Carpuat and Wu, 2007; Giménez and Màrquez, 2007; Stroppa et al., 2007), most of which have focused on modelling neighbouring lexical and syntactic dependencies

that an SMT phrase has as source-side context. Interestingly, the translation of an ambiguous source phrase may depend on the context that could be either adjacent or distant to the phrase to be translated (cf. Section 3.2). Therefore, exploration of the nature of source-context on which the meaning of an SMT phrase depends may enhance the quality of lexical selection of the state-of-the-art SMT models (Koehn et al., 2003; Chiang, 2007). This leads to our first *research question (RQ1)*:

- *RQ1: What kind of information can be modelled as useful source-language context in the state-of-the-art SMT systems in order to improve target phrase selection?*

In this thesis, we present novel approaches to embed various lexical and syntactic contextual features in the state-of-the-art SMT models: (a) basic contextual features (neighbouring words and POS tags) as in Stroppa et al. (2007), (b) lexical syntactic descriptions in the form of supertags, (c) deep syntactic information (grammatical dependency relations) as in Carpuat and Wu (2007) and Max et al. (2008),[1] and (d) semantic roles. While supertags capture long-distance dependencies in an indirect way, dependency relations encode them directly; by following a dependency relation one can, for example, obtain the lexical identity of the related word directly. We go a step further by introducing semantic dependencies that an SMT phrase possesses as contextual feature. This rich and complex syntactic and semantic information, which helps capture long-distance word-to-word dependencies in a sentence, may be more useful source-side contextual features to improve lexical selection in SMT. Furthermore, following (Costa-Jussà and Banchs, 2010) we explore various sentence similarity-based source-language contextual features which are measured on the basis of similarity between the input source sentence and the source-side of the training sentences. In a set of pilot experiments we examine the effect of incorporating the sentence similarity-based contextual features, both individually and in collaboration with the supertag-based features.

[1]We detail the differences between our approach and those of (Carpuat and Wu, 2007; Max et al., 2008) with regard to various aspects in Section 6.2.1.1 (cf. page 162).

One important aspect is how to include rich and complex syntactic and semantic context information in the phrase and rule tables. This invokes our second *research question (RQ2)*:

- *RQ2: How can the rich and complex syntactic and semantic contexts be incorporated into state-of-the-art SMT models?*

In order to incorporate these rich and complex knowledge sources into the SMT models, we have made use of supertaggers (Bangalore and Joshi, 1999; Clark and Curran, 2004), dependency parsers (Nivre et al., 2006; Van den Bosch et al., 2007), and semantic role labelers (Johansson and Nugues, 2008), most of which are readily available only for the English language. This forced us to choose English as the source language of the translation direction for most of our translation tasks. We detail the approaches we adopted to incorporate complex contextual information into the SMT models later (cf. Sections 3.3, 5.2, 6.2) in this thesis. Context-dependent phrase translation can naturally be seen as a multi-class classification problem. In other words, a source phrase with associated contextual information has to be classified to obtain a reduced but fine-grained set of possible target phrases. Like Stroppa et al. (2007), we use a memory-based classification framework (Daelemans and van den Bosch, 2005) that enables the estimation of these features while avoiding sparseness problems.

We tried to explore another related NLP problem – transliteration – in which we could share the advantages of our context-informed SMT models. Our third *research question (RQ3)* is:

- *RQ3: Can we deploy our context-sensitive SMT model in the related application of transliteration?*

Our context-informed SMT model was employed to perform transliteration on a standard English-to-Hindi data set. We show how the transliteration model can also benefit from the inclusion of source-side contextual features into the model.

Consistency and scalability are two important interrelated issues which one should investigate in a proper manner before reaching a conclusion. This raises our fourth *research question (RQ4)*:

- *RQ4: How does the performance of various context-informed SMT models vary with increasing sizes of training data sets?*

We investigate the effect of various contextual features in the SMT models by systematically varying the amounts of training data for several language pairs. A large set of experiments were carried out with increasing sizes of training data sets exploring several contextual features. The experimental results obtained with varied training data sizes yield informative learning curves which help us to draw a solid conclusion.

Each experiments carried out in this thesis, as well as the usual automatic evaluation, we provide an in-depth analysis performed on the translations produced by our context-sensitive SMT models and the respective baseline models.

1.2 Thesis Structure

The remainder of this thesis is organized as follows:

- Chapter 2 presents an overview of MT, followed by a discussion on foundation of SMT and significant research contributions on the state-of-the-art SMT models. This chapter also reassesses previous work on incorporating context into SMT.

- Chapter 3 introduces an overview of the state-of-the-art in SMT (Koehn et al., 2003; Chiang, 2007), followed by definitions of our context-informed SMT models. Then, we present an overview of memory-based classification algorithms used in this thesis, implementation aspects of our context-based models, data set statistics, and an overview of the MT evaluation techniques used to validate our experimental findings.

- Chapter 4 presents a series of experimental results obtained by integrating basic contextual features (words and POS tags) into the two state-of-the-art SMT models. Since we employ our word-contextual model in the application of transliteration, this chapter also reports outcomes of the context-sensitive transliteration systems.

- Chapter 5 introduces supertags as source-language contextual features and reports a series of experimental results obtained by incorporating supertag context into the state-of-the-art SMT models. Since our main intuition is to employ the supertag contextual features with the sentence-level similarity features, this chapter also introduces sentence-similarity contextual modelling and reports experimental results obtained by adding various sentence-similarity features into the PB-SMT model, both individually and in collaboration with the supertag-based contextual features.

- Chapter 6 introduces deep syntactic and semantic contextual features and reports the experimental results obtained by incorporating those features into the state-of-the-art SMT models.

- Chapter 7 concludes and discusses avenues for future work.

1.3 Publications

Parts of the research presented in this thesis have been published in peer-reviewed international conferences. Work which has resulted in publications includes:

- (Haque et al., 2009a) entitled "Using Supertags as Source Language Context in SMT" was published in proceedings of the 13th Annual Meeting of the European Association for Machine Translation (EAMT-09).

- (Haque et al., 2009b) entitled "Dependency Relations as Source Context in Phrase-Based SMT" was published in proceedings of PACLIC 23: the 23rd Pacific Asia Conference on Language, Information and Computation.

- (Haque et al., 2009c) entitled "English–Hindi Transliteration Using Context-Informed PB-SMT: the DCU System for NEWS 2009" was published in proceedings of Named Entities Workshop (NEWS 2009), ACL-IJCNLP 2009.

- (Haque et al., 2010a) entitled "Supertags as Source Language Context in Hierarchical Phrase-Based SMT" was published in proceedings of AMTA 2010: The Ninth Conference of the Association for Machine Translation in the Americas.

- (Okita et al., 2010) entitled "MaTrEx: the DCU MT System for NTCIR-8" was published in proceedings of NTCIR-8.

- (Penkale et al., 2010) entitled "MaTrEx: The DCU MT System for WMT 2010" was published in proceedings of the Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR (WMT-MetricsMATR 2010), ACL 2010.

- (Haque et al., 2010b) entitled "Sentence Similarity-Based Source Context Modelling in PBSMT" was published in proceedings of the International Conference on Asian Language Processing 2010, (IALP 2010).

- Haque et al. (2011) entitled "Integrating Source-Language Context into Phrase-Based Statistical Machine Translation" has been accepted for publication in the *Machine Translation* journal.

Chapter 2

Related Works

2.1 Overview of SMT Research Work

In this section, we give an introductory overview of statistical machine translation (SMT). First, we present a general overview of machine translation (MT), followed by word-based SMT (WB-SMT), and then we move from the word-based to phrase-based SMT (PB-SMT) model. Then, we look at a few significant research contributions in the PB-SMT model as well as in the hierarchical phrase-based SMT (HPB-SMT) model. Furthermore, we extend the discussion with a special focus on discriminative training in SMT.

In the subsequent section, we discuss the research work related to this thesis, which makes use of the contextual knowledge sources from the either side of translation pair to improve lexical selection in the SMT model.

2.1.1 General Overview of Machine Translation

Machine Translation is a computer application that translates a text from one natural language to another. Linguistic analysis classifies the MT task into three alternative processes according to the Vauquois pyramid (Vauquois and Boitet, 1985): *direct*, *transfer*, and *interlingua*. The Vauquois pyramid framework comprises two more steps: source-language *analysis* and target-language *generation*.

With regard to the translation approaches, MT can be broadly classified into two categories: rule-based machine translation (RBMT) and corpus-based machine translation (CBMT). RBMT relies on hand-built linguistic rules and bilingual dictionaries for each language pair. On the other hand, CBMT uses large amounts of bilingual and target monolingual corpus to acquire the translation knowledge required. Corpus-driven models of MT can further be divided into example based machine translation (EBMT) (Nagao, 1984) and statistical machine translation (SMT) (Brown et al., 1988). In EBMT, translation is performed by analogy; in contrast, statistical approaches are based on models created probabilistically from a bilingual corpus. Statistical models are by far the most dominant and popular empirical approaches in today's MT technology.

With regard to the translation unit and decoding method, SMT models can further be divided into three groups: word-based, phrase-based and syntax-based SMT. The phrase-based SMT (PB-SMT) model of Koehn et al. (2003) and hierarchical phrase-based SMT (HPB-SMT) model of Chiang (2007) are the current state-of-the-art in MT technology. The work illustrated in this thesis have been carried out with the above two state-of-the-art SMT models.

2.1.2 Beginning of Statistical Machine Translation

More than six decades ago, Weaver (1949) expressed the idea of applying statistical methods to translate a word by taking its contexts into account. However, researchers abandoned this approach due to the complexity involving implementation at that time.

Four decades after the proposal of Weaver (1949), Brown et al. (1988, 1990) expressed the MT process with a probabilistic model, namely, the *noisy channel* model of translation. The noisy channel model of translation (Brown et al., 1988, 1990) maximizes the probability of generating a sentence $e_1^I = e_1, ..., e_I$ in the target language from a given sentence $f_1^J = f_1, ..., f_J$ in the source language, namely $\mathrm{P}(e_1^I|f_1^J)$. Therefore, according to the noisy channel model, the translation task can

be viewed as a process that maximizes the probability $\mathrm{P}(e_1^I|f_1^J)$, as in (2.1):

$$\underset{I,e_1^I}{\operatorname{argmax}}\,\mathrm{P}(e_1^I|f_1^J) = \underset{I,e_1^I}{\operatorname{argmax}}\,\mathrm{P}(f_1^J|e_1^I).P(e_1^I) \tag{2.1}$$

where $\mathrm{P}(f_1^J|e_1^I)$ and $\mathrm{P}(e_1^I)$ denote respectively the translation model and the target language model. The translation model $\mathrm{P}(f_1^J|e_1^I)$ is estimated using a set of models: e.g. translation, fertility, and distortion probabilities (Brown et al., 1988, 1990). This model of MT is popularly known as word-based SMT (WB-SMT) model, since the translation unit of this model is word.

The translation process of the noisy channel model is essentially a search problem which Brown et al. (1988, 1990) facilitated with a variant of the *stack search* (Bahl et al., 1983). The translation models (translation, fertility, distortion probabilities) (Brown et al., 1988, 1990) are estimated applying the expectation maximization (EM) algorithm (Dempster et al., 1977) on the bilingual training sentences. More detail regarding the mathematical modelling of parameter estimation can be found in (Brown et al., 1993). The parameter estimation algorithms described in (Brown et al., 1993) are popularly known as the IBM models of word alignment, which comprise five different models (IBM model 1 to model 5). Word alignment is an advanced research topic in SMT. In the next section, we give a brief overview of the IBM models and discuss a few recent research papers on word alignment.

2.1.3 Word Alignment

The IBM models are the most popular and dominant approaches to the problem of word alignment. The IBM models learn from *incomplete data*. In other words, these models adopt an *unsupervised learning* method, namely the EM algorithm (Dempster et al., 1977). The EM algorithm works with sentences-level parallel corpora. The translation model in Equation (2.1) can be rewritten in terms of the word alignments, as in (2.2):

$$\mathrm{P}(f_1^J|e_1^I) = \sum_{a_1^J} \mathrm{P}(f_1^J, a_1^J|e_1^I) \tag{2.2}$$

where the word alignment $a_1^J = a_1, ..., a_J$ is introduced as a hidden variable into the translation model. Each alignment a_j maps a source[1] word position to either an 'empty' position ('0') or a target word position ($[1, I]$), i.e. $\forall a_j \in [0, I]$. As mentioned earlier, the IBM models of Brown et al. (1993) comprise five different models depending upon the inclusion of various features, such as fertility and distortion. Here we give an overview on the five IBM models:

- **IBM model 1**: All alignments have the same probability, i.e. the alignment of a target word (e_i) into a source word (f_j) is done uniformly.

- **IBM model 2**: The selection of a source word (f_j) depends on the target position i, i.e. model 2 introduces an alignment probability distribution.

- **IBM model 3**: In addition to the lexical translation and alignment probability, model 3 introduces fertility probability, namely the number of source words (f_j) a target word e_i may generate.

- **IBM model 4**: Model 4 addresses the problem of the ordering of words in the source sentence and introduces a distortion probability model. By taking the relative word positions of the source sentence ($f_1^J = f_1, ..., f_J$) and the target sentence ($e_1^I = e_1, ..., e_I$) into account, the distortion model reorders the source sentence.

- **IBM model 5**: Model 3 and 4 might have impossible alignments in which more than one source word is selected for the same position. Model 5 fixes this problem.

Vogel et al. (1996) introduced the Hidden Markov Model (HMM)-based approach to modelling word alignments, in which the current alignment position a_j depends

[1]Source is the side of translation pair.

on the previously aligned position a_{j-1}. The generalized HMM model (Vogel et al., 1996) includes three different models based on: (a) sentence length probability ($P(J|I)$), (b) transition probability ($P(a_j|a_{j-1}, I)$), and (c) translation probability ($P(f_j|e_{a_j})$). Unlike zero order dependencies of IBM model 1 and 2, the HMM word alignment model (Vogel et al., 1996) is based on first-order dependencies for transition probability ($P(a_j|a_{j-1}, I)$).

The IBM models of Brown et al. (1993) including the HMM expansion of Vogel et al. (1996) are said to be *generative* models, since these models generate a sentence in one language from another language. A comparative overview of various alignment models including the improved alignment models with many extensions is provided in Och and Ney (2000a,b, 2003). The widely used word alignment software GIZA++[2] implements the improved IBM and HMM models (Och and Ney, 2003). We refer to two more pieces of recent research work using *generative* alignment models: word-to-phrase alignment model[3] of Fraser and Marcu (2007a) and m-to-n alignments of Deng and Byrne (2005). In contrast, MT researchers have also applied *discriminative* methods to word alignment, which usually adopt supervised learning algorithms, e.g. Moore (2005), Taskar et al. (2005) and Blunsom and Cohn (2006).

2.1.4 Moving from Word to Phrase

The basic problem of the word-based model proposed by Brown et al. (1988, 1990, 1993) is that their model fails to capture neighbouring contexts, since the translation unit of their model is the individual word. As a result, on the one hand, the word-based model has poor lexical selection, and, on the other hand, this model fails to maintain phrasal cohesion between the phrases of source and target languages. To cope with such kinds of anomalies, Och et al. (1999) suggested a phrase-based model, where they moved from the basic translation unit being a word to a phrase.

[2]http://code.google.com/p/giza-pp/

[3]This model is able to produce alignments which consists of M-to-N non-consecutive translational correspondences.

The proposed alignment template method of Och et al. (1999) includes mainly two consecutive alignment levels:

- Word-level alignment: This approach first takes the intersection of the source-to-target (A1) and the target-to-source (A2) alignment matrices. Then, a refined alignment matrix (A) is produced by adding additional links occurring only in A1 or in A2.

- Phrase-level alignment: Those phrase pairs are extracted from the sentence pair that are *consistent* with the refined alignment matrix A.

For further details of this heuristic approach of phrase extraction, we refer the reader to Zens et al. (2002) and Och and Ney (2004). Koehn et al. (2003) slightly modified the word-level alignment of Och et al. (1999) and proposed two different heuristics, namely, *grow-diag* and *grow-diag-final* (Koehn et al., 2003; Koehn, 2009). Koehn et al. (2003) add a neighbouring alignment point that presents in the union matrix ($A1 \cup A2$) but not in the intersection matrix ($A1 \cap A2$). The process starts from the top right corner[4] of the refined alignment matrix, checks for the alignment points for the first target word, then continues with alignment points for the second target word, and so on. This process is iterated until no alignment point can be added any more (*grow-diag*). In the next step, non-adjacent alignment points are added under the same requirements (*grow-diag-final*). The approach of Koehn et al. (2003) serves as our baseline phrase-based SMT model which we define with a mathematical derivation in Chapter 3.

In addition to the phrase-based conditional probability models (Och and Ney, 2004; Koehn et al., 2003), Marcu and Wong (2002) proposed a phrase-based joint probability model which can simultaneously learn source and target phrases from a sentence-aligned parallel corpus. However, it is rarely used in practice due to its considerable additional computational complexity.

[4]Left and bottom sides of alignment matrix represent respectively target and source sentences.

2.1.5 Decoding & Reranking

2.1.5.1 Decoding

The translation process (i.e. decoding) of the noisy channel model (Brown et al., 1990) (cf. Equation (2.1)) requires an extensive search which is a problem belonging to the class of NP-complete (Knight, 1999) algorithms. There is, accordingly, a growing interest in applying heuristic techniques to the decoding process in order to make it more efficient. A few of the foremost research contributions which improve the decoding by employing heuristic search techniques include:

- A* search (performed as depth-first search) (Och et al., 2001)
- Syntactic parsing-based decoding (Yamada and Knight, 2002)
- Greedy hill climbing decoding (Germann, 2003; Langlais et al., 2007)
- Stack-based beam search decoding (Koehn et al., 2003; Koehn, 2004a)

The current *state-of-the-art* decoder for the phrase-based SMT model is the beam search decoder (Koehn et al., 2003; Koehn, 2004a). This beam search decoding, which is usually performed as breadth-first search, comprises many efficient features, such as: (a) a dynamic programming algorithm to estimate the future cost of an untranslated partial hypothesis, (b) various punning techniques, and (c) a well-implemented data structure to store hypotheses with scores, and other many advantages. The publicly available PB-SMT toolkit Moses (Koehn et al., 2007) deploys a beam search decoder. Our baseline phrase-based SMT model (cf. Section 3.3.1 on page 43) adopts stack-based beam search decoding (Koehn et al., 2003; Koehn, 2004a).

2.1.5.2 Reranking

The baseline system generates an n-best list translation hypotheses. For each hypothesis translation, various lexical, and syntactic features are computed. Then, a

process selects the best translation among the n-best list of hypothesis translations. In sum, this describes the reranking process in SMT. Note that we do not apply any reranking procedures in our work.

Discriminative reranking of the n-best list has emerged as an advanced research topic in SMT. Och et al. (2004) utilized lexical as well as syntactic information from parsers, chunkers, and POS taggers for improving the syntactic well-formedness of the MT output. In their approach, the highest-scoring candidate translation from an n-best list was selected by applying various features both individually and collaboratively into a log-linear model (Och and Ney, 2002).

In a different approach, Kumar and Byrne (2004) proposed Minimum Bayes Risk (MBR) decoding which selects the best candidate translation from an n-best list on the basis of similarity function, i.e. BLEU.

2.1.6 The Log-linear Model

Today's standard phrase-based SMT models (Koehn et al., 2003) follow the log-linear model representation (Och and Ney, 2002) which can put together an arbitrary number of (usually log-linear) features into a single model. Each bilingual log-linear feature applies to single phrase pairs. The log-linear model (Och and Ney, 2002), which is based on the maximum entropy (MaxEnt) principle (Berger et al., 1996), can estimate phrase translation scores directly with a set of log-linear features at decoding time. We present the formal definition of the log-linear model (Och and Ney, 2002) with our baseline PB-SMT system (Koehn et al., 2003) in Section 3.3.1 (cf. page 43). The state-of-the-art PB-SMT model (Koehn et al., 2003) usually employs the following set of log-linear features:

- Phrase translation probability and its inverse
- Lexical translation probability and its inverse
- Word penalty and phrase penalty

- Distance-based or lexicalized phrase reordering models
- N-gram language model

Any additional feature that applies to the source and target phrase pairs can be incorporated in the log-linear model. Each feature of the log-linear model is associated with a weight which is usually estimated using minimum error rate training (MERT)[5] (Och, 2003). Note that it is acknowledged that there is a degree of randomness in how values are assigned to different parameters by MERT. That is, optimizing features via MERT for one particular system may give different values in different times. (Moore and Quirk, 2008) is a way to improve upon this solution. However, in this thesis, for each system set up, we use MERT only once. Tuning is an expensive process, and for the sheer number of experiments carried out in this thesis, addressing this problem would be unduly burdensome. In next two sections, we give a brief overview of the reordering model and the language model.

2.1.6.1 Reordering Model

Reordering is a well-known problem in translation. One natural language differs from another in several grammatical aspects, among which syntactic word ordering is crucial in MT. The basic reordering model of SMT is the distance-based reordering model which is conditioned on the distance between the words in the source sentence. Tillmann (2004) first proposed a lexicalized reordering model for SMT, which conditions reordering on phrases. The lexicalized reordering model considers three different types of *orientation* of a phrase: *monotone*, *swap* and *discontinuous*. Our baseline phrase-based SMT model (Koehn et al., 2003) (cf. Section 3.3.1) adopts the state-of-the-art lexicalized reordering model of Tillmann (2004).

[5]http://www.statmt.org/moses/?n=FactoredTraining.Tuning

2.1.6.2 Language Model

A language model (LM) predicts the likelihood of appearance of a sequence of words in the language. In other words, in SMT, the language model probabilistically measures the linguistic well-formedness of the sentences generated by the MT system. In real situation, a zero probability may be assigned to unseen n-grams. Many smoothing techniques have been introduced to solve this problem. The idea behind the smoothing techniques is that some probability mass is subtracted out ('discounting') from seen n-gram word-sequences and redistributed ('back-off') to unseen n-grams. MT researchers usually build language models with the interpolated modified Kneser-Ney smoothing (Chen and Goodman, 1998) technique. Interpolation causes the discounted n-gram probability estimates at the specified order n to be interpolated with lower order estimates. The LM order is usually set to 5- or higher-gram to capture a wide range of contexts on the target side. SRILM[6] is a freely available toolkit for building statistical language models. In order to carry out our experiments, we use SRILM with the above setting (i.e. 5-gram and interpolated modified Kneser-Ney smoothing) to build the language models.

2.1.7 Target Syntactified Model

State-of-the-art PB-SMT models (Koehn et al., 2003; Och et al., 1999) produce acceptable translations for language-pairs with similar word orders. However, these models generate more ungrammatical output for language pairs which differ a lot in terms of word order (e.g. Chinese-to-English, Arabic-to-English). In order to obtain grammatical MT output, researchers have made use of the syntactic information available from the target side of the translation pair. Marcu et al. (2006) proposed a phrase-based SMT model where the target-language parse trees are mapped into the source sentence via a transduction process. Their syntactified phrase-based MT improves over a baseline PB-SMT model in terms of translation quality in a Chinese-

[6]http://www-speech.sri.com/projects/srilm/

to-English translation task. Hassan et al. (2006, 2007, 2008) and Hassan (2009) showed that incorporating lexical syntactic descriptions in the form of supertags in the target language model and on the target side of the translation model could improve significantly over the state-of-the-art PB-SMT model (Koehn et al., 2003).

2.1.8 Hierarchical Phrase-Based SMT

The state-of-the-art phrase-based SMT model (Koehn et al., 2003) has a few weaknesses despite the fact that PB-SMT dominates other approaches to MT. The first drawback of the PB-SMT model is that this model cannot incorporate non-contiguous phrases. Secondly, reordering in PB-SMT is modelled suboptimally since PB-SMT cannot handle long-distance reordering properly. As mentioned earlier (cf. Section 2.1.7), the problem usually arises for those language pairs that are syntactically divergent and have a lesser degree of phrasal cohesion (like English and Chinese).

To remedy such kinds of weakness, Chiang (2005, 2007) proposed a model of hierarchical phrase-based SMT (HPB-SMT) that uses the bilingual phrase pairs of phrase-based SMT (Koehn et al., 2003) as a starting point to learn hierarchial rules using probabilistic synchronous context-free grammar (PSCFG). The decoding process in the HPB-SMT model is based on bottom-up chart parsing (Chiang, 2005, 2007). This chart parsing decoder, known as Hiero, does not require explicit syntactic representations on either side of the phrases in rules. We give a formal definition of the Hiero model (Chiang, 2007) in Section 3.3.2 (cf. page 43).

The Hiero model of Chiang (2007) employs an efficient pruning technique, namely 'cube' pruning. Cube pruning generates the n-best new chart entries efficiently on the basis of the different model estimates including language model scores. We refer the reader to (Huang and Chiang, 2007) for a detailed description with examples of the cube pruning algorithm.

2.1.8.1 Syntactic Constraints

We have just mentioned that the state-of-the-art Hiero (Chiang, 2007) model does not require explicit syntactic representations on either side of the phrases in rules. The phrases in Hiero rules comprise a combination of terminal and nonterminal (NT) symbols. Nonterminal in Hiero may not be a proper linguistic constituent like noun phrases (NP) and verb phrases (VP). Moreover, Hiero nonterminals take only a single form (i.e. NT rather than linguistic nonterminal VP, NP). MT researchers have tried to model syntax-aware nonterminals instead of these non-syntactic Hiero nonterminals on either side of the rules. For instance, Zollmann and Venugopal (2006) and Marton and Resnik (2008) included 'soft' constituent-level constraints based on the phrase structure parse trees of the target language and the source language, respectively, to improve translation quality of the Hiero model (Chiang, 2007). However, such models (Zollmann and Venugopal, 2006; Marton and Resnik, 2008) generates much bigger rule-tables than baseline rule-table (Hiero), which brings additional computational complexity.

2.1.9 Discriminative Training

Discriminative training is an emerging research topic in the SMT. Minimum error rate training (MERT) of Och (2003) is a bottleneck for developing an SMT system with a large number of features since it can optimize the weights of only a limited number of features. In contrast to the limited number of features in a PB-SMT model (Koehn et al., 2003), the discriminative training model can optimize the weights of a large number of features over millions of bilingual training sentence pairs. Discriminative training is an example of supervised learning. The discriminative model assumes each phrase translation is a feature, and iteratively learns the usefulness of that phrase translation and sets an appropriate weight.

Tillmann and Zhang (2006) employed a stochastic gradient descent method to learn the weights of millions of features derived from the source and target block

('phrase') sequences. They set a binary value to each translational feature. In addition to the 'unigram' block-based feature, Tillmann and Zhang (2006) employed block orientation,[7] as well as source and target n-gram features in their model. Liang et al. (2006) applied an online perceptron training algorithm to set the weights of millions of translational features. In addition to the translation and the language model log-probabilities, Liang et al. (2006) employed lexical and POS-based features to their model for learning weights. Wang and Shawe-Taylor (2007) applied a kernel regression method to achieve a similar goal.

In order to learn the weights of a large number of feature functions, Wellington et al. (2006) employed the regularization learning algorithm to a tree translation model while reducing over-fitting and generalization errors. Cowan et al. (2006) used a perceptron algorithm for discriminative training in a German-to-English translation task. The discriminative model of Cowan et al. (2006) predicts the English aligned extended projection from a given German clause structure.

Like previous approaches (Liang et al., 2006; Tillmann and Zhang, 2006), Watanabe et al. (2007) exploited a large number of binary features in the Hiero model (Chiang, 2007) using an online discriminative training method. They used the margin infused relaxed algorithm (MIRA) (Crammer et al., 2006) to optimize the weights of millions of features. In addition to the Hiero baseline features, Watanabe et al. (2007) incorporated binary feature, insertion and deletion features, word-pair feature, target bigram and normalized token features into the SMT model. Chiang et al. (2008) explored the use of the MIRA algorithm as an alternative to MERT (Och, 2003). Chiang et al. (2009) analyzed the impact of adding a large number of features into the Hiero model as well as the syntax-based model of Galley et al. (2006) while optimizing their weights using MIRA. In our work, we deal with a limited number of features that MERT (Och, 2003) can fairly handle.

In a similar manner to the above models, we deploy a discriminative learning

[7]Blocks are phrase pairs consisting of target and source phrases and local phrase reordering is handled with block orientation.

model in our work in order to incorporate contextual dependencies. There are a few basic differences between our approach with those mentioned above. We introduce new log-linear features (i.e. context-informed log probabilities) while retaining the strengths of the existing state-of-the-art SMT models (Koehn et al., 2003; Chiang, 2007). The above models usually consider each phrasal translation as a single feature. We apply a memory-based learning algorithm (Daelemans and van den Bosch, 2005) to the training examples, and build decision trees on the basis of contextual features of the source phrases; then, the context-dependent phrasal translations are learned from the decision trees at decoding time. On the other hand, the above mentioned discriminative models generally deploy the online learning algorithms to set the weight to each instance of the training data.

2.2 Research Works Relating to the Thesis

In this section, we describe previous research which has suggested novel methods to incorporate contextual information into statistical models of machine translation in order to improve translational choice and the quality of translation. Context has been incorporated into both the source and target sides of the translation pair. Techniques to incorporate context into SMT can be broadly divided into two categories: *source-context modelling* (e.g. Carpuat and Wu, 2007; Giménez and Màrquez, 2007; Stroppa et al., 2007) and *target-context modelling* (e.g. Berger et al., 1996; Hasan et al., 2008). In the first two subsections of this section we describe twenty three studies of context modelling with English as the target language. In the third subsection we highlight six studies that use English as the source language.

In the three subsections we group related work according to six key aspects. Each aspectual overview of related work is accompanied by a table highlighting the contrastive features of the studies discussed (Tables 2.1 to 2.6). In each table we list the contextual features employed by each study and the types of SMT models employed. In the second column of each table 'SL→TL' stands for 'source language→target

language', referring to the translation pair and direction; 'DS' refers to the 'data sets' used to train SMT models; and 'S/L' stands for 'small/large', indicating a division between training set sizes below and above 500,000 words that we use to structure our experiments. In the third column, 'SL' and 'TL' respectively stand for 'source' and 'target' languages of the translation pair from which the listed contextual features are extracted.

2.2.1 Source Context Modelling

Approaches to integrating source-language contextual information into different stages in the SMT model can in turn be broadly divided into: (i) *discriminative word alignment* (e.g. Brunning et al., 2009; Patry and Langlais, 2009) for creating improved word-to-word translation lexicons, and (ii) *discriminative translation filtering* (e.g. Carpuat and Wu, 2007; Chan et al., 2007; Stroppa et al., 2007) by learning context-dependent translation probabilities.

2.2.1.1 Discriminative Word Alignment

García-Varea et al. (2001, 2002) present a MaxEnt approach to integrate contextual dependencies of both the source and target sides of the statistical alignment model to develop a refined context-dependent lexicon model. Using such a model, on the German-to-English Verbmobil and the French-to-English Canadian Parliament Hansards corpora, they obtained better alignment quality in terms of improved alignment error rate (AER) (Och and Ney, 2000a). However, since alignment is not an end-task in itself and is most often used as an intermediate task to generate phrase alignments in PB-SMT systems, improved AER scores do not necessarily result in improved translation quality, as noted by a number of researchers (Fraser and Marcu, 2007b; Ma et al., 2008).

More recently, Patry and Langlais (2009) proposed an alignment model which does not assume prior word alignments and considers all source words jointly when evaluating the probability of a target word. They use a multilayer perceptron classi-

fier to estimate this probability. The word alignment results surpassed IBM model 1 when their model was extended to include alignment information. However, the proposed model is not verified with experimentation as to whether it can help improve translation quality of an SMT system.

2.2.1.2 Discriminative Translation Filtering

Discriminative translation filtering in SMT, in which contextual information from the source language is used to weight or select from the potentially large set of lexical or phrasal translations, can furthermore be divided into two categories: (i) *hard interaction* (e.g. Carpuat et al., 2005) and (ii) *soft interaction* (e.g. Chan et al., 2007; Stroppa et al., 2007). Alternatively, the same techniques can also be classified according to their use of (i) *hard constraints* (e.g. Stroppa et al., 2007) or (ii) *soft constraints* (e.g. Carpuat et al., 2007; Marton and Resnik, 2008; Xiong et al., 2010).

- ***Hard vs Soft Interaction***: In soft interaction, the WSD-like translation predictions, i.e. the context-informed translation models are allowed to interact with other log-linear models (e.g. language model, distortion model and additional translation models) at decoding time. In other words, scoring, ranking, pruning and selection of translation hypotheses during decoding are performed with a consensus of all SMT models including the additional context-informed model. In hard interaction, the WSD-like translation predictions are used during pre-processing or post-processing and they do not interfere with the SMT process itself. In other words, the weights of the context-dependent translations of a source phrase are not interwove with other SMT models to select the best candidate translations.

- ***Hard vs Soft Constraints***: In the soft constraints model, the decoder is allowed to use all possible candidate phrases for a source phrase, while soft constraints such as weights are introduced to influence the decoder's lexical

selection model. In the hard constraints model, the decoder is forced to use a restricted but supposedly more appropriate set of candidate phrases for a source phrase; in addition, the context-informed model imposes weights on the candidate phrases on the basis of additional contextual information to influence the decoder's lexical selection model.

According to the above classifications, like the work of Stroppa et al. (2007), our context-informed models (cf. Chapter 3) interact 'softly' with the other SMT models, and we impose hard constraints on the decoder. The main reason behind following the approach of Stroppa et al. (2007) is that their contextual models significantly outperform the state-of-the-art PB-SMT using only the basic contextual features[8] in two standard translation tasks.

Discriminative translation filtering in SMT can further be divided into the following four categories according to its deployment into different types of SMT engines:

- **Word-Based SMT**: Table 2.1 lists related research that integrates context into word-based SMT models. Brown et al. (1991a,b) were the first to propose the use of dedicated WSD models in word-based SMT systems, using an English-to-French translation task as their testbed. An instance of a word is assigned a sense based on mutual information with the word's translation. Evaluation was limited to the case of binary disambiguation, i.e. deciding between only the two most probable translation candidates, and to a reduced set of common words. A significant improvement in translation quality was reported according to manual evaluation.

  Vickrey et al. (2005) built classifiers inspired by those used in WSD to fill in blanks in a partially completed translation. This blank-filling task is a simplified version of the translation task, in which the (possibly incorrect) target context surrounding the word translation is already available. They integrated a WSD-based model into the decoder in a 'soft' way, i.e. allowing

[8]Basic features refer to words and POS tags.

it to interact with other models. The evaluation of the accuracies of a blank filling task is essentially a WSD evaluation task.

However, initial attempts to embed context-rich approaches from WSD methods into SMT systems to enhance lexical selection did not lead to any improvement in translation quality (Carpuat and Wu, 2005). Carpuat and Wu (2005) integrated a WSD model into a word-based SMT system in two ways: (i) the WSD model constrains the set of potential senses considered by the decoder; and (ii) the SMT output is post-processed by directly replacing translation candidates with the WSD predictions. The integration of the WSD model into the SMT system was performed in a 'hard' manner, and both approaches were found to hurt translation quality. However, in their later work (Carpuat and Wu, 2005b), they showed that SMT systems alone perform much worse than WSD systems on a WSD task, which suggests that SMT should benefit from the WSD predictions.

Authors	SL→TL [DS] [S/L]	Contextual Features	Integrated into
Brown et al. (1991a,b)	Fr→En[CPH][L]	SL:Neighbouring words and basic POS	WB-SMT model
Vickrey et al. (2005)	Fr→En[Europarl][L]	SL:POS and Neighbouring words	WB-SMT model
Carpuat & Wu (2005a,b)	Zh→En [UN:LDC][L]	SL:Position-sensitive syntactic, and local collocations	WB-SMT model

Table 2.1: Related research integrating context into word-based SMT (WB-SMT) models. Notations: {SL: Source Language, TL: Target Language, DS: Data Sets, S/L: Small/Large, En: English; Fr: French; Zh:Chinese; CPH: Canadian Parliament Hansards, UN: United Nations}.

- ***Phrase-Based SMT***: Table 2.2 summarizes related research on integrating context into PB-SMT models. Carpuat et al. (2006) reported small improvements in BLEU score when incorporating single-word WSD predictions in a Pharaoh (Koehn, 2004a) baseline. They train their WSD system on the same corpus used to build SMT models. Automatic evaluation shows these gains are not consistent across all evaluation metrics.

Target language models arguably play the most significant role in today's PB-SMT systems (Koehn et al., 2003). However, for some time now people have believed that incorporation of some source-language information into SMT systems was bound to help. Stroppa et al. (2007) integrated source-side contextual features into a *state-of-the-art* log-linear PB-SMT system (Koehn et al., 2003) by adding the context-dependent phrasal translation probabilities learned using a decision-tree classifier (Daelemans and van den Bosch, 2005). They considered up to two words and/or POS tags on either side of the source phrase as contextual features. Significant improvements over a baseline PB-SMT system were obtained on Italian-to-English and Chinese-to-English IWSLT tasks. Several proposals have recently been made to exploit the accuracy and the flexibility of discriminative learning fully (Liang et al., 2006; Tillmann and Zhang, 2006). Work of this type generally requires a redefinition of the training procedure; in contrast, Stroppa et al. (2007) introduce new features while retaining the strength of existing state-of-the-art systems.

More recent approaches of integrating state-of-the-art WSD methods into PB-SMT (Carpuat and Wu, 2007; Giménez and Màrquez, 2007; Gimpel et al., 2008; Max et al., 2008) have also met with success in improving the overall translation quality. Giménez and Màrquez (2007) extended the work of Vickrey et al. (2005) by (i) considering the more general case of frequent phrases and moving to full translation rather than the blank-filling task on the target side, and (ii) moving from word translation to phrase translation. The contextual features were defined by taking into account words of the immediate context, n-grams, part-of-speech, lemmas, chunk label as well as global features (bag-of-words). Further study by Giménez and Màrquez (2009) shows that their discriminative models yield significantly improved lexical choice over a PB-SMT model, which does not necessarily lead to improved grammaticality.

Carpuat and Wu (2007) use bag-of-words features, local collocations, position-specific local part-of-speech tags, and basic dependency features as contextual

features for a phrase translation disambiguation task, producing consistent gains across all evaluation metrics on IWSLT 2006 and NIST Chinese-to-English translation tasks.

Costa-Jussà and Banchs (2010) integrate source context information in the Moses decoder (Koehn et al., 2007) by incorporating a contextual feature function estimated using a cosine distance similarity metric. The feature function is computed for each phrase by determining cosine distance between the input sentence to be translated and the source sentences in the training corpus. They showed a slight improvement over the Moses baseline on the internal test set for Arabic-to-English and Chinese-to-English translation tasks in the IWSLT'09 evaluation campaign.

Authors	SL→TL [DS] [S/L]	Contextual Features	Integrated into
Carpuat & Wu (2006)	Ar→En[IWSLT][S] It→En[IWSLT][S] Jp→En[IWSLT][S]	SL: Position-sensitive syntactic, and local collocations	PB-SMT model
Carpuat & Wu (2007)	Zh→En[IWSLT][S] Zh→En[NIST][L]	SL:Bag-of-words, collocations, POS and dependency features	PB-SMT model
Giménez & Màrquez (2007, 2009)	Sp→En[Europarl][L]	SL:Local context, n-grams, POS, lemmas, chunk label & bag-of-words	PB-SMT model
Stroppa et al., (2007)	Zh→En[IWSLT][S] It→En[IWSLT][S]	SL:Neighbouring words and POS tags	PB-SMT model
Costa-Jussà & Banchs (2010)	Ar→En[IWSLT][S] Zh→En[IWSLT][S]	SL: Sentence similarity features	PB-SMT model

Table 2.2: Related research integrating context into PB-SMT models. Notations: {SL: Source Language, TL: Target Language, DS: Data Sets, S/L: Small/Large, En: English; It: Italian; Sp: Spanish; Zh: Chinese}.

- ***Hierarchical Phrase-Based SMT***: Table 2.3 lists related research that integrates context into hierarchical PB-SMT (HPB-SMT) (Chiang, 2007) models. Chan et al. (2007) were the first to use a WSD system to integrate additional features in HPB-SMT, achieving statistically significant performance improvements for several automatic measures for Chinese-to-English translation.

Despite not mentioning the obvious link between the two pieces of work, He et al. (2008) show that the source-language features used by (Stroppa et al., 2007) are also of benefit when used with the Hiero decoder (Chiang, 2007).

More recently, Shen et al. (2009) proposed a method to include linguistic and contextual information in the HPB-SMT model. The features employed in the system are non-terminal labels, non-terminal length distribution,[9] source context, and a language model created from source-side grammatical dependency structures. While their source-side dependency language model does not produce any improvement, the other features seem to be effective in Arabic-to-English and Chinese-to-English translation. Chiang et al. (2009) define new translational features using neighbouring word contexts of the source phrase, which are directly integrated into both the translation model of the Hiero system and the syntax-based system of Galley et al. (2006).

Authors	SL→TL [DS] [S/L]	Contextual Features	Integrated into
Chan et al. (2007)	Zh→En[FBIS][S]	SL:Local collocations, POS tags and neighbouring words	Hiero
He et al. (2008)	Zh→En[IWSLT][S] Zh→En[NIST 03][S]	SL: POS; SL &TL: Words and length distribution features	Hiero
Shen et al. (2009)	Ar→En[NIST 06, 08][L] Zh→En[NIST 06, 08][L]	SL:Nonterminal labels, length, context LM and dependency LM	Hiero
Chiang et al. (2009)	Zh→En[NIST][L]	SL:Neighbouring words	Hiero & syntax-based model

Table 2.3: Related research integrating context into Hiero models. Notations: {SL: Source Language, TL: Target Language, DS: Data Sets, S/L: Small/Large, En: English; Zh:Chinese; Ar: Arabic; FBIS: Foreign Broadcast Information Service}.

- ***Alternative SMT Architectures***: Table 2.4 lists related research that integrates context into alternative SMT models. Bangalore et al. (2007) proposed an SMT architecture based on stochastic finite state transducers (FST), that addresses global lexical selection in which parameters are discriminatively trained using a MaxEnt model considering n-gram features from the source

[9]Lengths of subphrases covered by non-terminals.

Authors	SL→TL [DS] [S/L]	Contextual Features	Integrated into
Bangalore et al. (2007)	Ar→En[UN][L] Fr→En[CPH][L] Zh→En[IWSLT][S]	SL:Bag-of-words	FST-based MT model
Ittycheriah et al. (2007)	Ar→En[UN, NIST 06][L]	SL:Lexical, morphological & syntactic features	Proposed DTM2 model
Gimpel & Smith (2009)	De→En[BTEC][S]	SL & TL:Syntactic features from dependency trees	Proposed MT model

Table 2.4: Related research integrating context into alternative SMT models. Notations: {SL: Source Language, TL: Target Language, DS: Data Sets, S/L: Small/Large, En: English; Fr: French; De: German; Zh:Chinese; Ar: Arabic; CPH: Canadian Parliament Hansards; UN: United Nations; BTEC: Basic Travel Expression Corpus; FST: Finite State Transducer}.

sentence. They deal with reordering in a different manner, where source sentences are reordered according to the target-language-specific ordering. The results obtained on Chinese-to-English IWSLT translation, Arabic-to-English translation of Proceedings of the United Nations and French-to-English translation of Proceedings of the Canadian Parliament show significant improvements.

Ittycheriah et al. (2007) introduced the Direct Translation Model 2 (DTM2), a MaxEnt-based SMT architecture, which differs from PB-SMT models in that it extracts a non-redundant set of minimal phrases from a word-aligned parallel corpus such that no two phrases overlap with each other. The main difference between the DTM2 decoder and the standard PB-SMT decoders is that DTM2 employs discriminative MaxEnt models to obtain the translation likelihoods by deploying lexical, morphological and syntactic contextual features. They report improvements over a state-of-the-art PB-SMT decoder in Arabic-to-English translation task.

Gimpel and Smith (2009) present an MT framework based on lattice parsing with a quasi-synchronous grammar that can incorporate arbitrary features from both source and target sentences. They show that phrase features and dependency syntax produce improvements in translation quality on the German-

to-English portion of the Basic Travel Expression Corpus (BTEC), although compared to a state-of-the-art SMT system their model produces considerably lower scores.

2.2.2 Target Context Modelling

Table 2.6 enumerates related research that integrates context into word alignment models. Berger et al. (1996) suggested context-sensitive modelling of word translations in order to integrate local contextual information into their IBM translation models using a MaxEnt model. Probability distributions are estimated by MaxEnt based on position-sensitive local collocation features in a window of three words around the target word. This work is not supported by significant evaluation results. Mauser et al. (2009) extended the work of Hasan et al. (2008) (cf. Section 2.2.3) by integrating additional discriminative word lexicons into the PB-SMT model, by using sentence-level source information to predict appropriate target words.

Authors	SL→TL [DS] [S/L]	Contextual Features	Integrated into
Berger et al. (1996)	Fr→En[CPH][L] De→En[Verbmobil][S]	TL:Neighbouring words	IBM model
García-Varea et al. (2001, 2002)	De→En[Verbmobil][S] Fr→En[CPH][L]	SL & TL:Neighbouring words and word class	IBM model
Mauser et al. (2009)	Zh→En[GALE][L] Ar→En[NIST 08][L] Zh→En[NIST 08][L]	TL:Neighbouring words & SL:Sentence level lexical feature	IBM model & proposed discriminative WA model
Patry & Langalais (2009)	Fr→En[Europarl][S]	SL:Bag-of-words	Proposed WA model

Table 2.5: Related research integrating context into word alignment models. Notations: {SL: Source Language, TL: Target Language, DS: Data Sets, S/L: Small/Large, En: English; Fr: French; De: German; Zh:Chinese; Ar: Arabic; CPH: Canadian Parliament Hansards; WA: Word Alignment}.

2.2.3 English as Source Language

It is a common belief that translating into a less inflected language (such as English) from a highly inflected language should be more effective than the other way round where context-sensitive translation is concerned. This belief is hardly challenged

in the related work cited in this section; all above-mentioned twenty three studies translate to English. Nonetheless, Table 2.6 lists six studies that take English as the source language. Most of these studies employ contextual features computed on the English input; the ample availability of Natural Language Processing (NLP) tools for English, such as part-of-speech taggers and parsers, makes this possible.

Both Max et al. (2008) and Gimpel et al. (2008) worked with a state-of-the-art PB-SMT system (Koehn et al., 2003) and focused on language pairs where the target is not English. Using local as well as broader contexts in addition to grammatical dependency information, Max et al. (2008) report no significant gains over a PB-SMT baseline model in terms of automatic evaluation, yet modest gains are observed in manual evaluation. Gimpel et al. (2008) employed lexical and positional features as well as various shallow syntactic features extracted from phrase structure parse tree. They worked with two different English-to-German data sets (WMT'07 and WMT'08) and performed a range of experiments. Most gains were not statistically significant for the WMT'07 translation task, while most gains were statistically significant for WMT'08 translation task. In addition to the English-to-German translation tasks, in their Chinese-to-English translation tasks, Gimpel et al. (2008) achieved statistically significant gains for the UN task; however, their approach did not work on the NIST News and the combined (UN + NIST News data) tasks. Furthermore, they were unable to show any improvements for the German-to-English translation task.

Venkatapathy and Bangalore (2007) conducted experiments on a small amount of English-to-Hindi training data with their proposed global lexical selection and sentence reconstruction model. Their bag-of-words model considers all words of the source sentence as features, regardless of their positions. Using these features, a MaxEnt-based classifier predicts target words that should occur in the target sentence. The target sentence is determined by permuting the generated target words using a language model.

On an English-to-Portuguese translation task, Specia et al. (2008) worked with a

Authors	SL→TL [DS] [S/L]	Contextual Features	Integrated into
Max et al. (2008)	En→Fr[Europarl][S]	SL:Neighbouring words & POS, & dependency relations	PB-SMT model
Gimpel & Smith (2008)	Zh→En[NIST 08, UN][L] De→En[WMT 07][L] En→De[WMT 07, 08][L]	SL:Lexical, shallow syntactic and positional features	PB-SMT model
Venkatapathy & Bangalore (2007)	En→Hi[News][S]	SL:Bag-of-words	Proposed global lexical selection model
Specia et al. (2008)	En→Pt[Europarl][L]	SL:Morphological features (person, tense & number)	Dependency treelet system
Hasan et al. (2008)	Zh→En[IWSLT][S] Sp→En[TC-STAR EPPS][L] En→Sp[TC-STAR EPPS][L]	TL:Neighbouring words	IBM model
Brunning et al. (2009)	Ar→En[NIST 08][S] En→Ar[NIST 08][S]	SL & TL:POS tag	MTTK WA model

Table 2.6: Related research using English as source language. Notations: {SL: Source Language, TL: Target Language, DS: Data Sets, S/L: Small/Large, En: English; Fr: French; De: German; Sp: Spanish; Zh:Chinese; Ar: Arabic; Pt: Portuguese; Hi: Hindi; WA: Word Alignment}.

syntactically motivated PB-SMT system (Quirk et al., 2005) and their WSD model was limited to disambiguating a small set of words, namely 10 highly frequent and ambiguous verbs.

Two more research works (Hasan et al., 2008; Brunning et al., 2009) considered English as the source language. Both approaches focused on improving word alignment for creating refined word-to-word translation lexicons. Hasan et al. (2008) present target context modelling into SMT using a triplet lexicon model that captures long-distance dependencies. Their approach is evaluated in a reranking framework; slight improvements are observed over IBM model 1 in terms of BLEU (Papineni et al., 2002) and translation edit rate (TER) (Snover et al., 2006). Brunning et al. (2009) introduce context-dependent alignment models for MT that exploit source-language context information to estimate word-to-word translation probabilities using a decision tree algorithm. Their model decreases the AER compared to a context-independent model, and improves translation quality in Arabic-to-English and English-to-Arabic translation tasks.

We can see from the above summary tables that a range of contextual features have been employed at different stages in the SMT model. In this work, we inte-

grate a range of contextual features into the state-of-the-art SMT models (Koehn et al., 2003; Chiang, 2007), including neighbouring words and POS tags of the source phrases as in (Stroppa et al., 2007). We introduce lexical syntactic descriptions in the form of supertags and semantic roles as new contextual features in the SMT model. Moreover, we investigate the integration of deep syntactic information (grammatical dependency relations) into the SMT models as in (Max et al., 2008). A more detailed account of the main difference between our approach and that of (Max et al., 2008) with respect to various aspects is given in Section 6.2. We explore a number of sentence similarity-based contextual features including cosine distance (Costa-Jussà and Banchs, 2010) in PB-SMT. Furthermore, various contextual features have been employed collaboratively in the PB-SMT and the Hiero models. Table 2.7 summarizes various source-language contextual features that we have taken into account in our experiments, most of which are introduced as new contextual features in the state-of-the-art SMT models (Koehn et al., 2003; Chiang, 2007).

Source Language Contexts	PB-SMT			Hiero		
	Used	Novel	Authors	Used	Novel	Authors
Words & POS tags	Y		(Stroppa et al., 2007, ,etc.)	Y		(Chan et al., 2007, ,etc.)
Supertags	Y	Y	(Haque et al., 2009a)	Y	Y	(Haque et al., 2010a)
Dependency Relations	Y		(Max et al., 2008) (Haque et al., 2009b)	Y	Y	(Yet to be published)
Semantic Roles	Y	Y	(Haque et al., 2011)			
Sentence-similarity Features			(Costa-Jussà and Banchs, 2010) (Haque et al., 2010b)			
Feature Combinations	Y			Y		

Table 2.7: List of contextual features employed in our experiments.

In this thesis, we report a range of experiments on several data sets considering English as the source language of the translation pairs (English-to-Hindi, English-to-Czech, English-to-Dutch, English-to-Chinese, English-to-Japanese, English-to-Spanish). We examine the scalability of our research to larger amounts of training data. Furthermore, we report on experiments with the Dutch-to-English translation, using experimental data from two different domains.

2.3 Summary

In this chapter, first we presented an overview of MT, and provided a brief overview of foundation of SMT. We mentioned the importance of word alignment in SMT. Then, we reassessed the idea state-of-the-art SMT models with a discussion on a few significant research contributions in those areas. Paying extra attention on research work involving discriminative training in SMT, we identified differences between our approach with discriminative learning in SMT. In the second part of this chapter we presented the previous studies related to this thesis, which have been grouped according to six key aspects. Research work in each group is summarized in tabular form to highlight their contrastive features.

In the next chapter, we will define the state-of-the-art SMT models as well as our context-informed SMT models formally. Then, we will elaborate the following areas: memory-based classification approaches used in this thesis, implementation issues of context-sensitive SMT models, data set statistics, and an overview of MT evaluation techniques used to validate our experimental results.

Chapter 3

Context-Informed SMT

3.1 The state-of-the-art SMT Models

The work demonstrated in this thesis has been carried out with two state-of-the-art SMT models: phrase-based SMT (PB-SMT) (Koehn et al., 2003) and hierarchical phrase-based SMT (HPB-SMT) (Chiang, 2007), both of which serve as our baseline models. In the following two subsections, we give formal definitions of the two state-of-the-art SMT models with mathematical derivations.

3.1.1 PB-SMT Baseline

In this section, we formally define the phrase-based SMT model of Koehn et al. (2003), which serves as one of our baseline models. The translation task in SMT can be viewed as a search problem (Brown et al., 1993), in which the goal is to find the most probable candidate translation $e_1^I = e_1, ..., e_I$ for the given input sentence $f_1^J = f_1, ..., f_J$. The best translation can be obtained applying the noisy channel model (Brown et al., 1990) of translation by maximizing $\mathrm{P}(e_1^I|f_1^J)$, as defined in Equation (2.1) (cf. page 11). In noisy channel model, the translation process can be viewed as a product of two terms: the translation model ($\mathrm{P}(f_1^J|e_1^I)$) and the target language model ($\mathrm{P}(e_1^I)$).

The log-linear translation model (Och and Ney, 2002) is a special case of the

noisy channel model of translation, in which the posterior probability $\mathrm{P}(e_1^I|f_1^J)$ is directly modelled as a (log-linear) combination of features, that usually comprise M translational features, and the language model, as in (3.1):

$$\log \mathrm{P}(e_1^I|f_1^J) = \sum_{m=1}^{M} \lambda_m h_m(f_1^J, e_1^I, s_1^K) + \lambda_{\mathrm{LM}} \log \mathrm{P}(e_1^I) \tag{3.1}$$

where $s_1^K = s_1, ..., s_k$ denotes a segmentation of the source and target sentences respectively into the sequences of phrases $(\hat{f}_1, ..., \hat{f}_k)$ and $(\hat{e}_1, ..., \hat{e}_k)$ such that (we set $i_0 := 0$):

$$\begin{aligned} \forall k \in [1, K] \quad & s_k := (i_k; b_k, j_k), (b_k \text{ corresponds to starting index of } f_k) \\ & \hat{e}_k := \hat{e}_{i_{k-1}+1}, ..., \hat{e}_{i_k}, \\ & \hat{f}_k := \hat{f}_{b_k}, ..., \hat{f}_{j_k} \end{aligned}$$

Since both the source and target phrases may appear in any position[1] of the respective sentence, we define them differently. Each feature h_m in Equation (3.1) can be rewritten as in (3.2):

$$h_m(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^{K} \hat{h}_m(\hat{f}_k, \hat{e}_k, s_k) \tag{3.2}$$

In theory, log-linear PB-SMT features can apply to the entire sentence, but in practice, those features apply to a single phrase-pair $(\hat{f}_k, \hat{e}_k)$. Thus translational features in (3.1) can be rewritten as in (3.3):

$$\sum_{m=1}^{M} \lambda_m \sum_{k=1}^{K} \hat{h}_m(\hat{f}_k, \hat{e}_k, s_k) = \sum_{k=1}^{K} \tilde{h}(\hat{f}_k, \hat{e}_k, s_k), \qquad \text{with } \tilde{h} = \sum_{m=1}^{M} \lambda_m \hat{h}_m \tag{3.3}$$

In Equation (3.3), $\hat{h}_m$ is a feature defined on phrase-pairs $(\hat{f}_k, \hat{e}_k)$, and λ_m is the feature weight of $\hat{h}_m$. One intuitively natural feature is the phrase translation log-probability ($\hat{h}_m = \log \mathrm{P}(\hat{e}_k|\hat{f}_k)$) where probabilities are estimated using relative

[1]For example, the first target phrase $\hat{e}_1$ ($k = 1$) may be aligned with the source phrase that appears in the last position of the source sentence.

frequency count for a phrase pair $(\hat{f}_k, \hat{e}_k)$ independent of any other context information. Other typical features used in PB-SMT (Koehn et al., 2003) are derived from the inverse phrase translation probability (log $\mathrm{P}(\hat{f}_k|\hat{e}_k)$), the lexical probability (log $\mathrm{P}_{\mathrm{lex}}(\hat{e}_k|\hat{f}_k)$), its inverse (log $\mathrm{P}_{\mathrm{lex}}(\hat{f}_k|\hat{e}_k)$), and language model. Our context-informed model will be expressed as additional features in the model.

3.1.2 HPB-SMT Baseline

In this section, we formally define the hierarchical phrase-based SMT model of Chiang (2007), which serves as another baseline model. The hierarchical phrase-based SMT model (also known as Hiero) is based on probabilistic synchronous context-free grammar (PSCFG). Synchronous rules in Hiero take the form as in (3.4):

$$X \rightarrow \langle \alpha, \gamma, \sim \rangle \tag{3.4}$$

where X is the nonterminal (NT) symbol, and α and γ are the source and target phrases, which contain combinations of terminal and nonterminals in the source and target language. The $\sim$ symbol indicates a one-to-one correspondence between NTs in α and γ. In practice, the number of NTs on the right hand side is constrained to at most two (Chiang, 2007), which must be separated by lexical items in α.

Like the log-linear representation of the PB-SMT features shown in Section 3.1.1, each Hiero rule is associated with a score which is derived using the log-linear model (Och and Ney, 2002) as in (3.5):

$$w(X \rightarrow \langle \alpha, \gamma, \sim \rangle) = \sum_i \lambda_i \phi_i \tag{3.5}$$

where ϕ_i is a feature defined on rules and λ_i is the feature weight of ϕ_i. One intuitively natural feature is phrase translation log-probability ($\phi(\alpha, \gamma)$= log $\mathrm{P}(\gamma|\alpha)$). Like the PB-SMT features, the other typical features used in Hiero are derived from the inverse phrase translation probability $\mathrm{P}(\alpha|\gamma)$, the lexical probability $\mathrm{P}_{\mathrm{lex}}(\gamma|\alpha)$ and its inverse $\mathrm{P}_{\mathrm{lex}}(\alpha|\gamma)$. In the hierarchical model, translation probabilities are es-

timated using a relative frequency count for a phrase pair $\langle \alpha, \gamma \rangle$ independent of any other context information. Thus, a limitation that Hiero shares with the PB-SMT model (Koehn et al., 2003) is that it also does not take into account the contexts in which the source-sides of the rules appear. In other words, it can be argued that rule selection in Hiero is suboptimally modelled. Our context-informed model will be expressed as an additional feature in the model. In addition to these features the system generally employs a word penalty, a phrase penalty, a glue rule penalty, and language model features.

The translation task in HPB-SMT can be expressed as a CYK parsing with beam search together with a post-processor for mapping source derivations to target derivations (Chiang, 2007). The most likely translation $\hat{e}$ of a source sentence f is formulated as a search for the most probable derivation d whose source side is equal to f, as in (3.6):

$$\hat{e} = e\left(\underset{d \in D(G): src(d)=f}{\operatorname{argmax}} p(d)\right) \tag{3.6}$$

where *e(d)* is the target-side yield of derivation d, and *D(G)* is the set of PSCFG derivations. In the next section we provide some motivation for work we carried out in this thesis.

3.2 Motivation

Target phrase selection, a crucial component of the state-of-the-art SMT models, plays a key role in generating accurate translation hypotheses. The phrase-based SMT model of Koehn et al. (2003) and syntax-based SMT model of Chiang (2007) possess a common weakness in the lexical selection model since both the SMT models select target equivalents for a source phrase only on the basis of the source phrase itself, regardless of its contexts, and the target language model. The target language model does represent contextual information, but only (clearly) of the target language. In other words, the sense-disambiguation tasks inherent to both the phrase-based and syntax-based models are modelled suboptimally as they ignore

source-side contexts when translating a source phrase. We argue that the disambiguation of a source phrase can be enhanced by taking into account the contexts of the source phrase.

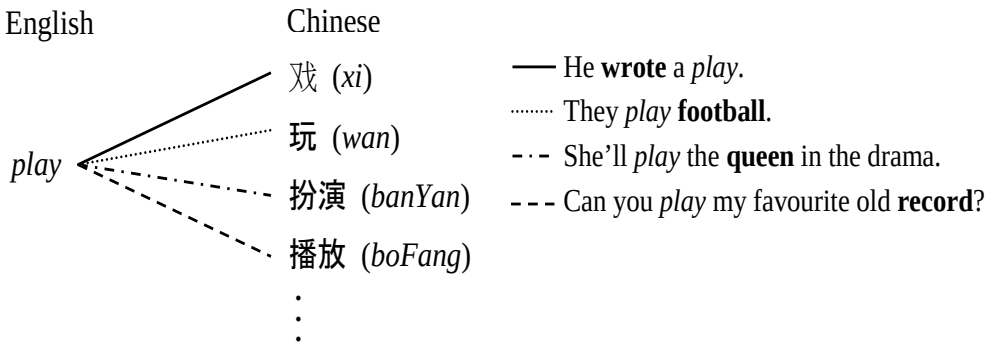

Figure 3.1: Examples of ambiguity for the English word '*play*', together with different translations depending on the contexts.

Figure 3.1 shows translation examples for a highly ambiguous English word '*play*'. The ambiguous word '*play*' has many translation equivalents in Chinese, some ('*xi*', '*wan*', '*banYan*', '*boFang*') of which are shown in Figure 3.1. Figure 3.1 also shows four English sentences, each containing the ambiguous word '*play*' which is translated into four different Chinese words depending on the context in which it appears. For example, the most suitable translation for the word '*play*' in the first English sentence '*He wrote a play*' is '*xi*' amongst the four Chinese candidate translations. The translation of '*play*' in this sentence depends on the neighbouring word '*wrote*'. Similarly, the appropriate translation for '*play*' in the English sentence '*Can you play my favourite old record?*' is '*boFang*' amongst the four Chinese candidate translations. In this English sentence, the translation of '*play*' depends on the distant word '*record*'. In addition to improving lexical selection, contexts in which source phrases appear improve ordering of candidate phrases in the target sentence. In other words, the exact target positions of the translations of source phrases may depend on its contextual dependencies in the source sentence.

Like the single-word phrase '*play*', translations of a multi-word ambiguous phrase depend on the context in which that phrase appears. We carried out a manual analysis to see how frequently ambiguous words occur in a text. We looked at

the English sentences in the development set of the Dutch-to-English data set (cf. Section 3.6). We observed that about 64% sentences of the development set contain at least one ambiguous word. We also found that 10.8% words of the development text are highly ambiguous. However, a sentence that a human reader may not regard as ambiguous can be interpreted as ambiguous by a MT system (Hutchins and Somers, 1992, page 88–89). Hutchins and Somers (1992) pointed out that it is very difficult for a MT system to recognize accidental structural ambiguities unless contextual 'knowledge' is provided to the system.

Translation of a source sentence in the PB-SMT and HPB-SMT models begins by generating all possible source phrases and gathering all candidate phrases for each source phrase. In the translation process, thousands of translation hypotheses (chart entries in Hiero model) are statistically generated using the pool of all target phrases to form the candidate translations. Thus, the decoder considers all target phrases for a given source phrase as possible candidate translations of that source phrase. The candidate phrases with higher translation probabilities have a better chance in occurring in the most likely candidate translations.[2] On the other hand, the candidate phrases with lower translation probabilities have a higher risk of being pruned out during the formation of translation hypotheses due to the decoder's beam size limit. Interestingly, the phrase translation probabilities are simply measured based on the frequency of occurrences of the source and target phrase pairs in the training corpus, while completely ignoring the contexts in which those phrases appear.

Let us go back to the example of the ambiguous word '*play*' in Figure 3.1. Let us assume that the translation probability of '*play*' into '*xi*' is higher than that of '*play*' into any of the remaining Chinese words ('*wan*', '*banYan*', '*boFang*'). During translation of any of the English sentences in Figure 3.1, the decoder would ignore whatever contextual dependency the source phrase '*play*' has in the source

[2]In addition to the translation models, the target language model also controls the selection of candidate phrases and thereby generation of most likely target sentences.

sentence, consider all Chinese words as probable candidate translations, and always give preference to '*xi*' in order to form the best candidate translations (best chart entries in Hiero).

In particular, the language model plays an important role in the selection of the candidate phrases. In addition to the language model, in our work we incorporate a range of source-language context features into the state-of-the-art PB-SMT (Koehn et al., 2003) and HPB-SMT (Chiang, 2007) models in order to perform discriminative translation filtering (cf. Section 2.2.1.2) by learning context-sensitive translation probabilities which in effect should improve target phrase selection. We see from Figure 3.1 that the translations of '*play*' may depend on the neighbouring lexical context (e.g. '*wrote*' in the first example sentence) as well as quite distant lexical context (e.g. '*record*' in the last example sentence). We investigate the incorporation of basic contextual features (words and POS tags) (cf. Chapter 4), lexical syntactic descriptions (supertags) (cf. Chapter 5), deep syntactic information (grammatical dependency relations) and semantic roles (cf. Chapter 6) into the SMT models. We conjecture that such kinds of complex and rich syntactic and semantic knowledge sources, some of which inherently capture long-distance word-to-word dependencies in a sentence, can be useful in improving lexical selection in PB-SMT and HPB-SMT.

3.3 Context-Informed SMT models

We conjecture that context-dependent phrase translation can be expressed as a multi-class classification problem, where a source phrase with given additional context information is classified into a distribution over possible target phrases. The size of this distribution is possibly limited, and would ideally omit improbable or irrelevant target phrase translations that the standard PB-SMT or HPB-SMT approach would normally include. In the subsequent sections, we define our context-informed PB-SMT and HPB-SMT models formally with mathematical derivations.

3.3.1 Context-Informed PB-SMT

A context-informed feature $\hat{h}_{\text{mbl}}$[3] can be viewed as the conditional probability of the target phrases $\hat{e}_k$ given the source phrase $\hat{f}_k$ and its context information (CI), as in (3.7):

$$\hat{h}_{\text{mbl}} = \log \text{P}(\hat{e}_k|\hat{f}_k, \text{CI}(\hat{f}_k)) \tag{3.7}$$

Here, CI may include any feature (e.g. lexical, syntactic, semantic), which can provide useful information to disambiguate the given source phrase. In addition to $\hat{h}_{\text{mbl}}$, we derive a simple two-valued feature $\hat{h}_{\text{best}}$, defined as in (3.8):

$$\hat{h}_{\text{best}} = \begin{cases} 1 & \text{if } \hat{e}_k \text{ maximizes P}(\hat{e}_k|\hat{f}_k, \text{CI}(\hat{f}_k)) \\ \approx 0 & \text{otherwise} \end{cases} \tag{3.8}$$

where $\hat{h}_{\text{best}}$ is set to 1 when $\hat{e}_k$ is one of the target phrases with highest probability according to $\text{P}(\hat{e}_k|\hat{f}_k, \text{CI}(\hat{e}_k))$; otherwise, $\hat{h}_{\text{best}}$ is set to a very low non-zero value (0.000001).

We performed experiments by integrating these two features $\hat{h}_{\text{mbl}}$ and $\hat{h}_{\text{best}}$ directly into the log-linear PB-SMT model (Koehn et al., 2003). Their weights are optimized using minimum error-rate training (MERT) (Och, 2003) on a held-out development set.

3.3.2 Context-Informed HPB-SMT

As mentioned earlier, dependencies between consecutive source phrases (α) are not directly expressed in HPB-SMT as is the case in PB-SMT. A context-informed feature ϕ_{mbl} in Hiero can be viewed as the conditional probability of the target phrases γ given the source phrase α and its context information (CI), which is expressed as in (3.9):

[3]The term 'mbl' stands for memory-based learning (cf. Section 3.4).

$$\phi_{\text{mbl}}(\alpha, \gamma) = \log \text{P}(\gamma|\alpha, \text{CI}(\alpha)) \tag{3.9}$$

In addition to ϕ_{mbl}, we derive a simple binary feature ϕ_{best}, defined as in (3.10):

$$\phi_{\text{best}} = \begin{cases} 1 & \text{if } \gamma \text{ maximizes } \text{P}(\gamma|\alpha, \text{CI}(\alpha)) \\ \approx 0 & \text{otherwise} \end{cases} \tag{3.10}$$

where ϕ_{best} is set to 1 when γ is one of the target phrases with the highest probability according to $\text{P}(\gamma|\alpha, \text{CI}(\alpha))$; otherwise, ϕ_{best} is set to 0.000001.

We performed experiments by integrating these two features ϕ_{mbl} and ϕ_{best} directly into the log-linear model of Hiero. Feature weights are optimized using MERT (Och, 2003) on a development set.

Figure 3.2 display a diagram of the context-sensitive SMT framework. In addition to the language model and the traditional translation model, SMT decoder makes use of our context-sensitive translation models in order to improve target phrase selection during decoding.

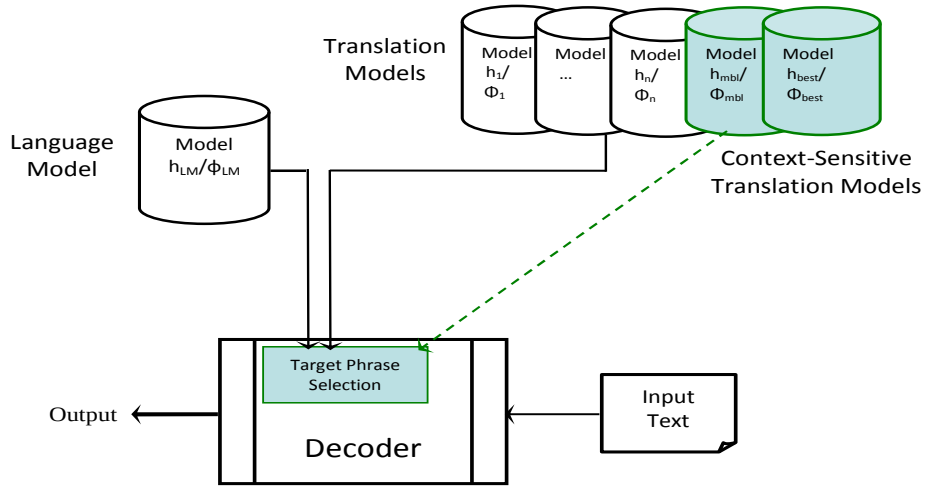

Figure 3.2: Context-sensitive translation models inside the log-linear SMT framework.

3.4 Memory-Based Classification

Stroppa et al. (2007) pointed out that directly estimating context-dependent phrase translation probabilities using relative frequencies is problematic. Indeed, Zens and Ney (2004) showed that the estimation of phrase translation probability (i.e. $\mathrm{P}(\hat{e}_k|\hat{f}_k)$ or $\mathrm{P}(\gamma|\alpha)$) using relative frequencies results in overestimation of the probabilities of long phrases; consequently, smoothing factors in the form of lexical-based features are often used to counteract this bias (Foster et al., 2006). In the case of context-informed features, this estimation problem can only become worse.

As an alternative, we make use of memory-based machine learning classifiers that are able to estimate context-dependent phrase translation probabilities (i.e. $\mathrm{P}(\hat{e}_k|\hat{f}_k, \mathrm{CI}(\hat{f}_k))$ in PB-SMT or $\mathrm{P}(\gamma|\alpha, \mathrm{CI}(\alpha))$ in Hiero) by similarity-based reasoning over memorized nearest-neighbour examples of source–target phrase translations to a new source phrase to be translated. In this work, we use three memory-based classifiers: IB1, IGTree and TRIBL[4] (Daelemans and van den Bosch, 2005). Both IGTree and TRIBL algorithms approximate the unabridged (and computationally expensive) memory-based IB1 (k-nearest neighbour (k-NN)) classifier (Aha et al., 1991). Some interesting properties of such classifiers include: (a) training can be performed efficiently, even with millions of examples, (b) any number of output classes can be handled, (c) the output can be seen as a posterior distribution. In the next three subsections we give a detailed account of description respectively on IB1, IGTree and TRIBL.

3.4.1 Unabridged Memory-Based Classification: IB1

Aha et al. (1991) introduced the IB1 algorithm as an implementation of the k-nearest neighbour classifier. The major difference between the IB1 algorithm originally proposed by Aha et al. (1991) and the TiMBL version is that the value of k refers to k-nearest *distances* rather than k-nearest *examples* in the Tilburg memory-based

[4]An implementation of IB1, IGTree and TRIBL is freely available as part of the TiMBL software package, which can be downloaded from http://ilk.uvt.nl/timbl

learner (TiMBL) version.

3.4.2 Fast Approximate Memory-Based Classification: IGTree

IGTree makes a heuristic approximation of k-NN search (Aha et al., 1991) by storing examples of source-target translation instances in the form of lossless-compressed decision trees, and performing a top-down traversal of this tree (Daelemans et al., 1997a). As a normal k-NN classifier, IGTree retains the labeling information of all training examples, but in a compressed form. In our case, a labeled example is a fixed-length feature-value vector representing the source phrase (as an atomic feature: both single-word and multi-word source phrases are treated as concatenated single values, just as its POS tags or supertags are) and its contextual information, associated with a symbolic class label representing the associated target phrase found through an alignment procedure. A weighting metric such as information gain (IG) is used to determine the order in which features are tested in the tree (Daelemans et al., 1997a). Prediction in IGTree is a straightforward traversal of the decision tree from the root node down, where a step is triggered by a match between a feature value of the new example and an arc fanning out of the current node. When a step ends in a leaf node, the homogeneous class (i.e. a single phrase translation) stored at that node is returned; when no match is found with an arc fanning out of the current node, the distribution of possible class labels at the current node is returned; in our case, a weighted distribution of target phrase translations, where the weights denote the counts in the subset of the training set represented at the current node. The source phrase itself is intuitively the feature with the highest prediction power; it should take precedence in the similarity-based reasoning, and indeed it does, as it always receives the highest IG value. In case of an input that mismatches on the source phrase, the overall target phrase distribution in the training set is returned. We refer the interested reader to (Stroppa et al., 2007) for details of how the IGTree has been used for classifying source phrases with additional contextual information.

3.4.3 A Hybrid Between IB1 and IGTree: TRIBL

TRIBL, which stands for Tree-based approximation of Instance-Based Learning, a hybrid combination of IGTree and unabridged k-NN classification, performs heuristic approximate nearest neighbour search (Daelemans et al., 1997b). A parameter n determines the switching point in the feature ordering from IGTree to normal k-NN classification. The TRIBL approximation performs an initial decision-tree split of the database of training examples on the n most informative features, like IGTree would. Throughout our experiments we set $n = 1$; thus, we split on the values of the single most informative feature, again by computing the information gain (IG) to determine the ordering. During classification, after sub-selecting training examples matching on the most informative feature, the nearest-neighbour distance function is applied to the remaining features (weighted by their IG) to arrive at the set of nearest-neighbours. In other words, TRIBL with $n = 1$ effectively creates individual k-NN classifiers per source phrase, where all classifiers share the same feature weights. When predicting a target phrase given a source phrase and its context information, the identity of the source phrase is (also intuitively) the feature with the highest prediction power. This implies that nearest neighbours always match on the source phrase, and are most similar (preferably, identical) with respect to their contextual features.

A parameter k determines the k closest radii of distances around the source phrase that encompass the nearest neighbours; then, the distribution of target phrases associated with these nearest neighbours is taken as the output of the classification step. The contribution of a single nearest neighbour in this set can be weighted by its distance to the source phrase to be translated, e.g. by assigning higher weights to closer neighbours. In our experiments, we empirically set the value of k, and use exponential decay for the distance-weighted class voting (Daelemans and van den Bosch, 2005). Choosing the optimal setting of k can be handled empirically, as are other hyperparameter settings for the k-NN part of TRIBL. We used a heuristic-automated hyperparameter estimation method based on wrapped

progressive sampling (Van den Bosch, 2004)[5] throughout our experiments.

A TRIBL classification produces a class distribution derived from the aggregate distance-weighted class voting generated by all found nearest neighbours, from which we estimate context-dependent phrase translation probabilities (i.e. $\mathrm{P}(\hat{e}_k|\hat{f}_k, \mathrm{CI}(\hat{f}_k))$ in PB-SMT or $\mathrm{P}(\gamma|\alpha, \mathrm{CI}(\alpha))$ in Hiero) for all possible target phrases (i.e. $\hat{e}_k$ or γ). By normalizing the class votes generated by TRIBL, we obtain the posterior probability distributions we are interested in. We refer the reader to Section 5.3.7 where we give an example to illustrate how a source phrase with additional contextual information is classified into a distribution over possible target phrases using the memory-based classifier.

In this regard, in order to build the set of examples required to train the classifier, we modify the standard phrase-extraction methods (Koehn et al., 2003; Chiang, 2007) to extract the context of the source phrases at the same time as the phrases themselves. Importantly, therefore, the context extraction comes at no extra cost.

3.4.4 Efficiency of Classification Algorithms

In the trade-off between generalization accuracy and efficiency, IB1 usually leads to improved accuracy at the cost of more memory and slower computation than IGTree and TRIBL. Although TRIBL is a fast approximation of unrestricted k-NN, it retains its relatively large memory consumption, which becomes hard to handle on current computing machinery when the number of examples is of the order of 10^7 or higher. IGTree is faster than TRIBL in terms of classification speed, but the latter leads to better accuracy than the former. In short, IGTree can be employed in large-scale translation tasks, whereas IB1 and TRIBL can be employed in small-scale translation tasks.

In the experiments reported in this thesis, experiments with small-scale data sets are usually performed with TRIBL; experiments on large-scale data sets are usually performed with IGTree. A small set of experiments with a particular small-scale

[5] http://ilk.uvt.nl/~antalb/paramsearch/

data set[6] are performed with IB1. Since IB1 is more expensive in terms of memory usage and computation than IGTree and TRIBL, we tested this classifier only with a single small-scale data set.

However, TRIBL was reprogrammed recently, and can now efficiently handle large numbers of training examples. This inspired us to deploy TRIBL classifiers for the large-scale translation tasks. We employed the TRIBL classifier in order to carry out *learning curve* experiments reported in Chapters 4, 5, and 6.

3.5 Feature Integration

The output of memory-based k-NN classification is a set of weighted class labels, representing the possible target phrases ($\hat{e}_k$ or γ) given a source phrase ($\hat{f}_k$ or α) and its context information (CI). Once normalized, these weights can be seen as the posterior probabilities of the target phrases ($\hat{e}_k$ or γ) which give access to $\mathrm{P}(\hat{e}_k|\hat{f}_k, \mathrm{CI}(\hat{f}_k))$ or $\mathrm{P}(\gamma|\alpha, \mathrm{CI}(\alpha))$. Thus, from the classifier's output we can derive the log-linear PB-SMT features $\hat{h}_{\mathrm{mbl}}$ and $\hat{h}_{\mathrm{best}}$ defined in Equation (3.7) and Equation (3.8), respectively. Similarly, we can derive the log-linear Hiero features ϕ_{mbl} and ϕ_{best} defined in Equation (3.9) and Equation (3.10), respectively.

In order to carry out the PB-SMT experiments, we used the widely used open-source toolkit Moses.[7] We integrate the context-informed features $\hat{h}_{\mathrm{mbl}}$ and $\hat{h}_{\mathrm{best}}$ directly into the log-linear framework of Moses. As Stroppa et al. (2007) point out, PB-SMT decoders such as Pharaoh (Koehn, 2004a) or Moses (Koehn et al., 2007) rely on a static phrase table, represented as a list of aligned phrases accompanied by several estimated metrics. Since these features do not express the context information in which those phrases occur, no context information is kept in the phrase table, and there is no way to recover this information from the phrase table.

In order to carry out syntax-based experiments, we used a freely available tree-

[6] IWSLT English-to-Chinese (cf. Section 3.6).
[7] http://www.statmt.org/moses/

based decoder moses-chart.[8] We integrate the context-informed features ϕ_{mbl} and ϕ_{best} directly into the log-linear model of Hiero. Like the Moses decoder, moses-chart also rely on a static rule table, represented as a list of aligned phrases accompanied by several estimated metrics; therefore, there is no way to insert the context information into the rule table, or to recover this information from the rule table.

In order to take into account the context-informed features within such decoders, we implemented a calling framework to translate the test set or development set, which we illustrate as follows. Each word appearing in the test set (or, during development, in the development set) is assigned a unique identifier. First we derive the phrase table or rule table from the training data. Subsequently, we generate all possible phrases from the test set. These phrases are then looked up in the phrase table or rule table, and when found, the phrase along with its contextual information is given to the memory-based classifier to be classified. As stated earlier, memory-based classifiers produce target phrase distributions according to the training examples found within the k-nearest distance radii around the source phrase to be classified. We derive target phrase probabilities from this distribution and temporarily insert them into a new phrase or rule table with the original phrase or rule table estimates, to take our feature functions into account directly in the log-linear model. Thus we create an updated phrase table or rule table. Figure 3.3 graphically illustrates the training and decoding processes of the context-sensitive SMT model.

A lexicalized reordering model is used for all the PB-SMT experiments undertaken on the development and test sets. The source phrases in the reordering table are replaced by the sequence of unique identifiers when the new phrase table is created. After replacing all words by their unique identifiers, we perform MERT using our updated phrase table to optimize the feature weights.

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

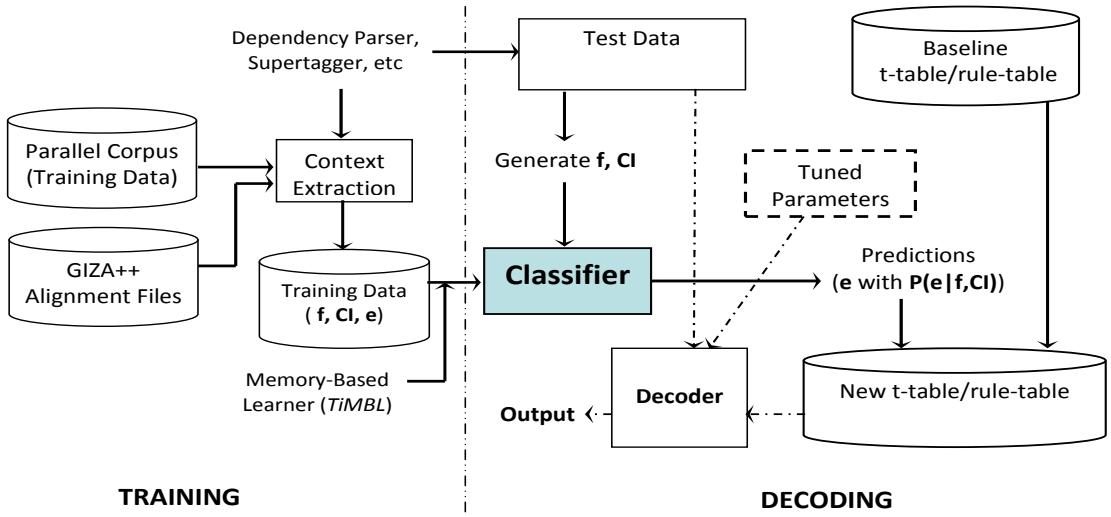

Figure 3.3: Training and Decoding Modules of the Context-Informed SMT Model.

3.6 Data

The various corpora we used for carrying out our experiments are listed in Table 3.1. In Table 3.1, we show statistics of each of data sets with number of sentences, source (S) and target (T) vocabulary size (VS) and average sentence length (ASL). An overview of each data set is given below:

English-to-Chinese (IWSLT 2006): The first set of experiments were carried out on English-to-Chinese training data obtained from the IWSLT 2006 evaluation campaign. The test set (489 sentences) and development set (500 sentences) were chosen from the IWSLT 2002 and IWSLT 2007 evaluation campaigns respectively. This multilingual speech corpus contains sentences similar to those that are usually found in phrasebooks for tourists going abroad.

Dutch-to-English (Open Subtitles): This corpus is collected as part of the Opus collection of freely available parallel corpora (Tiedemann and Nygaard, 2004).[9] The corpus contains user-contributed translations of movie subtitles.

English-to-Hindi (EILMT): This small EILMT tourism domain corpus was released for the shared task on English-to-Hindi SMT (Venkatapathy, 2008).[10]

[9]http://urd.let.rug.nl/tiedeman/OPUS/OpenSubtitles.php
[10]http://ltrc.iiit.ac.in/nlptools2008/index.html

Data Source	Data set	Sentences	VS (S)	VS (T)	ASL (S)	ASL (T)
English-to-Chinese: IWSLT	Train.	40,458	11,358	14,238	9.74	8.77
	Dev.	500	819	862	7.44	6.32
	Eval.	489	846	916	7.61	7.06
Dutch-to-English: Open Subtitles	Train.	286,160	59,863	44,594	7.34	8.85
	Dev.	1,000	1,286	1,228	7.00	8.02
	Eval.	1,000	1,435	1,327	6.67	7.29
English-to-Hindi: EILMT	Train.	6,755	16,344	24,734	24.56	25.97
	Dev.	500	3,689	4.021	24.58	25.99
	Eval.	495	3,598	3,879	24.07	25.51
English-to-Czech: WMT 2010	Train.	94,501	89,672	241,948	20.91	18.68
	Dev.	2051	10,531	14,757	21.18	17.74
	Eval. 1	2525	11,945	17,817	21.64	17.73
	Eval. 2	2489	12,326	13,726	21.70	18.57
Dutch-to-English: Europarl	Train.	1,311,111	247,079	126,141	26.97	27.74
	Dev.	1,000	4,020	3,450	24.81	25.17
	Eval.	1,000	4,312	3,751	24.95	26.15
English-to-Japanese: NTCIR-8	Train.	600,000	193,526	65,934	28.30	29.58
	Dev.	1,814	7,722	4,911	29.36	37.09
	Eval. 1	927	4,264	3,574	28.77	37.08
	Eval. 2	1,119	5,157	3,062	28.51	35.57
English-to-Chinese: NIST-08	Train.	500,000	112,773	128,647	34.38	31.99
	Dev.	1,082	5,097	5,693	33.29	28.12
	Eval.	1,357	2,990	3,320	29.38	24.48
English-to-Spanish: Europarl	Train.	1,639,764	260,099	980,000	24.27	25.96
	Dev.	2,000	8,651	10,371	26.54	27.28
	Eval.	2,000	8,689	10,552	27.15	27.90

Table 3.1: Corpus Statistics. Notation: {S: source, T: target, VS: vocabulary size, ASL: average sentence length}

English-to-Czech (WMT 2010): For the English-to-Czech translation task we employed the News Commentary training data set released in the Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR (WMT-MetricsMATR 2010).[11] To tune the system during development, we used the WMT 2008 test set of 2051 sentences. For evaluation purposes we used two different test sets: the WMT 2009 test set of 2525 sentences, and the WMT 2010 testset of 2489 sentences.

[11] http://www.statmt.org/wmt10/

Dutch-to-English (Europarl): The Dutch-to-English Europarl parallel corpus is extracted from the proceedings of the European Parliament (Koehn, 2005).[12]

English-to-Japanese (NTCIR-8): The experimental data sets were taken from the NTCIR-8 Patent Translation Task.[13] For the purpose of evaluation, we used two different test sets: the first test set contains 927 sentences (henceforth referred to as 'EJTestset1'), and the second test set contains 1,119 sentences ('EJTestset2').

English-to-Chinese (NIST-08): The training set of this English-to-Chinese data set contains sentence pairs of benchmark news text from the NIST Open Machine Translation 2009 Evaluation (MT09).[14] We used the NIST MT05 test set sentences for tuning purpose, and the NIST MT08 'current' test set sentences for evaluation.

English-to-Spanish (Europarl): This training corpus was provided in the Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR (WMT-MetricsMATR 2010).[15] We use the WMT 'test2006' set as the development set, and the WMT 'test2008' set as the test set; both sets contain 2,000 sentence pairs. In this data set, we observed a huge difference between the size of English vocabulary and that of Spanish (Spanish vocabulary size is more than three times larger than that of English).

This thesis reports an wide range of experiments which involves a range of language pairs, a number of domains and varied sizes of data sets, summary of which is graphically illustrated in Figure 3.4.

[12]http://www.statmt.org/europarl/
[13]http://www.cl.cs.titech.ac.jp/~fujii/ntc8patmt/
[14]http://www.itl.nist.gov/iad/mig//tests/mt/2009/
[15]http://www.statmt.org/wmt10/

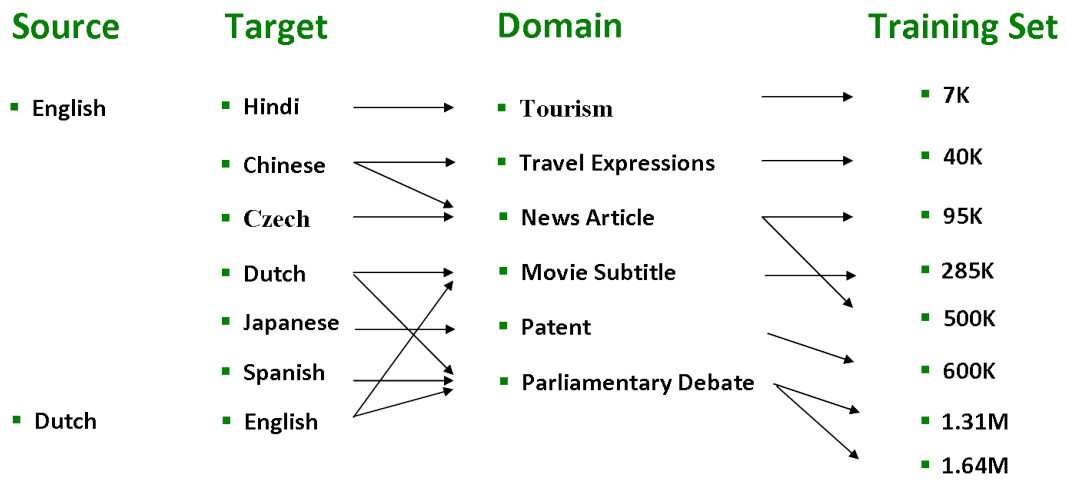

Figure 3.4: Language Pairs, Domain, and Data sets.

3.7 MT Evaluation

Evaluating translated output is the most sensitive step in machine translation since it plays an important role in improving the performance of MT systems. We performed MT evaluation with manual analysis, as well as with automatic metrics.

3.7.1 Manual Evaluation

Manual evaluation, the most reliable method for judging the quality of MT output, is performed on the basis of adequacy and fluency. Adequacy is measured on the basis of how similar the meaning of a translation is compared to its equivalent input source sentence. Fluency is measured on the basis of grammatical well-formedness of a translation. We performed manual qualitative analysis on the basis of adequacy and fluency comparing the output of a subset of our best-performing context-informed SMT systems with those of the respective baseline systems.

3.7.2 Automatic Evaluation

There are many reasons why MT researchers prefer performing evaluations with automatic metrics over manual analysis despite the fact that manual evaluation is the most reliable method to judge translation quality. Manual evaluation is an expensive and time-consuming process, and so cannot be conducted as often as

required during system development. On the contrary, automatic evaluation is very fast and language-independent, and it can be applied repeatedly to the translation output during system development. Therefore, automatic evaluation has become a popular approach in today's MT technology.

We evaluate our MT systems across a wide range of automatic evaluation metrics: BLEU (Papineni et al., 2002), NIST (Doddington, 2002), METEOR (Lavie and Agarwal, 2007), TER (Snover et al., 2006), WER (Levenshtein, 1966), and PER (Tillmann et al., 1997). A general overview of the above mentioned widely used evaluation metrics is given as follows.

BiLingual Evaluation Understudy (BLEU): BLEU is an n-gram precision-based metric. It compares a system's translation output with the reference translations on the basis of occurrences of n-gram word sequences in each pair of candidate and reference translation. BLEU score is measured by summing the logarithm values of 4-gram, trigram, bigram and unigram precisions and multiplying by a weight $\frac{1}{4}$.

NIST: NIST is a variant of the BLEU metric. The NIST metric differs with BLEU in the following respects: (i) firstly, BLEU assigns equal weights to each n-gram pair, while NIST assigns higher weights for more rarely occurring n-gram pairs, (ii) secondly, BLEU is measured over the logarithmic average of n-gram precisions, while NIST is measured over the arithmetic average of n-gram precisions, (iii) finally, BLEU and NIST differ from each other with respect to how they calculate the brevity penalty (used to prevent shorter candidate translations from receiving too high scores).

Metric for Evaluation of Translation with Explicit ORdering (METEOR): The METEOR metric performs evaluation on the basis of matching of candidate and reference translations in three consecutive steps: (i) exact matching of unigrams, (ii) stemmed matching using a Porter stemmer, and (iii) synonymy matching using a Word-Net. We followed the above three steps in order to

evaluate English sentences, whereas only the first set-up was followed in order to evaluate sentences of other languages. Note that METEOR metric supports a limited set of languages (English, Spanish, French, German, and Czech). In order to evaluate translations in other languages, we used METEOR with its default English settings with the first set-up (i.e. exact matching of unigrams).

Translation Edit Rate (TER): TER is an error metric that measures the number of edits (insertion, deletion, substitution, and shift) required to change a candidate translation into one of the reference translations.

Word Error Rate (WER): WER is based on the word-level Levenshtein distance. It measures the distance between the reference and candidate translations based on the number of insertions, substitutions and deletions of words.

Position-Independent Word Error Rate (PER): PER is same as WER except that it allows reordering of words between the candidate translation and the reference translation.

Additionally we performed statistical significance tests using bootstrap resampling (Koehn, 2004b) on BLEU and METEOR. The confidence level (%) of the improvements obtained by the best-performing context-informed systems with respect to the SMT baseline are reported. An improvement in system performance at a confidence level above 95% is assumed to be statistically significant.

3.8 Summary

In this chapter, we first presented the mathematical derivation of two state-of-the-art SMT models: PB-SMT (Koehn et al., 2003) and HPB-SMT (Chiang, 2007). Then, we defined our context-informed SMT models. This chapter also provided an overview of three memory-based classification algorithms: IB1, IGTree and TRIBL (Daelemans and van den Bosch, 2005). Thereafter, we described the approach we

adopted to integrate the contextual features into the log-linear frameworks of PB-SMT and HPB-SMT. Finally, we summarized our experimental data sets in a tabular form, and briefly described the MT evaluation methods used to validate our experiments.

In the next chapter, we will present a series of experiments which we carried out with basic contextual features (words and POS tags). The next chapter also introduces our context-sensitive transliteration model.

Chapter 4

Basic Contextual Features

In this chapter, first we demonstrate how we make use of basic features in order to form the contextual information (CI) of a source phrase. Here, 'basic features' refer to words and part-of-speech (POS) tags. Henceforth, CI with the basic features is referred to as basic contextual information, i.e. 'basic CI'. In Section 4.1, we describe how we form the basic CI of a source phrase in order to derive the log-linear context-informed PB-SMT features. Section 4.2 illustrates how we form the basic CI of a source phrase in order to derive the log-linear context-informed HPB-SMT features. In Section 4.3 and 4.4, we report the experimental results obtained by employing basic contextual features in the PB-SMT and HPB-SMT models, respectively. In Section 4.5, first we describe how machine transliteration, a well-known NLP problem related to machine translation, can be performed with our context-informed PB-SMT model; then, we present the experimental outcomes of our context-informed transliteration systems. We place this discussion here because the transliteration experiments were carried out with the word-based context-informed SMT system.

4.1 Basic Contextual Information for PB-SMT

In the following two subsections, we derive contextual information (CI) of a PB-SMT source phrase with the basic features (i.e. words and POS tags).

4.1.1 Lexical Features

Lexical features include the immediately neighbouring words within l token positions to the left and right (respectively $f_{i_k-l}...f_{i_k-1}$ and $f_{j_k+1}...f_{j_k+l}$) of a given focus phrase $\hat{f}_k = f_{i_k}...f_{j_k}$. Lexical features thus form a window of size $2l$. The lexical contextual information ($\mathrm{CI_{lex}}$) can be described as in (4.1):

$$\mathrm{CI_{lex}}(\hat{f}_k) = \{f_{i_k-l}, ..., f_{i_k-1}, f_{j_k+1}, ..., f_{j_k+l}\} \tag{4.1}$$

We consider the example sentence '*Can you play my favourite old record?*' in Figure 3.1 (on page 40) to illustrate how lexical CI for the single word focus phrase '*play*' is formed. In the example sentence, $\mathrm{CI_{lex}}$ for the focus phrase '*play*' is formed as: $\mathrm{CI_{lex}}$ = {can, you, my, favourite} (with $l = 2$).

4.1.2 Part-of-Speech Tags

In addition to lexical features, it is also possible to exploit other information sources characterizing the context. For example, we may consider the part-of-speech (POS) tags of the context words, as well as of the focus phrase itself. In our model, the POS tag of a multi-word focus phrase is the concatenation of the POS tags of the words composing that phrase. We generate a window of size $2l + 1$ features, including the concatenated complex POS tag of the focus phrase. Accordingly, the POS-based contextual information ($\mathrm{CI_{pos}}$) is described as in (4.2):

$$\mathrm{CI_{pos}}(\hat{f}_k) = \{\mathrm{pos}(f_{i_k-l}), ..., \mathrm{pos}(f_{i_k-1}), \mathrm{pos}(\hat{f}_k), \mathrm{pos}(f_{j_k+1}), ..., \mathrm{pos}(f_{j_k+l})\} \tag{4.2}$$

For the example sentence '*Can you play my favourite old record?*' (Figure 3.1), the POS-based CI for the focus phrase '*play*' is formed as: $\mathrm{CI_{pos}}$ = {pos(*can*), pos(*you*), pos(*play*), pos(*my*), pos(*favourite*)} = {MD, PRP, VB, PRP$, JJ} (with $l = 2$).

We also carried out experiments combining the two feature types (CI_{pos} and CI_{lex}). In order to derive the context-based log-linear feature $\hat{h}_{\text{mbl}}$ (defined in Equation (3.7)) using a memory-based classifier, the lexical- and/or POS-based CI (i.e. CI_{lex}, CI_{pos}, or the combined $\text{CI} = \text{CI}_{\text{lex}} \cup \text{CI}_{\text{pos}}$) for each source phrase in the test or development set is formed during classification. In order to carry out experiments with the basic contextual features, we incorporate the context-informed feature $\hat{h}_{\text{mbl}}$ and a binary feature $\hat{h}_{\text{best}}$ (defined in Equation (3.8)) in the log-linear framework of Moses, and experimental results obtained are reported in Section 4.3.

4.2 Basic Contextual Information for HPB-SMT

In this section, we derive contextual information (CI) of a Hiero source phrase with the basic features (i.e. words and POS tags) as follows.

4.2.1 Lexical Feature

These features include the direct left- and right-neighbouring words of length l (resp. $\text{w}_{\alpha^{-l}}, ..., \text{w}_{\alpha^{-1}}$ and $\text{w}_{\alpha^{+1}}, ..., \text{w}_{\alpha^{+l}}$) of a given source phrase α. In our experiments, we consider the context size up to 2 (i.e. $l := 1, 2$). It also includes boundary words ($\text{w}_{\text{nt}_s^{\text{start}}}$ and $\text{w}_{\text{nt}_s^{\text{end}}}$) of subphrases covered by nonterminals (nt) in the α. Like Chiang (2007), we restrict the number of nonterminals to two (i.e. $s := 2$). The resultant lexical features form a window of size $2(l+s)$ features. Thus, lexical contextual information (CI_{lex}) can be described as in (4.3):

$$\begin{aligned} \text{CI}_{\text{lex}}(\alpha) = \; & \{\text{w}_{\alpha^{-l}}, ..., \text{w}_{\alpha^{-1}}, \text{w}_{\alpha^{+1}}, ..., \text{w}_{\alpha^{+l}}, \\ & \text{w}_{\text{nt}_1^{\text{start}}}, \text{w}_{\text{nt}_1^{\text{end}}}, ..., \text{w}_{\text{nt}_s^{\text{start}}}, \text{w}_{\text{nt}_s^{\text{end}}}\} \end{aligned} \tag{4.3}$$

We consider the example sentence '*Can you play my favourite old record?*' in Figure 3.1 (on page 40) to illustrate how the lexical CI_{lex} feature for the Hiero focus phrase '*play* NT *old*' is formed, where NT is a nonterminal constituting the word sequence '*my favourite*'. The source phrase '*play* NT *old*' contains only one

nonterminal. Following Equation (4.3), we derive context information of that source phrase as: CI_{lex} = {Can, you, record?, ==, my, favourite, ==, ==}. Empty tokens are represented with a special symbol '=='.

4.2.2 Part-of-Speech Tags

We consider the POS of each word in the lexical features in Equation (4.3). POS-based contextual information (CI_{pos}) is described as in (4.4):

$$\text{CI}_{\text{pos}}(\alpha) = \{\text{pos}(w_k)\} \tag{4.4}$$

where $\forall k \in [1, |\text{CI}_{\text{lex}}|] : w_k \in \text{CI}_{\text{lex}}$.

We consider the example sentence '*Can you play my favourite old record?*' and the source phrase '*play* NT *old*' to illustrate how we derive the POS-based CI_{pos} feature for that source phrase. Following Equation (4.4), we define the context information of that source phrase by taking the POS tag of each word occurring in CI_{lex} as: CI_{pos} = {MD, PRP, NN, ==, PRP$, JJ, ==, ==}. POS tags of empty tokens are represented with the special symbol '=='.

We also carried out experiments joining the two feature types (CI_{lex} and CI_{pos}). In order to derive the context-informed log-linear feature ϕ_{mbl} (defined in Equation (3.9)) using a memory-based classifier, the lexical- and/or POS-based CI (i.e. CI_{pos}, CI_{lex}, or the combined $\text{CI} = \text{CI}_{\text{lex}} \cup \text{CI}_{\text{pos}}$) for each source phrase in the test or development set is formed during classification. In order to conduct experiments with the basic contextual features, we incorporate the memory-based context-informed feature ϕ_{mbl} and a binary feature ϕ_{best} (defined in Equation (3.10)) in the log-linear framework of Hiero, and the experimental results obtained are reported in Section 4.4.

4.3 Experiments with Context-Informed PB-SMT

We carried out experiments by applying lexical- and POS-based features systematically for different language pairs with varying training data sizes. The system outputs are evaluated across a wide range of automatic evaluation metrics, which we described in Section 3.7.

We divide the reports on our experiments into six subsections. Section 4.3.1 reports on small-scale data sets representing the language pairs English-to-Chinese, Dutch-to-English, English-to-Hindi, and English-to-Czech, with less than 300,000 training sentences. Section 4.3.2 reports on large-scale data sets with more than 500,000 training sentences, representing the language pairs Dutch-to-English, English-to-Dutch, and English-to-Japanese. Section 4.3.3 provides some analysis of the results obtained from the small- and large-scale translations. In Section 4.3.4, we compare the effectiveness of the basic contextual features with regard to different source and target languages. In Section 4.3.5, we present the results of the learning curve experiments we carried out on three different language pairs: English-to-Spanish, Dutch-to-English, and English-to-Dutch. Section 4.3.6 analyzes the results of the learning curve experiments.[1]

4.3.1 Experiments on Small-Scale Data Sets

4.3.1.1 English-to-Chinese

The first set of experiments were carried out on English-to-Chinese data provided by the IWSLT evaluation campaign (Haque et al., 2009a).[2] (see data set details in Section 3.6).

- **An additional Log-Linear Feature:** In this translation task, we derive an additional log-linear feature ($\hat{h}_{\text{mod}}$)[3] only for this data set, as in (4.5):

[1]Parts of the experiments carried out with our context-informed PB-SMT model including learning curve experiments have been reported, albeit in a different form, in Haque et al. (2011).

[2]Experiments reported in this section have been published in (Haque et al., 2009a).

[3]Here, 'mod' stands for 'modified'.

$$\hat{h}_{\text{mod}} = \log\ [\delta \text{P}(\hat{e}_k|\hat{f}_k, \text{CI}(\hat{f}_k)) + (1-\delta)\text{P}(\hat{e}_k|\hat{f}_k)] \tag{4.5}$$

In Equation (4.5), the log-linear feature $\hat{h}_{\text{mod}}$ is derived by interpolating the memory-based context-dependent phrase translation probability $\text{P}(\hat{e}_k|\hat{f}_k, \text{CI}(\hat{f}_k))$ with the baseline forward phrase translation probability $\text{P}(\hat{e}_k|\hat{f}_k)$ with respect to a weight δ.[4] However, we avoid adding the feature ($\hat{h}_{\text{mod}}$) for conducting experiments on the remaining data sets (cf. Section 3.6) for two reasons: (i) this feature, being influenced by other features, is not fully context-sensitive,[5] and (ii) the interpolation weight α is tuned manually on the development set, which is very expensive. We conducted a set of experiments for each context type or size by integrating the memory-based features ($\hat{h}_{\text{mod}}$, $\hat{h}_{\text{mbl}}$, and $\hat{h}_{\text{best}}$). In the first experiment, the baseline feature $\log\ \text{P}(\hat{e}_k|\hat{f}_k)$ is directly replaced by $\hat{h}_{\text{mod}}$. In the second experiment, we integrate the memory-based feature $\hat{h}_{\text{mbl}}$ together with the baseline features, keeping all the baseline features unaffected. In the third experiment, both the features $\hat{h}_{\text{mbl}}$ and $\hat{h}_{\text{best}}$ are integrated into the model in the same manner (i.e. keeping all the baseline features unaffected). As for the standard phrase-based approach, feature weights are optimized using MERT (Och, 2003) for each of the experiments we carried out. The best results obtained amongst the set of experiments are reported in the tables.

Note that we adopt the above experimental set-ups only for the English-to-Chinese IWSLT data set. In order to conduct experiments for the remaining data sets (cf. Section 3.6) reported in this thesis, we adopt only the third experimental set-up which adds only *pure* context-sensitive features: $\hat{h}_{\text{mbl}}$ and $\hat{h}_{\text{best}}$ (keeping all the baseline features unaffected).

[4]We observed that memory-based classifiers assigned large weights to more appropriate candidate phrases than less appropriate ones.

[5]Baseline phrase translation probability $\text{P}(\hat{e}_k|\hat{f}_k)$ is fully context-independent since it is measured based on the frequency of occurrences of the source and target phrase pairs in training corpus, while completely ignoring the contexts in which those phrases appear. This probability is exploited in order to form the log-linear feature $\hat{h}_{\text{mod}}$ (cf. Equation (4.5)), so $\hat{h}_{\text{mod}}$ is not fully context-sensitive.

Experiments were performed employing the lexical- and POS-based features both individually and in collaboration. Additionally, we considered varying windows of context size. This set of experiments were carried out with IGTree classifiers.

	BLEU		NIST		WER		PER	
Baseline	*20.56*		*4.67*		*57.82*		*48.99*	
Context Size	±1	±2	±1	±2	±1	±2	±1	±2
POS	21.52	21.70	4.70	4.76	57.87	57.21	49.62	49.10
Word	21.64	21.59	4.77	4.78	57.15	57.41	49.21	48.37
Word+POS	21.77	21.89 (96.2%)	4.78	4.83	56.77	56.51	48.58	48.03

Table 4.1: Experiments with uniform context size using IGTree

The results with uniform context size are shown in Table 4.1, which clearly shows that translation from English-to-Chinese can benefit from the addition of source-language features, as the inclusion of the basic contextual feature easily improves upon the baseline across most of the evaluation metrics. Adding neighbouring words as source-context adds almost a whole BLEU point (5.25% relative increase), and further improvements are to be seen when POS tags (a 1.14 BLEU point improvement; a relative improvement of 5.54%) are used. POS tags, when used as an individual feature, produce the highest improvement over the Moses baseline.

With respect to BLEU, we observed an improvement of 1.33 BLEU points (a 5.94% relative improvement) over the Moses baseline for the combination of Word+POS when a context window of ±2 is used. This experimental set-up appears to be most effective in this translation task.

Experiments	BLEU	NIST	WER	PER
Baseline	*20.56*	*4.67*	*57.82*	*48.99*
Word±2 + POS±1	21.61 (95.8%)	4.77	56.78	48.66

Table 4.2: Experiments with varying context size using IGTree.

The results with varying context size are shown in Table 4.2. Moderate improvements across the all evaluation metrics are to be observed for the combination of POS tags and neighbouring words with a varying context size.

In summary, on the English-to-Chinese translation task, we observed that word and POS contextual features (individually or in collaboration) significantly improves over the PB-SMT baseline.

As an additional point of analysis, we compared our best-performing context-informed (CI) model (Word±2+POS±2) with the Moses baseline in terms of various aspects. We carried out a deeper analysis to see how two systems (Word±2+POS±2 and Moses baseline) differ from each other in terms of automatic evaluation measures (BLEU and TER), closeness to the reference set, and weights assigned to the various translational models. Table 4.3 shows number of sentences in which the CI model produces better, worse or similar translations to those produced by the Moses baseline, as per two sentence-level evaluation measures (TER and BLEU).[6] We also measure closeness of the translations produced by the two systems (Word±2+POS±2 and Moses) with respect to the reference translations. We calculated percentages of sentences and words in the reference set that appear in the set of translations generated by the CI and the Moses systems (see Table 4.3).

The weights (λ_i) of the various log-linear features directly affects the phrase translation scores during translation; thus, λ_i plays a crucial role in selecting the most appropriate candidate phrases. Additionally, Table 4.4 compares weights assigned to the various translational features of the Word±2+POS±2 and the baseline systems obtained by MERT (Och, 2003) training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	24	14	61	387
Sentence-Level TER	85	80	287	34
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	8.45		7.81	
Matching Words (%)	58.54		57.44	

Table 4.3: Comparison between translations produced by the best-performing context-informed (CI) system (Word±2+POS±2) and Moses baseline.

[6]Sentence-level BLUE scores for most of the test set sentences are zero due to the fact that 4-gram word sequence matching are seldom found for short sentences.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mod}
Moses	0.2388	0.0686	0.0076	0.0477	0.0842	0.0507	0.7030	-
Word±2+POS±2	0.1742	0.0426	0.0371	-	0.0266	0.0323	0.1379	0.0316

Table 4.4: Comparison of weights for each translational feature of the two systems (Word±2+POS±2 and Moses baseline) obtained by MERT training. [Notations: lm: language model, btp: backward translation probability, blexp: backward lexical weighting probability, ftp: forward translation probability, flexp: forward lexical weighting probability, phrpty: phrase penalty, wrdpty: word penalty, mod: modified (cf. Equation (4.5))].

4.3.1.2 Dutch-to-English

Dutch-to-English experiments were performed on the Dutch-to-English Open Subtitles corpus (cf. Section 3.6) (Haque et al., 2009b).[7] For the Dutch-to-English translation task, we carried out experiments using the TRIBL classifier. We employed our previously best-performing set-ups obtained from the English-to-Chinese translation task in order to form the basic contextual information of a source phrase. The experimental results are reported in Table 4.5. In all cases, the width of the left and right context is set to 2. In this translation task, an additional experiment (labeled POS±2†)[8] was performed in which the concatenated parts-of-speech of the focus phrases were not included as a feature. As can be observed from Table 4.5, the POS±2† experiment produces the best improvements (0.90 BLEU points; 2.78% relative) over the baseline. However, the improvement is not statistically significant.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*32.39*	*6.11*	*55.39*	*50.15*	*49.67*	*43.12*
Word±2	32.48	6.11	55.72	50.40	50.43	42.91
POS±2	33.07	6.13	56.17	50.07	49.38	42.85
POS±2†	33.29 (74%)	6.17	55.72	49.56	48.91	42.77
Word±2+POS±2	32.59	6.09	55.36	50.11	49.63	43.10

Table 4.5: Experiments with words and parts-of-speech as contextual features.

In this translation task, we also carried out an analysis on the translations produced by our best-performing context-informed (CI) system (POS±2†) and the Moses baseline. Table 4.6 shows how two systems differ from each other in terms of

[7] Experiments reported in this section have been published in Haque et al. (2009b).

[8] Signalling one particular exception, we use the *dag*(†) symbol for experiments in which syntactic information of the focus phrase is ignored.

sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 4.7 compares weights of the various translational features of the POS±2† and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	66	45	252	637
Sentence-Level TER	118	77	643	162
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	16.9		17.2	
Matching Words (%)	61.86		61.72	

Table 4.6: Comparison between translations produced by the best-performing context-informed (CI) system (POS±2†) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1065	0.0583	0.0078	0.0579	0.0650	0.0998	-0.2357	-	-
POS±2†	0.1083	0.0225	0.0276	0.0616	0.0899	0.0794	-0.3419	0.0553	0.0014

Table 4.7: Comparison of weights for each translational feature of the two systems (POS±2† and Moses baseline) obtained by MERT training.

4.3.1.3 English-to-Hindi

For English-to-Hindi translation we conducted experiments using the relatively small EILMT tourism corpus (cf. Section 3.6). Like other Indian languages, Hindi is a relatively free word order language. Experiments were carried out with the TRIBL classifier. The experimental results are displayed in Table 4.8.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*10.93*	*4.54*	*28.59*	*74.87*	*82.06*	*56.67*
Word±1	10.76	4.53	28.27	75.42	82.75	57.26
Word±2	11.24 (85%)	4.58	28.27	74.89	82.18	56.45
POS±1	10.82	4.55	28.59	74.76	81.85	56.65
POS±2	11.00	4.55	28.90	74.67	81.89	56.72
Word±2+POS±2	10.98	4.54	28.90	74.84	81.99	56.61

Table 4.8: Experiments applying basic contextual features in English-to-Hindi translation.

Experimental results in Table 4.8 show that the Word±2 produces the best improvement (0.31 BLEU points, 2.84% relative increase) over the baseline. This improvement is not statistically significant, although close to the statistical significance

level. The POS±2 and Word±2+POS±2 improve on the Moses baseline; nevertheless, the improvements with respect to the baseline are small and not statistically significant. The other evaluation metrics tend to show improvements almost similar to those observed on BLEU. In summary, neighbouring words as an individual feature seem to provide the most effective source-language context in this translation task.

We also carried out an analysis on the translations produced by our best-performing context-informed (CI) system (Word±2) and the Moses baseline. Table 4.9 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 4.10 compares weights of the various translational features of the Word±2 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	79	55	20	341
Sentence-Level TER	161	168	166	0
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	0		0	
Matching Words (%)	51.72		51.86	

Table 4.9: Comparison between translations produced by the best-performing context-informed (CI) system (Word±2) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0980	0.0453	0.0652	0.0510	0.1948	0.0910	-0.1683	-	-
Word±2	0.0584	0.0484	0.0143	0.0196	0.099	0.0400	-0.2395	0.0161	0.0321

Table 4.10: Comparison of weights for each translational feature of the two systems (Word±2 and Moses baseline) obtained by MERT training.

4.3.1.4 English-to-Czech

For the English-to-Czech translation task (Penkale et al., 2010)[9] we employed the TRIBL classifier and the previously best-performing set-up in terms of context width and feature combinations. The evaluation results on the WMT 2009 test set are

[9]Parts of experimental results in the English-to-Czech translation task have been summarized in Penkale et al. (2010) which describes the DCU machine translation system in the evaluation campaign of the Joint Fifth Workshop on Statistical Machine Translation and Metrics in ACL-2010.

reported in Table 4.11.[10] As can be seen from Table 4.11, none of the contextual features cause improvements over the Moses baseline on BLEU. However, we achieve a small improvement in terms of METEOR over the Moses baseline for the POS±2.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*7.83*	*3.90*	*34.13*	*87.66*	*80.53*	*67.88*
POS±2	7.80	3.90	34.25	87.87	80.84	67.95
Word±2	7.50	3.83	33.84	88.73	81.68	68.67

Table 4.11: Experimental results on WMT 2009 test set.

Experimental results on the WMT 2010 test set are shown in Table 4.12. The POS±2 and the Word±2 do not cause any improvements in any of the evaluation metrics.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*8.05*	*3.97*	*34.61*	*86.01*	*78.54*	*67.48*
POS±2	7.91	3.94	34.57	86.50	79.03	67.84
Word±2	7.57	3.88	34.16	87.13	79.77	68.39

Table 4.12: Experimental results on WMT 2010 test set.

We also carried out an analysis on the translations produced by the POS±2 and the Moses baseline (with WMT 2010 test set). Table 4.13 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 4.14 compares weights of the various translational features of the POS±2 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	188	202	101	1998
Sentence-Level TER	545	658	1299	7
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	0.28		0.28	
Matching Words (%)	30.69		30.73	

Table 4.13: Comparison between translations produced by the context-informed (CI) system (POS±2) and the Moses baseline.

[10]The English-to-Czech baseline system produced a very low BLEU score (7.83 BLUE points) because: (i) firstly, morphologically-rich languages (like Czech) are more difficult to translate into than from, and (ii) secondly, Czech is a free word-order language; only one set of reference translations is not sufficient for evaluating a free word-order language.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1017	0.0405	0.0265	0.0550	0.0222	0.2377	-0.1147	-	-
POS±2	0.1586	0.1616	0.0902	0.0095	0.0269	0.0920	0.0048	0.0048	0.0013

Table 4.14: Comparison of weights for each translational feature of the two systems (POS±2 and Moses baseline) obtained by MERT training.

4.3.2 Experiments on Large-Scale Data Sets

4.3.2.1 Dutch-to-English

To explore the question whether similar improvements to the ones obtained on small-scale data sets reported in the previous subsection can be achieved with large-scale data sets, we carried out a similar series of experiments. Our first experimental data set is Dutch-to-English Europarl data (cf. Section 3.6). Analogous to the experiments on small-scale data sets, we experimented with adding contextual features representing words, part-of-speech tags, and their combinations. We used the IGTree classifier to carry out these experiments, as TRIBL's memory requirements become too demanding with data sets of this size.[11]

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*27.29*	*6.686*	*56.81*	*58.65*	*63.97*	*45.18*
Word±2	27.13	6.66	56.78	59.1	64.44	45.41
POS±2	26.93	6.67	56.51	59.06	64.19	45.51
POS±2†	26.90	6.67	56.61	58.94	64.10	45.52

Table 4.15: Results on Dutch-to-English Translation considering words and part-of-speech tags as contexts.

We used the same experimental settings as used with the small-scale Open Subtitles data set reported in Section 4.3.1.2. Experimental results are reported in Table 4.15, where we can see that word and part-of-speech contexts are unable to yield any improvement over the Moses baseline on any of the evaluation metrics.

Additionally, we carried out an analysis on the translations produced by the Word±2 and the Moses baseline. Table 4.16 shows how two systems differ from

[11]For example, the memory structure built by TRIBL takes about 90 GB primary memory when trained on a training set of 70 million instances generated on the Dutch-to-English training data set containing 1.3 million sentence pairs, when only the Word±2 features are included. Additionally, the TRIBL classifier leads to very slow processing speed with large-scale data.

each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Table 4.17 compares weights of the various translational features of the Word±2 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	270	265	185	280
Sentence-Level TER	266	302	394	38
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	4.0		4.2	
Matching Words (%)	65.39		65.23	

Table 4.16: Comparison between translations produced by the context-informed (CI) system (Word±2) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1086	0.0641	0.0107	0.0569	0.0867	0.0977	-0.2459	-	-
Word±2	0.1015	0.0529	0.0237	0.0042	0.0757	0.0404	-0.2418	0.0118	0.0102

Table 4.17: Comparison of weights for each translational feature of the two systems (Word±2 and Moses baseline) obtained by MERT training.

4.3.2.2 English-to-Dutch

English-to-Dutch translation were carried out on the same Dutch-to-English Europarl data set described in Section 4.3.2.1, but in the reverse direction. Experimental results for basic contextual features are displayed in Table 4.18. As can be seen from the table, the Word±2 yields a small (statistically insignificant) BLEU improvement (0.24 BLEU points, 0.98% relative increase) over the Moses baseline, whereas, the POS±2 is unable to yield any improvement.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*24.26*	*6.177*	*52.68*	*64.37*	*68.81*	*50.02*
Word±2	24.50 (80%)	6.248	52.78	63.96	68.46	49.59
POS±2	24.04	6.150	52.17	64.44	68.69	50.1

Table 4.18: Results on English-to-Dutch Translation employing words and part-of-speech features.

In this translation task, we also carried out an analysis on the translations produced by the Word±2 and the Moses baseline. Table 4.19 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU

and TER) and closeness to the reference set. Table 4.20 compares weights of the various translational features of the Word±2 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	262	267	154	317
Sentence-Level TER	321	290	351	38
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	3.9		3.8	
Matching Words (%)	61.99		61.74	

Table 4.19: Comparison between translations produced by the context-informed (CI) system (Word±2) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1072	0.0102	0.0509	0.1103	0.0468	0.0936	-0.2689	-	-
Word±2	0.0781	0.0072	0.0642	0.0150	0.0522	0.0284	-0.2124	0.0567	0.0001

Table 4.20: Comparison of weights for each translational feature of the two systems (Word±2 and Moses baseline) obtained by MERT training.

4.3.2.3 English-to-Japanese

Our next sets of experiments were carried out on a large-scale English-to-Japanese data set (cf. Section 3.6) using the IGTree classifier. Experimental results are shown in Table 4.21. None of the contextual features are able to improve on the Moses baseline with any of the test sets. The NTCIR data has been reported to be very noisy (Okita et al., 2010), which might affect the results.[12]

Experiments	BLEU	NIST	TER	WER	PER
Baseline	*27.30*	*6.746*	*63.31*	*80.01*	*43.36*
Evaluation Results on EJTestset1					
POS±2	27.03	6.728	64.03	80.85	43.67
Word±2	26.65	6.656	64.18	80.20	43.83
Evaluation Results on EJTestset2					
Baseline	*27.76*	*6.838*	*60.64*	*77.49*	*42.61*
POS±2	27.39	6.744	61.65	78.79	43.18
Word±2	27.15	6.752	61.53	78.25	43.13

Table 4.21: Experimental results for large-scale English-to-Japanese translation.

[12]The contents in this section have been published, albeit in a different form, in Okita et al. (2010).

As shown in the previous sections as well as in this section, a large set of experimental results were obtained from four small-scale and three large-scale MT set-ups. In the following two sections, we provide some analysis of these results, along several different indices: the effectiveness of different context types, classification approaches, the nature of the data sets, divergence of translation pairs, and directionality of translation.

We also carried out an analysis on the translations produced by the POS±2 and the Moses baseline (with EJTestset2). Table 4.22 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 4.23 compares weights of the various translational features of the POS±2 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	313	424	190	0
Sentence-Level TER	333	436	42	116
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	0		0	
Matching Words (%)	69.15		69.86	

Table 4.22: Comparison between translations produced by the context-informed (CI) system (POS±2) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0739	0.0323	0.0292	0.0558	0.0418	0.0739	-0.1977	-	-
POS±2	0.0765	0.0184	0.0250	0.0432	0.0230	0.0127	-0.1860	0.0024	-0.0042

Table 4.23: Comparison of weights for each translational feature of the two systems (POS±2 and Moses baseline) obtained by MERT training.

4.3.3 Effect of Small vs Large-Scale Data Sets

4.3.3.1 Small-scale data sets

On small-scale data sets, little difference in translation quality is to be seen among the various context-informed models. On IWSLT English-to-Chinese translation, any differences in automatic evaluation scores are not statistically significant for the POS tag and word contexts. Nonetheless, the highest evaluation scores over the

baseline are achieved employing their combinations; indeed, most of the improvements over the baseline are statistically significant in terms of BLEU. Note that the IGTree classifier was employed for the English-to-Chinese translation task.

The small-scale Dutch-to-English translation task showed that the POS contextual features produced the biggest improvement over the baseline. For English-to-Hindi, the word contextual model produces the biggest improvement over the baseline in terms of BLEU when we look at individual features, although the improvement is not statistically significant. For English-to-Czech, POS and word contexts do not provide any improvement over the baseline at all. For these latter three language pairs, the TRIBL classifier was used.

4.3.3.2 Large-scale data sets

While the results on small-scale data sets were rather mixed, the results of the large-scale translation tasks provide a clearer overall picture. Note, however, that we had to work with IGTree classifiers to handle the large set of examples in these experiments.

For large-scale Dutch-to-English translation, word- and POS-based models do not show any improvement over a baseline PB-SMT model. For the reverse language direction, improvements for the word contextual model are not statistically significant, and the POS-based model performs below the baseline PB-SMT model. For large-scale English-to-Japanese translation, none of the contextual features showed any improvement over the baseline.

Comparing the effectiveness of the classifiers on large-scale translation tasks, IGTree proved useful for English-to-Dutch, but not for Dutch-to-English and English-to-Japanese. In contrast, both IGTree and TRIBL were effective for small-scale translation tasks.

4.3.4 Effect of Different Source and Target Languages

As stated earlier, this chapter models basic features (words and POS tags) as a source-language context in the state-of-the-art SMT models. Later in this thesis (i.e. Chapters 5 and 6), we will see that our research focuses on incorporating mainly rich and complex syntactic knowledge sources (i.e. supertags (Bangalore and Joshi, 1999), grammatical dependency relations (Nivre, 2005), and semantic roles (Carreras and Márquez, 2004)) into the state-of-the-art SMT systems in order to improve lexical selection. We considered English as the source language in all our translation tasks except for Dutch-to-English translation, owing to the fact that most of the syntactic tools[13] used in our experiments are readily available only for English. In this section, we examine the role of different source and target languages on our context-informed models.

4.3.4.1 English as target

The results of the small-scale Dutch-to-English translation task (cf. Section 4.3.1) clearly show that none of the improvements over the Moses baseline are statistically significant in terms of BLEU when words and POS features are employed either individually or collectively. In the large-scale Dutch-to-English task (cf. Section 4.3.2), word and POS contexts do not improve the PB-SMT model.

4.3.4.2 English as source

It is well understood that performing MT from English to a morphologically richer target language is inherently more difficult than the other way round. We have investigated the effects of deploying various contextual features for a wide range of target languages, and we compare the results obtained in this section.

For Hindi as the target language, consistent but statistically insignificant improvements over the baseline were observed when words and POS features were

[13]We employed rich and complex syntactic tools (i.e. supertaggers, dependency parsers, and semantic role labeler), which we will introduce in Chapters 5 and 6.

employed in the model. As Hindi is a relatively free word order language, albeit with a preference for SOV word order, we consider our results to be due to the fact that having only one reference translation available per sentence is not sufficient for meaningful evaluation. However, for Czech – another free word order language – we noted that words and POS-based context features do not improve the English-to-Czech PB-SMT system, and here too, only one set of reference translations was available.

For English-to-Japanese patent translation, none of the contextual features was able to beat the performance of the PB-SMT baseline (cf. Section 4.3.2). Note, however, that in addition to providing the largest amount of training data of all our experiments, the NTCIR data was very noisy (Okita et al., 2010), which may have affected the results somewhat.

For English-to-Dutch, neighbouring words improved the PB-SMT model when we employed them as source-language contextual features, while POS-based features did not contribute much.

4.3.5 Experiments on Increasing Size of Training Sets

4.3.5.1 English-to-Spanish

Thus far, using two different approximate memory-based classifiers, different data set sizes, different language pairs, texts from different genres and domains, various source-side context features and context widths, we have obtained mixed results. We observed that the context-informed models tend to perform better than the baseline PB-SMT model on small-scale training data, but the relative gains tend to diminish when we use larger training sets, which may be partly due to the less optimal behaviour of the IGTree classifier. So far, however, we have not systematically varied the amount of training data sizes given a particular data set, to see whether the relative advantage of TRIBL over IGTree changes with the amount of training data available, and how their performance relates to the baseline with varying amounts

of training data available.

In this section we explore a new language pair, English-to-Spanish. We conduct *learning curve* experiments on increasing training data sets while adding an optimized set of contextual features (i.e. best-performing experimental set-ups). We segment the English-to-Spanish data set into several incremental slices of increasing size, and perform a series of experiments on each of these data sets.

To conduct learning curve experiments, we employ the English-to-Spanish Europarl data set (cf. Section 3.6). We segmented the English-to-Spanish training set into eight pseudo-exponentially increasing training sets: 10K, 20K, 50K, 100K, 200K, 500K, 1M, and 1,639,764 training sentences. To perform experiments on this sequence of training sets, we used both IGTree and TRIBL. We were only able to use the TRIBL classifier with training sets containing up to 100K sentences due to TRIBL's relatively high memory requirements.

IGTree as classifier: Experimental results of the English-to-Spanish translation employing IGTree as a classifier are displayed in Table 4.24. As can be seen from the Table 4.24, for every size of training data, the POS±2 and the Word±2 systems always remain below the Moses baseline according to the performance measured by any of the evaluation metrics.

TRIBL as classifier: The experimental results employing TRIBL as a classifier are displayed in Table 4.25. Statistically significant improvements in BLEU and METEOR are to be seen at larger amounts of training data (50K and 100K) when neighbouring words and POS tags are employed as source-side contextual features.

Learning Curves: While we were able to see some broad tendencies in previous sections as well as in Table 4.25, we present here a more analytical study of the effect of increasing amounts of training data with drawing learning curves, for both IGTree and TRIBL classifiers.

Train. Size	Experiments	BLEU	NIST	METEOR	TER	WER	PER
10K	Baseline	*22.68*	*6.00*	*26.15*	*68.93*	*67.58*	*50.81*
	POS±2	21.61	5.85	25.52	69.86	68.67	51.63
	Word±2	21.94	5.90	25.67	69.78	68.36	51.36
20K	Baseline	*24.58*	*6.38*	*27.77*	*66.51*	*65.16*	*48.71*
	POS±2	23.66	6.27	27.26	67.49	66.27	49.40
	Word±2	23.97	6.30	27.35	67.35	66.21	49.26
50K	Baseline	*27.33*	*6.84*	*29.93*	*64.15*	*62.97*	*46.63*
	POS±2	26.34	6.67	29.25	65.07	64.01	47.41
	Word±2	26.58	6.75	29.44	64.66	63.52	46.88
100K	Baseline	*28.64*	*7.09*	*30.91*	*62.30*	*61.26*	*45.1*
	POS±2	27.78	6.98	30.47	63.30	62.32	45.81
	Word±2	28.37	7.02	30.55	63.00	61.89	45.71
200K	Baseline	*29.96*	*7.32*	*31.90*	*60.97*	*60.01*	*44.06*
	POS±2	29.10	7.17	31.25	62.14	61.18	44.91
	Word±2	29.33	7.23	31.42	61.57	60.62	44.63
500K	Baseline	*31.08*	*7.47*	*32.67*	*59.86*	*58.83*	*43.18*
	POS±2	30.54	7.42	32.37	60.41	59.52	43.63
	Word±2	30.54	7.42	32.25	60.38	59.50	43.54
1M	Baseline	*31.52*	7.54	32.94	59.45	58.50	42.84
	POS±2	31.15	7.51	32.75	59.57	58.72	42.93
	Word±2	31.24	7.49	32.76	59.6	58.67	43.03
1.64M	Baseline	*31.92*	*7.60*	*33.24*	*59.06*	*58.14*	*42.42*
	POS±2	31.41	7.51	33.01	59.52	58.55	42.98
	Word±2	31.61	7.57	32.92	59.36	58.45	42.76

Table 4.24: Results of English-to-Spanish learning curve experiments with IGTree classifier.

Train. Size	Experiments	BLEU	NIST	METEOR	TER	WER	PER
10K	Baseline	*22.68*	*6.00*	*26.15*	*68.93*	*67.58*	*50.81*
	POS±2	21.61	5.85	25.52	69.86	68.67	51.63
	Word±2	21.94	5.9	25.67	69.78	68.36	51.36
20K	Baseline	*24.58*	*6.38*	*27.77*	*66.51*	*65.16*	*48.71*
	POS±2	24.67	6.40	27.88 (89.87%)	66.59	65.31	48.72
	Word±2	24.59	6.39	27.86	66.70	65.40	48.74
50K	Baseline	*27.33*	*6.84*	*29.93*	*64.15*	*62.97*	*46.63*
	POS±2	27.51 (98.1%)	6.90	30.14 (99.16%)	63.39	62.48	46.08
	Word±2	27.39 (72%)	6.86	29.98 (72.59%)	63.75	62.74	46.41
100K	Baseline	*28.64*	*7.09*	*30.91*	*62.30*	*61.26*	*45.1*
	POS±2	29.05 (98.7%)	7.16	31.16 (98.5%)	62.14	61.19	45.04
	Word±2	28.84 (96.7%)	7.12	31.04 (94.7%)	62.39	61.26	45.10

Table 4.25: Results of the English-to-Spanish learning curve experiments with TRIBL classifier.

We plot the BLEU, METEOR and TER learning curves of the word- and POS-based context-informed models (POS±2 and Word±2) for TRIBL and IGTree, as well as the Moses baseline in the left-side graphs (top, centre and bottom graphs represent respectively BLEU, METEOR and TER learning curves) of Figure 4.1. Each graph in the figure adopts a logarithmic horizontal axis, representing the number of training sentences. In addition, the three right-hand side graphs of Figure 4.1 represent the BLEU (top), METEOR (centre) and TER (bottom) score-difference curves of the two classifier experiments against the baseline, highlighting the gains and losses against the baseline. The IGTree curves extend up to the maximum of 1.64M training sentences; as noted earlier, due to limitations in memory, the TRIBL experiments extend only up to 100K training sentences.

We see from the top two graphs in Figure 4.1 that BLEU learning and score-difference curves of the POS±2 and Word±2 systems always remain below the Moses baseline curves while employing IGTree as the classifier. In contrast, BLEU learning and score-difference curves mostly remain above the Moses baseline curves when we employ TRIBL as the classifier. METEOR learning and score-difference curves (centre-left and right graphs in Figure 4.1) resemble the BLEU curves. The bottom two graphs of Figure 4.1 show TER learning and score-difference curves, respectively. Note that TER is an error metric, so a lower score indicates better performance. When we used IGTree as the classifier, TER learning and score-difference curves remain always above the baseline curves; on the other hand, TER learning and score-difference curves mostly remain below the baseline curves when TRIBL is employed as the classifier.

To summarize, in the English-to-Spanish translation task, (a) TRIBL appears to be effective on both small and moderately large-scale data sets, though its memory needs prohibit it from being used with the larger-sized training sets; on the other hand, it does not improve the context-informed models on the smallest amounts of training data tested (e.g. 10K sentences); (b) IGTree does not offer improvements over the baseline either with the small- or the large-scale context-informed models,

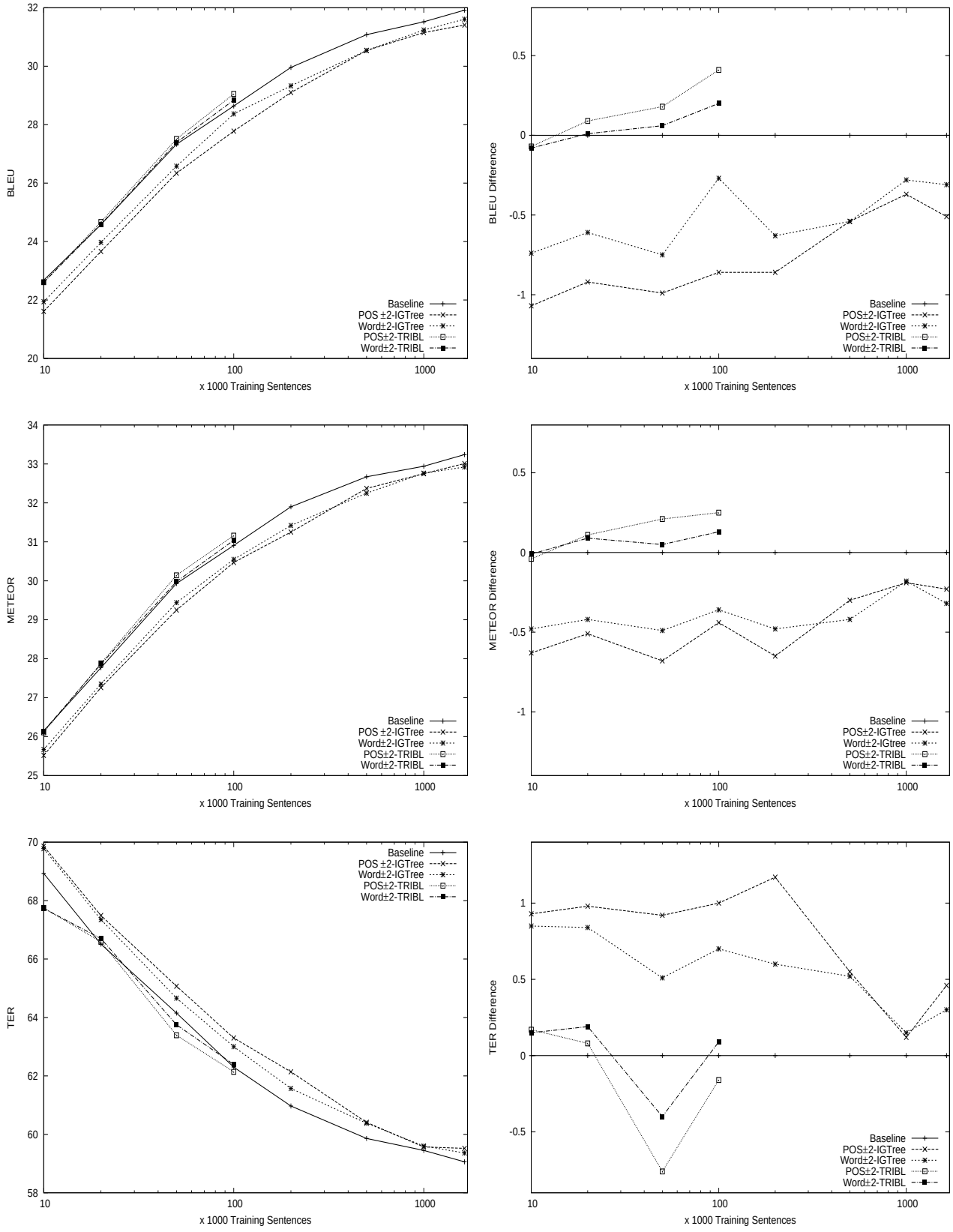

Figure 4.1: Learning curves (left-hand side graphs) and score-difference curves (right-hand side graphs) comparing the Moses baseline against word- and POS-based context-informed models (POS±2 and Word±2) in English-to-Spanish translation task. The curves are plotted with scores obtained using three evaluation metrics: BLEU (top), METEOR(centre) and TER (bottom).

while the performance of the large-scale context-informed models with the IGTree classifier are close to the performance of the Moses baseline.

We carried out an analysis on the translations produced by our best-performing context-informed (CI) system (POS±2)[14] and the Moses baseline. Table 4.26 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 4.27 compares weights of the various translational features of the POS±2 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	643	562	310	485
Sentence-Level TER	606	605	751	38
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	2.1		2	
Matching Words (%)	59.95		59.65	

Table 4.26: Comparison between translations produced by the best-performing context-informed (CI) system (POS±2) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0795	0.0448	0.0091	0.0449	0.0527	0.1151	-0.1451	-	-
POS±2	0.1043	0.0411	0.0741	-0.0168	0.0260	0.1096	-0.1164	0.0619	0.0336

Table 4.27: Comparison of weights for each translational feature of the two systems (POS±2 and Moses baseline) obtained by MERT training.

4.3.5.2 Dutch-to-English

Originally we ran our large-scale experiments on the Dutch-to-English and English-to-Dutch language pairs with the IGTree classifiers (cf. Section 4.3.2). As stated in the above section, we observe in the English-to-Spanish learning curve experiments that TRIBL seems to be a more effective classifier than IGTree in improving the performance of the context-informed SMT systems, but we were able to use TRIBL classifiers only for the small-scale translations due to its relatively high memory requirements. However, TRIBL was reprogrammed recently, and can now efficiently

[14]The best-performing POS±2 system which we used for this analysis was built on 100K training set with TRIBL (cf. Table 4.25).

handle large numbers of examples. This inspired us to deploy TRIBL classifiers for the large-scale translation.

To investigate the consequences of the different context-informed SMT systems on the increasing sizes of training sets while employing TRIBL as the classifier, we carried out experiments on the Dutch-to-English and English-to-Dutch language pairs. First, like the division of our English-to-Spanish data set (cf. Section 4.3.5.1), we segmented the Dutch-to-English training set (cf. Section 3.6) into eight pseudo-exponentially increasing training sets: 10K, 20K, 50K, 100K, 200K, 500K, 1M, and 1,311,111 training sentences. In this section, we report the outcomes of the Dutch-to-English learning curve experiments.

Train. Size	Experiments	BLEU	NIST	METEOR	TER	WER	PER
	Baseline	*18.31*	*5.14*	*46.73*	*69.13*	*74.06*	*54.77*
10K	POS±2	18.62 (97.1%)	5.19	46.78 (60.6%)	68.52	73.19	54.50
	Word±2	18.58 (93.8%)	5.18	46.84 (82.4%)	68.74	73.34	54.73
	Baseline	*20.13*	*5.47*	*49.14*	*67.22*	*72.24*	52.79
20K	POS±2	20.37 (91.9%)	5.51	49.32 (92.6%)	66.79	71.59	52.49
	Word±2	20.41 (89.7%)	5.49	49.25 (71.8%)	66.79	71.54	52.69
	Baseline	*23.35*	*5.96*	*52.54*	*63.85*	*68.91*	49.76
50K	POS±2	23.71 (90.7%)	6.02	52.66 (75.6%)	63.62	68.36	49.67
	Word±2	23.71 (96.3%)	6.00	52.68 (78%)	63.72	68.47	49.81
	Baseline	24.72	6.19	54.18	62.30	67.54	48.35
100K	POS±2	25.06 (68%)	6.25	54.38 (85.1%)	61.82	66.86	48.18
	Word±2	25.24 (98.9%)	6.26	54.50 (98%)	61.81	66.87	48.03
	Baseline	25.89	6.36	55.30	61.02	66.34	47.15
200K	POS±2	26.01 (35%)	6.45	55.52 (94.2%)	60.48	65.67	46.81
	Word±2	26.18 (87%)	6.42	55.50 (92.9%)	60.60	65.61	46.95
	Baseline	26.56	6.53	56.23	59.89	65.02	46.25
500K	POS±2	27.06 (99.7%)	6.63	56.78 (99.9%)	59.38	64.40	45.71
	Word±2	26.94 (99.9%)	6.57	56.40 (89.6%)	59.62	64.75	46.13
	Baseline	27.06	6.63	56.60	59.08	64.29	45.54
1M	POS±2	27.14 (54%)	6.68	56.83 (92.6%)	58.86	63.98	45.43
	Word±2	27.30 (88.5%)	6.67	56.85 (95.5%)	58.82	63.95	45.45
	Baseline	27.29	6.68	56.81	58.65	63.96	45.17
1.31M	POS±2	27.35 (13%)	6.71	56.84 (32.1%)	58.61	63.72	45.23
	Word±2	27.59 (84%)	6.72	57.11 (98.1%)	58.46	63.65	45.06

Table 4.28: Results of the Dutch-to-English learning curve experiments with TRIBL classifier.

Results obtained from the Dutch-to-English learning curve experiments are dis-

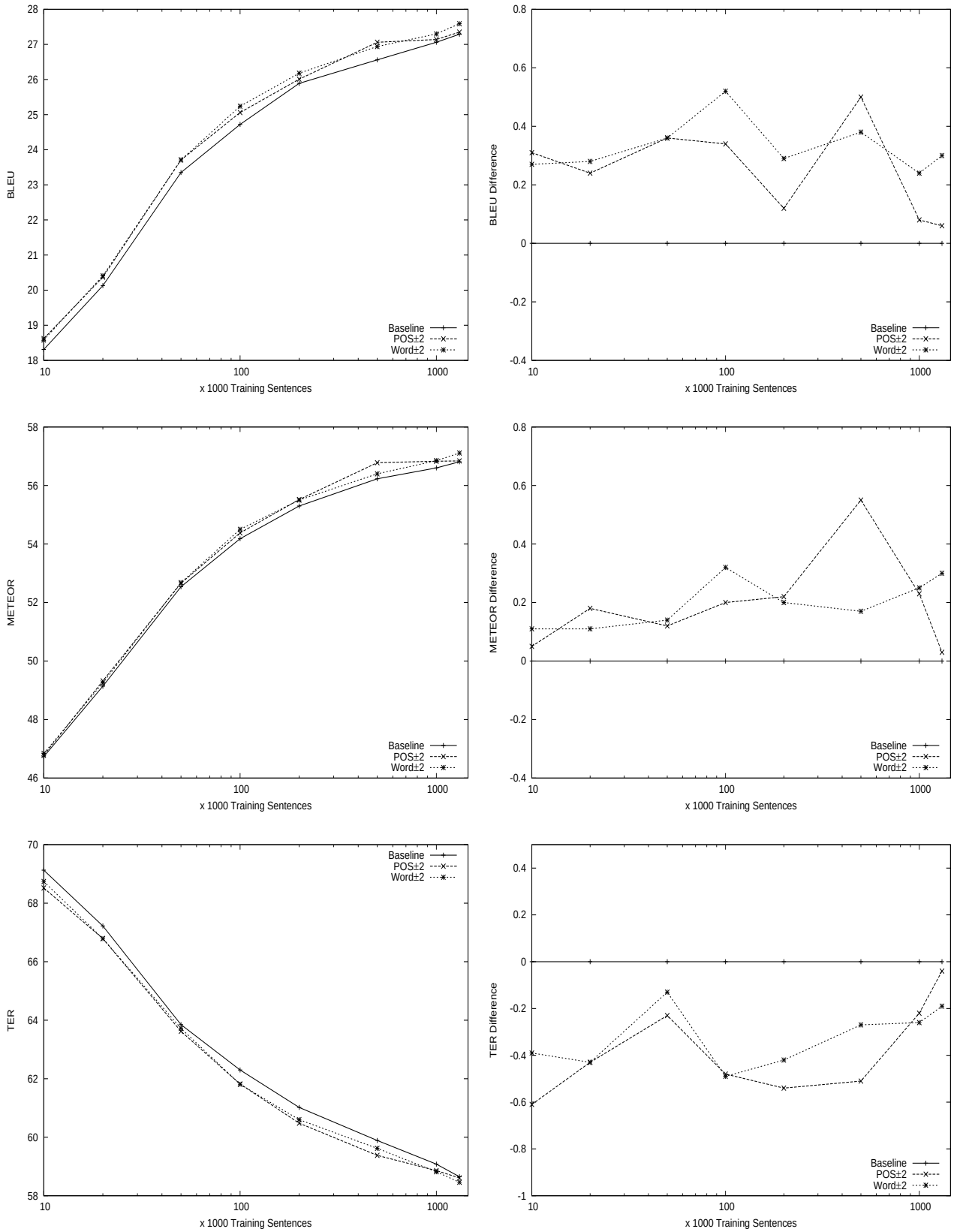

Figure 4.2: Learning curves (left-hand side graphs) and score-difference curves (right-hand side graphs) comparing the Moses baseline against word- and POS-based context-informed models (POS±2 and Word±2) in Dutch-to-English translation task. These curves are plotted with scores obtained using three evaluation metrics: BLEU (top), METEOR(centre) and TER (bottom).

played in Table 4.28, where we see that POS±2 and Word±2 produce statistically significant BLEU improvements with respect to the Moses baseline at most training data sizes. Performance measured by the METEOR and the TER evaluation metrics seem to be follow the same trend as BLEU. Drawing graphs with the evaluation scores produced by the BLEU, METEOR and TER metrics, we thoroughly investigate the effects of the word- and POS-based models against the Moses baseline with increasing sets of training data.

Figure 4.2 shows the learning and score-difference curves comparing the Dutch-to-English Moses baseline against the two context-based models (POS±2, Word±2). The three left-hand side graphs in Figure 4.2 show respectively BLEU (top), METEOR (centre) and TER (bottom) learning curves representing the performance of the context-informed models against the baseline. In contrast, the three right-hand side graphs in Figure 4.2 show respectively BLEU (top), METEOR (centre) and TER (bottom) score-difference curves, highlighting the gains and losses against the baseline.

We observe that the BLEU and METEOR curves (learning and score-difference) of the word- and POS-based models always remain above the baseline curve from the starting point (10K training data) to the end point (1.31M training data). The bottom two graphs of Figure 4.2 show that the TER curves (learning & score-difference) of the context-informed models (POS±2, Word±2) always remain below the baseline curves, which indicates the effectiveness of the basic contextual features in this translation task also according to the TER evaluation metric.

In sum, both words and POS tags, when used as source-language contexts in this translation task, show approximately similar improvements when using TRIBL as the classifier.

As an additional point of analysis, we compare translations produced by our best-performing context-informed (CI) system (Word±2)[15] with those by the Moses

[15]The best-performing PR system which we used for this analysis was built on the largest available training data (1.31M sentences) with TRIBL classifier (cf. Table 4.28).

baseline. Table 4.29 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 4.30 compares weights of the various translational features of the Word±2 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	248	201	271	280
Sentence-Level TER	255	223	481	41
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	4.3		4.2	
Matching Words (%)	65.28		65.23	

Table 4.29: Comparison between translations produced by the best-performing context-informed (CI) system (Word±2) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1086	0.0641	0.0107	0.0569	0.0867	0.0977	-0.2459	-	-
Word±2	0.0747	0.0324	0.0257	0.0583	0.0644	0.0822	-0.1882	-0.0095	0.0034

Table 4.30: Comparison of weights for each translational feature of the two systems (Word±2 and Moses baseline) obtained by MERT training.

4.3.5.3 English-to-Dutch

In this section, we report the outcomes of the English-to-Dutch learning curve experiments. Results obtained from the English-to-Dutch learning curve experiments are displayed in Table 4.31, where we see that none of the BLEU improvements for the POS±2 and the Word±2 systems with respect to Moses baseline are statistically significant. However, statistically significant improvements in METEOR for the Word±2 system with respect to the baseline are to be observed at larger amounts of training data. The Word±2 system improves on the Moses baseline at most training data sizes (100K to 1.31M) according to the performance measured by TER, while POS±2 system improves on the Moses baseline at smaller amounts (10K, 20K, and 100K) as well as larger amounts (1M and 1.31M) of training data. We draw graphs with the evaluation scores produced by the BLEU, METEOR and TER metrics for the POS- and word-based SMT models as well as for the Moses baseline, by which we investigate the effects of the context-informed models thoroughly against the

baseline with a increasing set of training data.

Figure 4.3 illustrates BLEU (top), METEOR (centre) and TER (bottom) learning curves (left-hand side graphs) and score-difference curves (right-hand side graphs) which compare the performance of the Moses baseline and the basic context-informed models (POS±2 and Word±2) in the English-to-Dutch translation task.

Train. Size	Experiments	BLEU	NIST	METEOR	TER	WER	PER
10K	Baseline	17.20	4.99	43.74	72.40	75.39	57.17
	POS±2	17.12	4.99	43.76	72.20	75.22	57.30
	Word±2	16.32	4.92	43.07	72.88	75.91	57.39
20K	Baseline	19.03	5.31	46.30	70.33	73.69	55.08
	POS±2	19.11(78.1%)	5.32	46.23	70.14	73.48	55.07
	Word±2	18.56	5.29	45.87	70.74	74.16	55.26
50K	Baseline	21.70	5.74	49.31	67.32	71.19	52.48
	POS±2	21.80 (33.4%)	5.75	49.33	67.40	71.07	52.51
	Word±2	21.06	5.69	48.62	67.92	71.66	52.94
100K	Baseline	22.53	5.88	50.30	66.42	70.48	51.84
	POS±2	22.49	5.89	50.22	66.14	70.35	51.72
	Word±2	22.18	5.89	50.23	66.31	70.52	51.50
200K	Baseline	23.47	6.04	51.46	65.30	69.40	50.90
	POS±2	23.54 (65%)	6.05	51.56 (73.14%)	65.37	69.22	50.71
	Word±2	23.19	6.06	51.27	65.07	69.22	50.71
500K	Baseline	24.06	6.11	52.06	64.59	68.58	50.36
	POS±2	23.89	6.11	52.01	64.70	68.77	50.35
	Word±2	24.02	6.18	52.36 (96.9%)	64.09	68.45	49.84
1M	Baseline	24.26	6.17	52.39	64.55	68.72	50.12
	POS±2	24.15	6.16	52.37	64.51	68.69	50.11
	Word±2	24.46 (83%)	6.25	52.61 (93.4%)	63.58	67.96	49.51
1.31M	Baseline	24.26	6.17	52.68	64.36	68.80	50.02
	POS±2	24.39 (88%)	6.20	52.61	64.11	68.38	49.77
	Word±2	24.36 (57%)	6.28	52.84 (88.35%)	63.66	68.10	49.20

Table 4.31: Results of the English-to-Dutch learning curve experiments with TRIBL classifier.

We see from the top-left graph in Figure 4.3 that the BLEU learning curve of the word-based model (Word±2) starts much below the baseline curve, crosses it at 1M training data, and remains above it till the end. On the other hand, the BLEU learning curve of the POS±2 resembles the Moses baseline curve, although, it resides above the baseline at 20K, 50K, 200K and 1.31M amounts of training data. Moreover, one can perceive this phenomenon looking at the BLEU score-difference

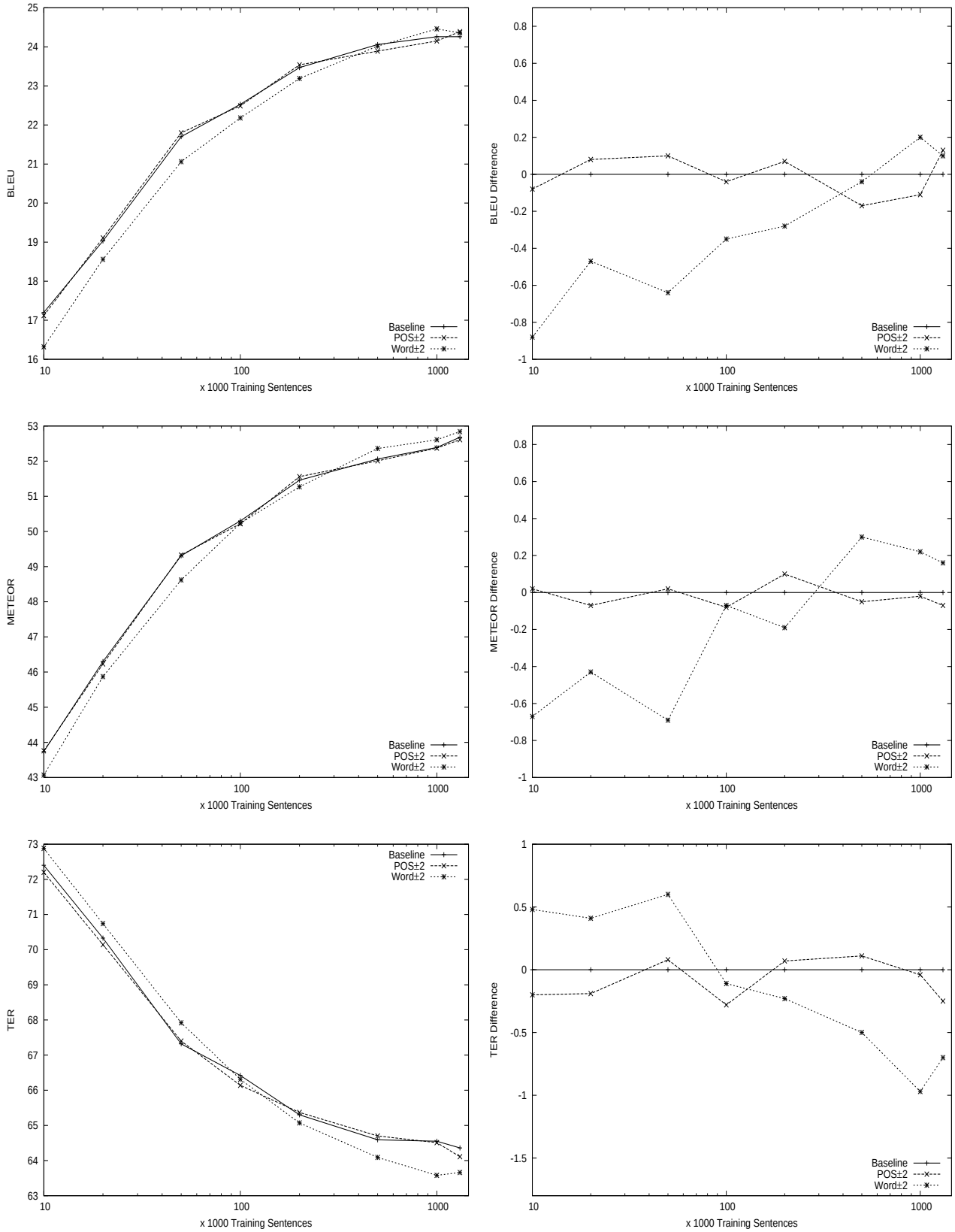

Figure 4.3: Learning curves (left-hand side graphs) and score-difference curves (right-hand side graphs) comparing the Moses baseline against word- and POS-based context-informed models (POS±2 and Word±2) in English-to-Dutch translation task. These curves are plotted with scores obtained using three evaluation metrics: BLEU (top), METEOR(centre) and TER (bottom).

curves for the POS±2 and the Word±2.

We observe from the centre-left and -right graphs in Figure 4.3 that METEOR learning and score-difference curves approximately follow the BLEU curves. The METEOR learning-curve for the Word±2 crosses the baseline curve for the larger amounts of training data (500K to 1.3M). In contrast, the METEOR learning-curve for the POS±2 stays beneath the baseline curve for the larger amounts of training data (500K to 1.3M).

The bottom-left and -right graphs in Figure 4.3 show TER learning and score-difference curves, respectively. The TER learning and score-difference curves of the POS±2 system approximately resemble the Moses baseline curve. On the other hand, TER learning curve of the Word±2 starts above the baseline curve, crosses it for 100K amount of training data, and remains below it till the end.

We sum up the above observations despite the fact that it is difficult to reach a concrete conclusion. In this translation task, word-based features appear to be an effective source-language context for the larger amounts of training data. Integration of POS tags as a source-language context into the PB-SMT model does not show much consistency in improving the baseline.

As an additional point of analysis, we compare translations produced by our best-performing context-informed (CI) system (POS±2)[16] with those by the Moses baseline. Table 4.32 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 4.33 compares weights of the various translational features of the POS±2 and the baseline systems obtained by MERT training.

4.3.6 Analysis of Learning Curve Experiments

As far as English-to-Spanish translation is concerned, we see in Section 3.6 that the vocabulary size of the Spanish training set is four times larger than that of the

[16]The best-performing POS±2 system which we used for this analysis was built on the largest available training data (1.31M) (cf. Table 4.31).

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	226	189	256	329
Sentence-Level TER	274	219	469	38
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	3.8		3.8	
Matching Words (%)	61.71		61.74	

Table 4.32: Comparison between translations produced by the best-performing context-informed (CI) system (POS±2) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1072	0.0102	0.0509	0.1103	0.0468	0.0936	-0.2689	-	-
POS±2	0.0892	-0.0053	0.0446	0.0528	0.055	0.0677	-0.2773	0.02574	0.0045

Table 4.33: Comparison of weights for each translational feature of the two systems (POS±2 and Moses baseline) obtained by MERT training.

English training set. Accordingly, availing of source-side contextual information has much more potential to improve translation quality in the Spanish-to-English direction rather than the other way round. Nevertheless, even for the reverse language direction, source-side basic contexts (words and part-of-speech tags) were able to improve over the baseline (cf. Section 4.3.5.1). If we look at the Dutch-to-English learning curves, we see that both words and POS tags show their importance as a source-side context while integrated into the any size of PB-SMT model. Like the English-to-Spanish direction, the English-to-Dutch direction should be a difficult choice if we want to utilize source-side contexts in order to improve lexical selection. Nevertheless, words and POS tags appear to be effective source-side contexts in the large-scale English-to-Dutch translation. Moreover, in this translation task, when the PB-SMT model is built with small amounts of training data, benefits from adding POS tags as a source-language context are also to be seen.

4.4 Experiments using Context-Informed HPB-SMT

We integrate words and POS tags as source-language contexts into a hierarchical PB-SMT (HPB-SMT) model, and perform a range of experiments. In Section 3.3.2, we demonstrated how source-language context can be incorporated into a Hiero

model, and in Section 4.2, we presented how basic features (words and POS tags) are used in order to model contextual information (CI) of a Hiero source phrase.

In this section, we report experimental results obtained from the two different translation tasks. The system outputs are evaluated across a wide range of automatic evaluation metrics, which were discussed in Section 3.7. Section 4.4.1 reports the outcomes of English-to-Hindi and English-to-Dutch translations. Section 4.4.2 provides some analysis of results obtained from the two translation tasks.[17]

4.4.1 Experimental Results

4.4.1.1 English-to-Hindi

Experimental results of the English-to-Hindi translation task are displayed in Table 4.34 which shows the performance of the two context-based models (Word±2, POS±2) against the Hiero baseline.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*11.08*	*4.44*	*29.19*	*75.30*	*82.09*	*57.98*
Word±2	11.58 (96.1%)	4.50	26.32	74.27	80.53	57.51
POS±2	11.79 (99.3%)	4.53	29.32	73.96	79.93	57.10

Table 4.34: Results on English-to-Hindi translation obtained integrating basic contexts into Hiero.

We see from Table 4.34 that addition of words and part-of-speech tags as source-language contexts improves the Hiero baseline significantly across most evaluation metrics. The POS- and word-based models add 0.71 BLEU points (6.8% relative increase) and 0.50 BLEU points (4.5% relative) to the Hiero baseline. These improvements with respect to the baseline are statistically significant. Improvements obtained with the error metrics (TER, WER and PER) seem to be quite similar to those obtained with the BLEU.

We see one exception in Table 4.34, where Word±2 produces about 3 METEOR points less than the Hiero baseline. Like Dutch, the METEOR evaluation metric

[17]The experiments reported in this section have partly been published, albeit in a different form, in Haque et al. (2010a).

does not support the Hindi language (Lavie and Agarwal, 2007), which might be the reason for the above anomaly.

As an additional point of analysis, we compare translations produced by our best-performing context-informed (CI) system (POS±2) (cf. Table 4.34) with those by the Hiero baseline. Table 4.35 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 4.36 compares weights of the various translational features of the POS±2 and the baseline systems obtained by MERT training.

	CI>Hiero	CI<Hiero	CI=Hiero	Zero
Sentence-Level BLEU	101	61	17	316
Sentence-Level TER	212	150	133	0
Closeness to Reference Set	CI		Hiero	
Matching Translations (%)	0		0	
Matching Words (%)	51.49		51.10	

Table 4.35: Comparison between translations produced by the best-performing context-informed (CI) system (POS±2) and the Hiero baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{glue}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1885	0.0817	-0.0093	0.0914	0.0312	0.1227	-0.0598	-0.4151	-	-
POS±2	0.1421	0.0060	0.0158	0.0463	0.0680	-0.0855	0.0035	-0.6120	0.0166	0.0031

Table 4.36: Comparison of weights for each translational feature of the two systems (POS±2 and Hiero baseline) obtained by MERT training.

4.4.1.2 English-to-Dutch

Experiments on English-to-Dutch translation (Haque et al., 2010a) were carried out on the Open Subtitles corpus (cf. Section 3.6).[18] The results obtained are shown in Table 4.37. We observe that the English-to-Dutch subtitle translation task benefits from the addition of source-language context features, as the inclusion of any type of contextual feature improves upon the Hiero baseline across all evaluation metrics. Adding words as source contexts adds 0.41 BLEU points (a relative improvement of 1.87%) to the baseline. Somewhat higher improvements are observed with the

[18]Experimental results reported in this section has been described in (Haque et al., 2010a)

addition of POS context (0.47 BLEU; 2.15% relative increase). However, none of the BLEU improvements are statistically significant with respect to the HPB-SMT baseline.

Exp.	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*21.92*	*5.29*	*43.06*	*56.72*	*55.43*	*48.60*
Word±2	22.33 (77%)	5.33	43.23 (96%)	56.36	55.32	48.28
POS±2	22.39 (80%)	5.33	43.66 (96.2%)	56.68	55.6	48.54
Word±2+POS±2	22.22 (30%)	5.29	43.85 (93.4%)	56.93	55.71	48.54

Table 4.37: Experimental results with individual features, compared against the Hiero baseline.

When focusing on the METEOR evaluation metric, we find that among the individual features, POS±2 produces the highest improvements (0.60 points; 1.4% relative increase) over the baseline. Improvement in METEOR is also observed for the the Word±2 (0.17 METEOR; 0.4%). Unlike BLEU, METEOR improvements for the Word±2 and the POS±2 with respect to the baseline are statistically significant. Improvement in TER for the Word feature (a reduction of 0.36 TER points) is comparable to improvements in METEOR and BLEU evaluation metrics. We see an exception for the POS±2 which produces only a 0.04 point reduction in TER over the Hiero baseline.

Subsequently, we performed an experiment in which we combined the lexical features with the syntactic features. Results obtained combining word and POS contexts are shown in the last row of Table 4.37. Interestingly, POS features together with word contexts cause system performance to deteriorate compared to the individual results; we observe only a 0.30 BLEU point improvement (1.38% relative increase, not statistically significant) over the baseline.

As an additional point of analysis, we compare translations produced by our best-performing context-informed (CI) system (POS±2) (cf. Table 4.37) with those by the Hiero baseline. Table 4.38 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 4.39 compares weights of the various translational features of the POS±2 and the baseline systems obtained by

MERT training.

	CI>Hiero	CI<Hiero	CI=Hiero	Zero
Sentence-Level BLEU	46	40	129	785
Sentence-Level TER	103	106	679	112
Closeness to Reference Set	CI		Hiero	
Matching Translations (%)	12.4		12.4	
Matching Words (%)	58.18		57.66	

Table 4.38: Comparison between translations produced by the best-performing context-informed (CI) system (POS±2) and the Hiero baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{glue}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0823	0.0205	0.0217	0.1702	0.0409	-0.3477	0.3421	0.095	-	-
POS±2	0.1081	0.0023	0.0180	0.1807	0.0436	-0.3842	-0.3841	-0.088	0.0070	0.0021

Table 4.39: Comparison of weights for each translational feature of the two systems (POS±2 and Hiero baseline) obtained by MERT training.

4.4.2 Discussion

Basic features such as neighbouring words and POS tags were successfully integrated as source-side contexts into the state-of-the-art hierarchical phrase-based SMT system, Hiero. Both words and POS tags appear to be effective source-language contexts in English-to-Hindi translation since addition of such context types surpasses the Hiero baseline with statistically significant gains in BLEU. For English-to-Dutch translation, incorporation of basic contextual features provides moderate gains in BLEU over a Hiero baseline, which are close to the significance level. Moreover, we observed statistically significant improvements in METEOR for both word- and POS-based models. Thus, words and POS tags prove to be effective source-language contexts in the state-of-the-art Hiero model.

4.5 Machine Transliteration: An Application of Context-Informed PB-SMT

4.5.1 Machine Transliteration Overview

A process which translates proper nouns and technical terms across languages with different alphabets and sound inventories is commonly called transliteration (Knight and Graehl, 1998). In other words, the transliteration process finds a phonetic equivalent in the target language for a given named entity (NE) or proper noun written in the source language. If two languages use the same set of alphabets, the transliteration task becomes easier since the word can be copied verbatim in the target output. For example, a phrase like 'Antonio Gil' in English usually gets translated as 'Antonio Gil' in Spanish (Knight and Graehl, 1998). Different challenges arise in transliteration for those language pairs that differ in alphabets and sound inventories, such as English-to-Hindi, English-Chinese, and English-to-Arabic, etc. There are numerous ways of performing automatic transliteration, e.g. with the noisy channel model (Knight, 1999), joint source channel model (Li et al., 2004), decision-tree based model (Kang and Choi, 2000), statistical SMT model (Matthews, 2005), etc.

4.5.2 Impact of Transliteration on Machine Translation

Machine transliteration is of key importance in many cross-lingual natural language processing applications, such as MT, information retrieval and question answering. Named entities (NEs) or proper nouns rarely appear in the training sentences from which MT systems are usually built. Moreover, NEs are very productive in nature. Therefore, translating out-of-vocabulary NEs is a real problem for any MT system. While translating an input source sentence having an out-of-vocabulary NE, the MT system usually copies that OOV NE verbatim to the target output. Thus, an NE composed of the source-language alphabet appears in the MT output, which a native

speaker of the target language can seldom recognize. In this regard, transliteration plays an important role in the MT process, as it can translate out-of-vocabulary NEs of the source language into NEs of the target language. MT researcher have deployed transliteration engines into the MT system order to translate OOV NEs (Langlais and Patry, 2007; Habash, 2008).

4.5.3 Machine Transliteration with Context-Informed PB-SMT

We adapt the PB-SMT model of Koehn et al. (2003) for transliteration, where characters are translated rather than words as in a character-level translation system (Lepage and Denoual, 2006). However, we go a step further from the basic PB-SMT model by using source-language contextual features (Stroppa et al., 2007). We also create transliteration models by constraining the character-level segmentations, i.e. treating a consonant-vowel cluster as one transliteration unit. Our machine transliteration models are based on the PB-SMT (Koehn et al., 2003) toolkit Moses (Koehn et al., 2007).

In order to carry out the transliteration experiments, we adopt two different experimental approaches depending on the nature of the transliteration unit (character-level and syllable-level) employed, which are described as follows.

4.5.3.1 Character-Level Transliteration

In this approach, named entities are split into characters (i.e. alphabets), each of which can be viewed as the transliteration unit. The transliteration system inputs a NE represented as a sequence of characters in the source language, and generates a target NE in the form of a sequence of characters in the target language. Henceforth, we refer to this approach as character-level (CL) transliteration.

Each NE in the training data (parallel NE list) is represented as a sequence of characters in its respective language in order to build the transliteration model.

4.5.3.2 Syllable-Level Transliteration

In this approach, we break NEs into transliteration units which bear close resemblance to syllables. In our English-to-Hindi transliteration task, we split English NEs into transliteration units having a C*V* pattern (C: consonant, V: vowel) and Hindi NEs are divided into transliteration units having a Ch+M pattern (M: Hindi matra/vowel modifier, Ch: characters other than matras) (Ekbal et al., 2006). The transliteration system inputs an NE represented as a sequence of syllables in the source language, and generates the target NE in the form of a sequence of syllables in the target language. Henceforth, we refer to this approach as syllable-level (SL) transliteration.

Each NE of the training data is represented as a sequence of syllables in its respective language in order to build the transliteration model.

4.5.3.3 Experimental Set-Up

We carried out transliteration experiments on both character-level (CL) and syllable-level (SL) data. In order to build our context-informed transliteration system, we derive two memory-based log-linear features $\hat{h}_{\text{mbl}}$ and $\hat{h}_{\text{best}}$ (cf. Section 3.3.1), and integrate them into the log-linear framework of Moses. Like lexical contextual information (CI) defined in Equation (4.1), we form contextual information (CI) of a transliteration unit (character or syllable) in order to derive the above memory-based log-linear features. In short, we followed the word-based context-informed PB-SMT model in order to conduct context-sensitive transliteration experiments, where characters or syllables are assumed as words as in the standard phrase-based approach. The context-based transliteration experiments were performed with three different memory-based classifiers: IGTree, IB1 and TRIBL (Daelemans and van den Bosch, 2005). The Moses PB-SMT system serves as our baseline.

4.5.3.4 Data

In order to carry out experiments, we used 10,000 parallel NEs from the NEWS 2009[19] (Li et al., 2009) English-to-Hindi training data (Kumaran and Tobias, 2007). Henceforth, we refer to this small amount of data as SmallEH. Additionally, we used English-to-Hindi parallel person names data (105,905 distinct name pairs) of the Election Commission of India (ECI).[20] We add SmallEH data together with the ECI data to conduct large-scale experiments. Henceforth, we refer to the combined large size of data as LargeEH. Both the development set and test set are taken from NEWS 2009, each of which contains 1,000 parallel NEs.

4.5.3.5 Evaluation

In order to evaluate transliteration output with respect to the reference transliteration set, we used the Word Accuracy in Top-1 (ACC) metric described in (Li et al., 2009). It measures the correctness of the first transliteration candidate in the candidate list generated by a transliteration system. ACC = 1 means that the top candidate is correct transliteration, i.e. it matches one of the references, and ACC = 0 means that the top candidate is incorrect. ACC is defined as in Equation (4.6):

$$\text{ACC} = \sum_{i=1}^{N} \begin{cases} 1 & \text{if } \exists r_{ij} : r_{ij} = c_{i1} \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

where N: total number of names (source words) in the test set, r_{ij}: j-th reference transliteration for i-th name in the test set, and c_{ik}: k-th candidate transliteration (system output) for i-th name in the test set ($1 \leq k \leq 10$).

[19] http://www.acl-ijcnlp-2009.org/workshops/NEWS2009/
[20] http://www.eci.gov.in/DevForum/Fullname.asp

4.5.4 Experimental Results

In addition to the baseline Moses system, we carried out three different set of experiments with IGTree, IB1 and TRIBL. Each of these experiments was carried out on both the SmallEH and the combined larger data (LargeEH), both at character level (CL) and syllable level (SL), and considering ±1/±2 tokens as context. For each experiment, we produce the 10-best distinct hypotheses; nevertheless, just top candidate translation is used in evaluation (cf. Equation (4.6)).

Data Set	SmallEH				LargeEH			
Transliteration Unit	CL		SL		CL		SL	
Moses Baseline	.290		.391		.352		.407	
Context Size	±1	±2	±1	±2	±1	±2	±1	±2
IB1	0.391	0.386	0.406	0.359	0.431	0.420	0.437	0.427
IGTree	0.372	0.371	0.412	0.416	0.413	0.407	0.445	0.427
TRIBL	0.382	0.399	0.408	0.395	0.439	0.421	0.444	0.439

Table 4.40: Results of Context-Informed PB-SMT on Transliteration.

We observed that many of the (unseen) transliteration units (TU) in the test set remain untranslated in SL systems due to the problem of data sparseness. Whenever an SL system fails to translate a TU, we fall back on the corresponding CL system to translate the TU as a post-processing step.

The experimental results are displayed in Table 4.40. As can be seen from Table 4.40, the accuracy of the SL baseline system (0.391 ACC points) is much higher than that of the CL baseline system (0.290 ACC points) on the SmallEH data set. Similarly, the accuracy of the SL baseline system (0.407 ACC points) is also much higher than that of the CL baseline system (0.352 ACC points) on the LargeEH data set.

We see from Table 4.40 that the accuracy of any of the context-sensitive transliteration systems is much higher compared to that of the respective baseline. On the SmallEH data set, the CL system with ±2 size of context window produced the highest accuracy (0.399 ACC points; a 0.109 ACC point improvement; 36.45% relative increase) over the Moses baseline when TRIBL is employed as the classifier. On the same data set, the SL system with ±2 size of context window produced the highest

accuracy (0.416 ACC points; a 0.025 ACC point improvement; 6.40% relative) over the Moses baseline; this time, IGTree was used as the classifier.

Similar improvements are to be observed on the large-scale data set, i.e. LargeEH. The CL system with ±1 size of context window produced highest accuracy (0.439 ACC points; a 0.087 ACC point improvement; 24.71% relative) over the baseline when TRIBL is employed as the classifier. Overall highest accuracy was achieved (0.445 ACC points; a 0.038 ACC point improvement; 9.33% relative) for the SL system on LargeEH data set when ±1 size of context window and the IGTree classifier were used. The transliteration experiments described so far (in Table 4.40) have been reported in (Haque et al., 2009c) which presents DCU transliteration system for NEWS 2009 shared task (Li et al., 2009). Our transliteration systems secured 9th, 10th, 11th and 14th places among the 19 submissions in the NEWS 2009 shared task.

Subsequently, we changed our Moses baseline set-up, which results in an improved baseline model. We made two major changes in the baseline configuration: (a) maximum phrase length is set to 7 instead of 2 in our previous baseline set-up,[21] (b) the objective function of MERT (Och, 2003) was the BLEU evaluation metric (Papineni et al., 2002) in our previous baseline set-up, which is replaced with an edit distance-based evaluation metric PER (Tillmann et al., 1997). The improved baseline accuracies can be seen from Table 4.41, which displays the results produced by the different context-sensitive models. As can be seen from Table 4.41, the new configuration also affects the performance of all context-sensitive models which produce much better scores compared to the scores reported in Table 4.40. This time, we achieve the highest accuracy (0.476 ACC points; a 0.028 ACC point improvement; 6.25% relative increase) for the CL system with ±1 size of context window on the LargeEH data set while employing IB1 as the classifier.

We also conducted experiments with an additional set-up in which we added three

[21]The syllable-level system breaks NEs into usually a average of 3 to 5 syllables. That is why we chose maximum phrase length to 2 in our previous baseline set-up.

Data Set	SmallEH				LargeEH			
Transliteration Unit	CL		SL		CL		SL	
Moses Baseline	0.442		0.400		0.448		.407	
Context Size	±1	±2	±1	±2	±1	±2	±1	±2
IB1	0.441	0.461	0.390	0.392	0.476	0.470	0.408	0.421
IGTree	0.445	0.451	0.385	0.390	0.456	0.470	0.427	0.426
TRIBL	0.451	0.470	0.412	0.414	0.473	0.460	0.428	0.429
IB1+IGTree+TRIBL	0.453	0.465	0.393	0.400	0.466	0.479	0.435	0.447

Table 4.41: Transliteration results.

memory-based log-linear features together into the PB-SMT model. Three different features (IB1, IGTree and TRIBL) were derived separately from the distribution of target transliteration units (classification result) for a given source transliteration unit and its additional context information. Then these three features were collectively integrated into the PB-SMT model. The last row of Table 4.41 shows the experimental result obtained applying this set-up, where we see that overall highest accuracy (0.479 ACC points; a 0.031 ACC point improvement; 6.92% relative increase) is obtained for the CL system with ±2 size of context window and the combined set-up. We compare our best-performing system (cf. last row in Table 4.41) on new configuration with the top-performing systems[22] of the NEWS 2009 English-to-Hindi transliteration shared task (Li et al., 2009). We discovered that our best-performing system (CL±2: 0.479 ACC points) could have secured 3rd place amongst all the systems in the English-to-Hindi transliteration shared task.

4.5.5 Transliteration Examples

Figure 4.4 shows outputs (transliterations of two English NEs: 'Kaamalwala' and 'Mahil') produced by our best-performing context-informed (CI) transliteration system (CL±2 with combined set-up) and the baseline system. As can be seen from the Figure 4.4, our context-sensitive transliteration system generates target NEs that are similar to the references. In contrast, both target NEs generated by the baseline transliteration model is incorrect.

[22]Shared task report can be found at http://research.microsoft.com/pubs/115623/2009_NEWS_SharedTaskReport.pdf

Source	Reference	Baseline	CI Model
Kaamalwala	कामलवाला	क म ल व ा ल ा	क ा म ल व ा ल ा
Mahil	माहिल	म ह ि ल	म ा ह ि ल

Figure 4.4: Examples comparing transliterations produced by our best-performing context-informed (CI) transliteration system (CL±2 with combined set-up, cf. Table 4.41) and the baseline model.

4.5.6 Discussion

We have successfully employed source-context modeling into a state-of-the-art PB-SMT model for the English-to-Hindi transliteration task. We have shown that taking source context into account substantially improves the baseline transliteration systems. This work can be viewed as a deployment of the context-informed PB-SMT model in a different NLP application, i.e. machine transliteration.

4.6 Summary

In this chapter, we defined contextual information for a source phrase with basic features (words and POS tags). Then, we reported a varied set of experimental results obtained by integrating basic contextual features into the PB-SMT model (Koehn et al., 2003). This set of experiments involved a range of language pairs and a series of small- and large-scale data sets. In our experiments we employed two memory-based classification algorithms: IGTree and TRIBL. IGTree was used for large-scale translations.

On examining the evaluation results, we discovered that basic contextual features appear to be effective source-language contexts in small-scale translations. On the large-scale translations, we found that basic contextual features do not help much in improving MT quality. Experiments with increasing sizes of training sets yield a

set of learning curves comparing various context-based SMT models with the Moses baseline. We employed TRIBL as a classifier to conduct most of the learning curve experiments, where basic features appeared to be effective source-context for large-scale translations as well. As far as context-sensitive translations are concerned, TRIBL proved to be a more effective classifier than IGTree in improving MT quality.

We also demonstrated evaluation results obtained by employing basic contextual features in the Hiero model (Chiang, 2007). We examined the effectiveness of basic context in two different translation tasks: English-to-Hindi and English-to-Dutch. We observed that basic features seemed to be effective source-language contexts in the Hiero model as well.

Finally, we introduced our context-informed transliteration models. In addition to the result of the Moses baseline, we report a series of experimental results obtained from those models. With an English-to-Hindi transliteration task, we showed that our context-informed transliteration models improve significantly over the Moses baseline in terms of transliteration accuracy.

In the next chapter, we employ supertags as source-language contexts in the two state-of-the-art SMT models (Koehn et al., 2003; Chiang, 2007). Furthermore, we will explore various sentence-similarity features, and investigate the integration of such feature types into a PB-SMT model (Koehn et al., 2003) individually and collectively with supertag-based features.

Chapter 5

Lexical Syntactic Features

In the previous chapter, we investigated the incorporation of basic features (words and POS tags) as a source-language context into the state-of-the-art PB-SMT (Koehn et al., 2003) and the hierarchical PB-SMT (HPB-SMT) (Chiang, 2007) models. In this chapter, we introduce lexical syntactic information as a source-language context in the state-of-the-art SMT models (Koehn et al., 2003; Chiang, 2007). A range of experiments was carried out integrating lexical syntactic context into those models, which we will report in this chapter. This chapter is organized as follows. In Section 5.1, we present an overview of the lexical syntactic information. Section 5.2 illustrates how we make use of lexical syntactic information in order to form contextual information (CI) of a source phrase. The experimental results obtained adding lexical syntactic context into the PB-SMT and the HPB-SMT models are demonstrated in Sections 5.3 and 5.4, respectively. In Section 5.5, we describe how we derive sentence similarity-based contextual features, and integrate them into the PB-SMT model individually and in collaboration with lexical syntactic features. We place this discussion here because sentence similarity-based features were employed together with the supertag-based features in order to carry out context-based experiments.

5.1 Overview of Lexical Syntax

A lexicalized grammar can be viewed as a finite set of atomic structures each associated with a lexical item and a small set of operations to combine them into a complex structure. An atomic structure represents a syntactic construct (an elementary tree or a lexical category) associated with a lexical item. A finite set of operations is applied to assemble the elementary structures into a parse tree. Each elementary structure represents a complex linguistic category that expresses the specific syntactic behaviour of a word in terms of the arguments it takes, and more generally, the syntactic environment in which it appears.

In order to integrate lexical syntactic context into the state-of-the-art SMT models (Koehn et al., 2003; Chiang, 2007), we made use of two types of lexicalized grammar, namely combinatory categorial grammar (CCG) (Steedman, 2000) and lexicalized tree adjoining grammar (LTAG) (Joshi and Schabes, 1992). Both LTAG and CCG may assign one or more syntactic structures to each word in a sentence. Supertagging (Bangalore and Joshi, 1999) was introduced to reduce the number of elementary structures for each word, which in effect improves the parsing efficiency. Thus, a supertagger chooses the most correct 'supertag' (syntactic structure) amongst a set of syntactic structures. Both the LTAG (Chen et al., 2006) and the CCG (Hockenmaier, 2003) supertag sets were acquired from the WSJ section of the Penn-II Treebank using hand-built extraction rules.

In this regard, in a separate strand of research, Hassan et al. (2006, 2007, 2008) showed that incorporating supertags (Steedman, 2000; Joshi and Schabes, 1992) in the target language model and on the target side of the translation model could improve significantly on state-of-the-art approaches to MT. Despite the significance of this work, it is currently not possible to develop a fully supertagged PB-SMT system given supertaggers readily exist only for English.

5.1.1 Lexicalized Tree Adjoining Grammar

In lexicalized tree adjoining grammar (LTAG), the supertag constitutes an LTAG elementary tree category with a set of word-to-word dependencies. Bangalore and Joshi (1999) used a standard Markov model tagger to estimate probabilities for assigning LTAG elementary trees to words. An LTAG elementary tree encapsulates syntactic dependencies on the words in a sentence. In other words, LTAG supertags describe syntactic information such as the POS tag of a word, its subcategorization information and the phrase-hierarchy in which it appears.

There are two types of LTAG elementary trees: initial trees and auxiliary trees. Initial trees are minimal linguistic structures that contain no recursion. In contrast, auxiliary trees represent recursive structures, which are adjuncts to elementary structures. The LTAG elementary trees can be combined using two operations: substitution and adjunction. An initial tree is inserted into an elementary tree under the operation of substitution. An auxiliary tree is attached to an elementary tree under the operation of adjunction.

Figure 5.1 shows the LTAG supertags for the example sentence '*Can you play my favourite old record?*', which are assembled into a parse tree with the substitution and adjunction operations. The left-side of Figure 5.1 shows seven elementary trees, three of which are initial trees representing terminals '*you*', '*play*', and '*record?*'. The remaining elementary trees in Figure 5.1 are auxiliary trees representing terminals '*Can*', '*my*', '*favourite*', and '*old*'. The right-side of Figure 5.1 displays a phrase structure parse tree which is formed by combining the elementary trees (shown at the left-side of Figure 5.1) under the substitution and adjunction operations.

5.1.2 Combinatory Categorial Grammar

In combinatory categorial grammar, the supertag constitutes a CCG lexical category with a set of word-to-word dependencies. Clark and Curran (2004) used a MaxEnt model to estimate the probabilities for assigning lexical categories to each word of a

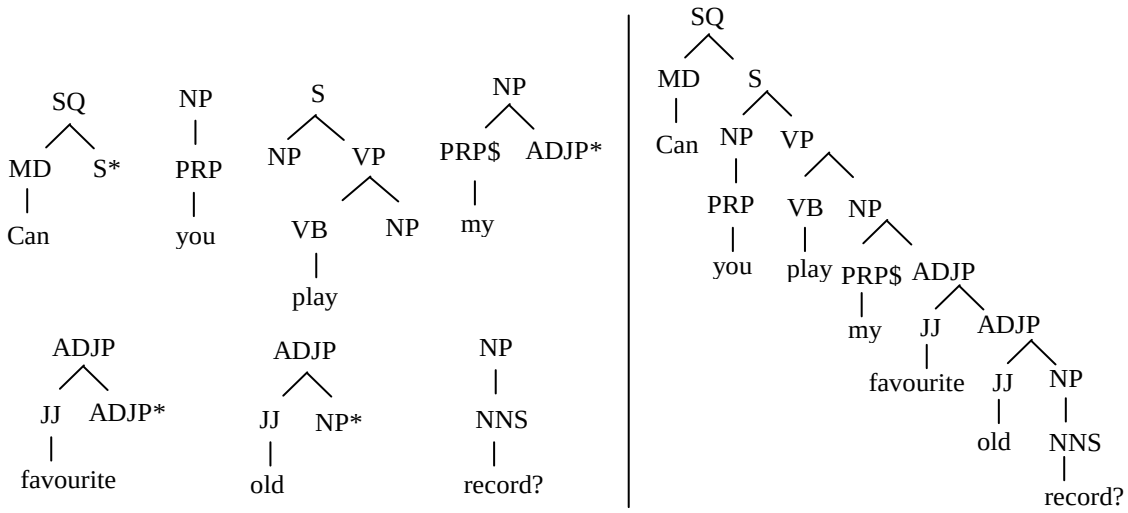

Figure 5.1: Example of LTAG supertags, which are combined under the operations of substitution and adjunction into a parse tree.

sentence. Each lexical category encapsulates its syntactic dependencies in different contexts. The CCG supertag of a word is related to the supertags of its neighbouring words which allows long-range word-to-word dependencies in a sentence to be captured in an indirect way.

A CCG lexical category may be either atomic (S, N, NP) or complex (S\S, S\NP, (S\NP)/NP). CCG supertags are combined under three types of operators (application operators, composition operators and type raising) (Hockenmaier, 2003) to form a parse tree.

Figure 5.2 shows the CCG supertags for the example sentence '*Can you play my favourite old record?*', which are assembled into a parse tree with forward and backward application operations. For an example, The right-hand side of Figure 5.2 shows that the word '*old*' is combined with '*record?*' under the operation of forward application. In other words, '*old*' can be thought of as a function that takes a category 'N' to its right and returns a category 'N'.

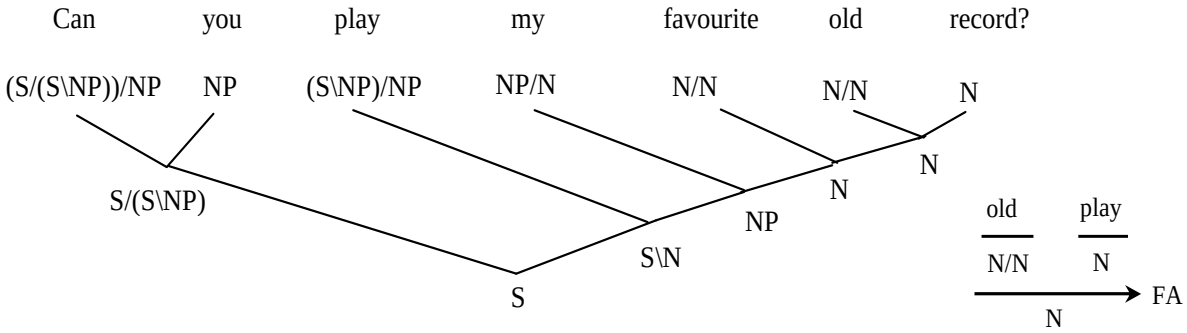

Figure 5.2: Example of CCG supertags, which are combined under the operations of forward and backward application into a parse tree.

5.1.3 Comparison of CCG and LTAG

In LTAG, a lexical item is associated with an elementary tree. On the other hand, CCG supertag constitutes a lexical category with a set of word-to-word dependencies. Both LTAG elementary trees and CCG lexical categories share a common property since both represent similar kind of arguments which preserve syntactic dependencies amongst the words in a sentence. Accuracies of LTAG and CCG supertaggers are quite similar (LTAG: 92% (Bangalore and Joshi, 1999); CCG: 92.39% (Clark and Curran, 2007)).

Despite the similarities between CCG and LTAG, these two approaches differ in following respects:

- The CCG supertag set is automatically extracted from CCGbank (Hockenmaier and Steedman, 2002) which is derived from the Penn-II Treebank. The number of tags in CCG tagset is 304 to 1,206 depending upon different frequency cut-offs. In contrast, Chen and Vijay-Shanker (2000) extract LTAG supertags automatically from the Penn-II Treebank with a different strategy.[1] The size of the tagset is 800 to 1800 depending upon different frequency cut-offs.

- Bangalore and Joshi (1999) used a standard Markov model tagger to assign

[1]In order to extract elementary trees from the Treebank, Chen and Vijay-Shanker (2000) made use of a head percolation table to identify head word of a node in the parse tree. They identified each node's status as complement or adjunct. According to Chen and Vijay-Shanker (2000), trunks of elementary trees are determined by finding paths in the parse tree using hand-built extraction rules.

LTAG elementary trees to words, while Clark and Curran (2004) used a Max-Ent model to assign lexical categories to each word of a sentence.

- LTAG elementary trees represent rigid structures, while CCG categories allow more flexibility in the derivation process. Hassan (2009) pointed out that the flexibility of CCG derivations allows for the handling of non-constituent constructions that LTAG cannot handle. This results in more spurious parse trees in CCG than LTAG.

5.2 Supertags as Context Information

We derive contextual information (CI) for a source phrase with supertags. Section 5.2.1 shows how CI for a PB-SMT source phrase is formed. In Section 5.2.2, we illustrate how CI for a Hiero source phrase is formed.

5.2.1 Context Information for PB-SMT

Like the CI_{pos} feature defined in Equation (4.2) (cf. page 58), we define the contextual information (CI_{st}) with supertags as in (5.1):

$$\text{CI}_{\text{st}}(\hat{f}_k) = \{\text{st}(f_{i_k-l}), ..., \text{st}(f_{i_k-1}), \text{st}(\hat{f}_k), \text{st}(f_{j_k+1}), ..., \text{st}(f_{j_k+l})\} \tag{5.1}$$

Similar to the CI_{pos} feature, we form the supertag for a multi-word focus phrase by concatenating the supertags of the words composing it. Thus, the supertag-based CI constitutes a window of size $2l + 1$ features. In our experiments, we consider context widths ± 1 and ± 2 (i.e. $l = 1, 2$) surrounding the focus phrase. For example, the CI of the focus phrase '*play*' in Figure 5.2 with CCG supertags is formed as: CI_{st} = {st(*you*), st(*play*), st(*my*)} = {NP, (S\NP)/NP, NP/N} (with $l = 1$). We also carried out experiments joining the two supertag types (CCG and LTAG) (cf. Section 5.3.1.1).

5.2.2 Context Information for Hierarchical PB-SMT

Like our CI_{pos} feature defined in Equation (4.4) (cf. page 61), we define the contextual information (CI_{st}) for a Hiero source phrase α with supertags as in (5.2):

$$\text{CI}_{\text{st}}(\alpha) = \{\text{st}(w_k)\} \tag{5.2}$$

where $\forall k \in [1, |\text{CI}_{\text{lex}}|] : w_k \in \text{CI}_{\text{lex}}$ (CI_{lex} is defined in Equation (4.3)).

Similar to the CI_{pos} (cf. Equation (4.4)) features, the supertag syntactic features form a window of size $2(l + s)$. We carried out a series of experiments by integrating the supertag context into the Hiero model. In addition, we combine the syntactic features with the lexical features. For instance, when supertags are combined with lexical features, the CI is formed by the union of these features, i.e. $\text{CI} = \text{CI}_{\text{st}} \cup \text{CI}_{\text{lex}}$.

5.3 Experiments with Context-Informed PB-SMT

Since we intend to use supertags as source-side contextual features, we chose English as the source language, given the availability of supertag information for this language. We carried out experiments by systematically applying supertag features on different language pairs and with varying sizes of training data. Similar to the division of the reports on the experiments with basic features in Section 4.3, we divide the reports on experiments with supertag features into seven subsections. Section 5.3.1 reports experimental results on small-scale data sets representing the language pairs English-to-Chinese, English-to-Hindi, and English-to-Czech. Section 5.3.2 reports experimental results on large-scale data sets representing the language pairs English-to-Dutch, English-to-Japanese, and English-to-Chinese. Section 5.3.3 presents some analysis of results obtained from the small- and large-scale translations. In Section 5.3.4, we demonstrate the outcomes of the learning curve experiments which we carried out on two different language pairs: English-to-Spanish and English-to-Dutch. In Section 5.3.6, we provide some analysis of the results

of the learning curve experiments. Section 5.3.7 compares context-dependent and context-independent phrase translation.[2]

5.3.1 Experiments on Small-Scale Data Sets

5.3.1.1 English-to-Chinese

The first set of experiments were carried out on IWSLT'06 English-to-Chinese training data (cf. Section 3.6) (Haque et al., 2009a).[3] As stated earlier in Section 4.3.1.1 (cf. page 62), we adopted a different experimental set-up in order to perform experiments on this data set.

	BLEU		NIST		WER		PER	
Baseline	*20.56*		*4.67*		*57.82*		*48.99*	
Context Size	±1	±2	±1	±2	±1	±2	±1	±2
CCG	21.75	21.52	4.84	4.79	56.28	56.95	48.58	49.10
LTAG	21.92 (95.1%)	21.34	4.82	4.70	56.63	57.61	48.43	49.27
Word+CCG	21.52	21.53	4.75	4.78	57.21	57.38	48.95	49.45
Word+LTAG	21.64 (96.2%)	21.37	4.78	4.79	57.15	57.06	48.89	48.95

Table 5.1: Experiments of English-to-Chinese translation with uniform context size using IGTree

The results with uniform context size are shown in Table 5.1, which clearly shows that adding supertags as source-side context improves PB-SMT baseline across all evaluation metrics. When LTAG supertags are added as an individual context feature, the system produces the highest BLEU score (a 1.36 BLEU point improvement; 6.61% relative increase) among all the systems. When CCG supertags are used as an individual context feature, moderate improvements can be seen (1.19 BLEU points; 5.79% relative). In both cases, a context window of ±1 appears to be more useful than that of ±2. Furthermore, we conducted experiments by applying the supertag and lexical features in collaboration, the results of which can be seen in the last two rows of the Table 5.1. Moderate improvements over the baseline are observed

[2]Parts of the experiments carried out with our context-informed PB-SMT model including learning curve experiments have been reported, albeit in a different form, in Haque et al. (2011).

[3]Experiments reported in this section have been published, albeit in different form, in (Haque et al., 2009a).

for Word+CCG (a 0.96 BLEU point improvement; 4.66% relative increase) and Word+LTAG (1.08 BLEU improvement; 5.25% relative increase) when a context window of ±1 is used.

Experiments	BLEU	NIST	WER	PER
Baseline	*20.56*	*4.67*	*57.82*	*48.99*
Word±2+CCG±1	22.01	4.82	57.21	48.63
Word±2+LTAG±1	21.38	4.79	57.01	48.89
POS±2+CCG±1	21.08	4.68	58.22	50.05
CCG±1+LTAG±1	21.79	4.74	58.28	49.59
CCG±1+LTAG±1†	22.11 (97.5%)	4.82	56.95	48.81
Word±1+CCG±1+LTAG±1†	21.48	4.79	56.83	48.53
Word±2+POS±2+CCG±1	21.23	4.72	57.47	49.82
Super-Pair±1†	21.99	4.82	56.83	48.72

Table 5.2: Experiments of English-to-Chinese translation with varying context size using IGTree. The symbol † indicates an experimental set-up in which we ignore the syntactic information of the source phrase.

The results obtained employing combinations of features with varying context sizes can be seen in Table 5.2. Here, adding CCG supertags to the neighbouring words caused the system performance to reach a new high of 22.01 BLEU score, 1.45 BLEU points (7.05% relative improvement) over the PB-SMT baseline. Encouragingly, the best performance of all was seen when both supertag features were used in combination. Here an even higher BLEU score of 22.11 (7.54% relative improvement over the baseline) was obtained for CCG±1+LTAG±1, when ignoring the syntactic feature information of the focus phrase.

	BLEU	NIST	WER	PER
Baseline	*20.56*	*4.67*	*57.82*	*48.99*
CCG±1	22.08 (97.1%)	4.83	57.30	48.63
LTAG±1	22.06	4.75	58.05	49.04
CCG±1+LTAG±1†	21.72	4.76	58.48	49.18
Super-Pair±1†	22.03	4.79	57.35	49.15

Table 5.3: Experiments of English-to-Chinese translation using IB1.

We also tested the best-performing set-ups on IB1 and TRIBL classifiers, the results of which are shown in Table 5.3 and 5.4, respectively. The differences we see between using IGTree, TRIBL, and IB1 are generally small and somewhat un-

predictable. When considered as a single concatenated feature, the supertag-pair (Super-Pair) performed best on TRIBL. When the supertags are used as a standalone feature, IB1 produced the best score on LTAG (7.3% relative improvement), and TRIBL on CCG (7.88% relatively better).

	BLEU	NIST	WER	PER
Baseline	*20.56*	*4.67*	*57.82*	*48.99*
CCG±1	22.18 (98.5%)	4.85	56.31	48.55
LTAG±1	21.39	4.78	56.83	48.72
CCG±1+LTAG±1†	22.00	4.75	58.16	49.59
Super-Pair±1†	22.13	4.80	57.24	48.92

Table 5.4: Experiments of English-to-Chinese translation using TRIBL.

We also carried out an analysis on the translations produced by our best-performing context-informed (CI) system (CCG±1 with TRIBL, cf. Table 5.4) and the Moses baseline. Table 5.5 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 5.6 compares weights of the various translational features of the CCG±1 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	31	17	56	382
Sentence-Level TER	102	87	263	34
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	8.93		7.81	
Matching Words (%)	58.45		57.44	

Table 5.5: Comparison between translations produced by the best-performing context-informed (CI) system (CCG±1) and Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mod}
Moses	0.2388	0.0686	0.0076	0.0477	0.0842	0.0507	0.7030	-
CCG±1	0.1851	0.0575	0.0116	-	0.0290	-0.0107	0.1851	0.0544

Table 5.6: Comparison of weights for each translational feature of the two systems (CCG±1 and Moses baseline) obtained by MERT training.

5.3.1.2 English-to-Hindi

English-to-Hindi experiments were carried out using the EILMT tourism corpus (cf. Section 3.6). The experimental results using the TRIBL classifier are displayed in

Table 5.7.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*10.93*	*4.54*	*28.59*	*74.87*	*82.06*	*56.67*
CCG±1	11.14	4.58	27.94	74.84	82.19	56.76
CCG±2	11.07	4.57	28.59	74.76	81.85	56.65
LTAG±1	11.19	4.55	28.28	74.67	81.48	56.78
LTAG±2	11.17	4.57	28.59	74.73	81.98	56.63
CCG-LTAG±1	11.01	4.53	28.73	75.34	82.62	56.89
CCG±1+LTAG±1†	11.04	4.55	28.73	74.94	82.14	56.66
Super-pair±1†	11.02	4.58	27.62	74.45	81.45	56.45
Super-pair±2†	11.15	4.58	28.27	74.87	82.22	56.45

Table 5.7: Experiments applying various supertag features in English-to-Hindi translation.

For the English-to-Hindi translation task, we copied the previously best-performing set-up obtained from the English-to-Chinese translation task. Experimental results in Table 5.7 show that among the various features, LTAG±1 produces the best improvement (0.26 BLEU points, 2.37% relative) over the baseline, but this improvement is not statistically significant. Other context-informed features also produce small but consistent improvements over the baseline in terms of BLEU. Nevertheless, these improvements are not statistically significant with respect to the baseline. The other evaluation metrics show similar improvements.

We also carried out an analysis on the translations produced by our best-performing context-informed (CI) system (LTAG±1) and the Moses baseline. Table 5.8 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 5.9 compares weights of the various translational features of the LTAG±1 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	67	57	22	339
Sentence-Level TER	158	146	191	0
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	0		0	
Matching Words (%)	51.54		51.86	

Table 5.8: Comparison between translations produced by the best-performing context-informed (CI) system (LTAG±1) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0980	0.0453	0.0652	0.0510	0.1948	0.0910	-0.1683	-	-
LTAG±1	0.0652	-0.0051	0.0603	0.0334	0.05419	0.0306	-0.2402	0.0421	0.0090

Table 5.9: Comparison of weights for each translational feature of the two systems (LTAG±1 and Moses baseline) obtained by MERT training.

5.3.1.3 English-to-Czech

For the English-to-Czech translation task (Penkale et al., 2010)[4] we employed the previously best performing set-up in terms of context width and feature combinations, and the TRIBL classifier. The evaluation results on the WMT 2009 test set are reported in Table 5.10. We observe that a small improvement over the Moses baseline is achieved for the CCG±1 feature in terms of BLEU. Moderate improvements in METEOR and TER are to be observed for all features except LTAG±1. The highest METEOR score over the baseline is obtained for the supertag pair features (Super-Pair: a 0.15 METEOR point improvement; 0.44% relative). On the TER evaluation metric, the best performing set-up, CCG±1, yields an absolute reduction of 0.42 TER points below the baseline. Similar trends are also observed for WER and PER metrics. Moreover, gains for the CCG±1 feature are seen across all evaluation metrics over the baseline.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*7.83*	*3.90*	*34.13*	*87.66*	*80.53*	*67.88*
CCG±1	7.88	3.95	34.23	87.24	80.15	67.39
LTAG±1	7.67	3.89	34.00	87.90	80.86	68.00
CCG-LTAG±1	7.80	3.90	34.35	88.24	81.17	68.16
Super-Pair±1	7.82	3.90	34.38	87.96	80.84	68.18

Table 5.10: Supertag-based experimental results on the WMT 2009 test set.

Experimental results on the WMT 2010 test set are shown in Table 5.11. We observe that the improvements on this test set are similar to the improvements obtained on the WMT 2009 test set. CCG±1 yields the highest improvements across all evaluation metrics except for METEOR. As far as the METEOR evaluation metric is concerned, the Super-Pair±1 feature produces the highest improvement

[4]Parts of experimental results in the English-to-Czech translation task have been summarized in Penkale et al. (2010).

(a 0.32 METEOR point gain, 0.92% relative) over the baseline. CCG±1 yields a 0.21 BLEU point gain (2.68% relative increase) and a 0.43 TER point reduction compared to the baseline.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*8.05*	*3.97*	*34.61*	*86.01*	*78.54*	*67.48*
CCG±1	8.26	4.02	34.76	85.58	78.06	66.96
LTAG±1	8.00	3.95	34.57	86.41	78.95	67.72
CCG-LTAG±1	8.09	3.96	34.90	86.62	79.18	67.91
Super-Pair±1	8.11	3.95	34.93	86.62	79.05	68.08

Table 5.11: Supertag-based experimental results on the WMT 2010 test set.

In summary, slight improvements over the baseline are seen for supertag features, but none of the improvements over the baseline models are statistically significant in terms of BLEU.

We also carried out an analysis on the translations produced by our best-performing system (CCG±1) and the Moses baseline (with WMT 2010 test set). Table 5.12 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 5.13 compares weights of the various translational features of the CCG±2 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	212	161	124	1992
Sentence-Level TER	630	533	1320	6
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	0.24		0.28	
Matching Words (%)	30.27		30.73	

Table 5.12: Comparison between translations produced by the best-performing context-informed (CI) system (CCG±2) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1017	0.0405	0.0265	0.0550	0.0222	0.2377	-0.1147	-	-
CCG±1	0.1029	0.0364	0.0099	-0.0472	0.0682	0.0989	-0.2044	0.0676	0.0001

Table 5.13: Comparison of weights for each translational feature of the two systems (CCG±1 and Moses baseline) obtained by MERT training.

5.3.2 Experiments on Large-Scale Data Sets

5.3.2.1 English-to-Dutch

We carried out a similar series of experiments to those that we reported in the previous section to see whether similar improvements can be achieved with large-scale data sets. The first experimental data set was English-to-Dutch Europarl data (cf. Section 3.6). Analogous to the experiments on small-scale data sets, we experimented with adding contextual information features representing supertags and their combinations. We used the IGTree classifier to carry out these experiments.

Experimental results are displayed in Table 5.14. As can be seen from the table, CCG-LTAG±1 yields the highest improvement (0.38 BLEU points; 1.57% relative) over the baseline, which is statistically significant at the 96% level of confidence. Other context-informed features also produce consistent improvements over the baseline in terms of BLEU.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*24.26*	*6.177*	*52.68*	*64.37*	*68.81*	*50.02*
CCG±1	24.58	6.229	52.46	63.79	68.2	49.85
LTAG±1	24.33	6.267	52.51	63.53	68.00	49.13
CCG-LTAG±1†	24.45	6.250	52.54	63.87	68.30	49.74
CCG-LTAG±1	24.64 (96%)	6.235	52.79	63.90	68.27	49.58
Super-Pair±1†	24.35	6.184	52.31	64.45	68.71	50.32
Super-Pair±1	24.34	6.224	52.70	64.03	68.42	49.6

Table 5.14: Results on English-to-Dutch Translation employing supertag features.

In this translation task, we carried out an analysis on the translations produced by our best-performing context-informed (CI) system (CCG-LTAG±1) and the Moses baseline. Table 5.15 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Table 5.16 compares weights of the various translational features of the CCG-LTAG±1 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	271	256	157	316
Sentence-Level TER	338	288	340	36
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	3.8		3.8	
Matching Words (%)	61.96		61.74	

Table 5.15: Comparison between translations produced by the context-informed (CI) system (CCG-LTAG±1) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1072	0.0102	0.0509	0.1103	0.0468	0.0936	-0.2689	-	-
Word±2	0.0840	0.0224	0.0274	0.0325	0.0247	0.139	-0.1706	0.0094	0.0073

Table 5.16: Comparison of weights for each translational feature of the two systems (CCG-LTAG±1 and Moses baseline) obtained by MERT training.

5.3.2.2 English-to-Japanese

Our next set of experiments were carried out on a large-scale English-to-Japanese data set (cf. Section 3.6). As with the other large-scale experiments, the experiments were carried out using IGTree classifiers. Experimental results are shown in Table 5.17. System performance was evaluated using two different test sets. None of the contextual features are able to improve on the Moses baseline with any of the test sets.[5]

Experiments	BLEU	NIST	TER	WER	PER
Baseline	*27.30*	*6.746*	*63.31*	*80.01*	*43.36*
Evaluation Results on EJTestset1					
CCG±1	27.11	6.722	63.97	80.40	43.84
LTAG±1	27.18	6.736	63.53	80.06	43.51
CCG-LTAG±1	27.13	6.690	64.19	80.81	44.04
Super-Pair±1	27.10	6.727	63.84	80.44	43.59
Evaluation Results on EJTestset2					
Baseline	*27.76*	*6.838*	*60.64*	*77.49*	*42.61*
CCG±1	27.41	6.768	61.49	78.38	43.23
LTAG±1	27.37	6.771	61.19	78.04	43.13
CCG-LTAG±1	27.31	6.734	61.68	78.51	43.27
Super-Pair±1	27.40	6.773	61.19	78.29	43.14

Table 5.17: Experimental results for large-scale English-to-Japanese translation.

We also carried out an analysis on the translations produced by the LTAG±1

[5]The contents in this section have been published, albeit in a different form, in Okita et al. (2010).

and the Moses baseline (with EJTestset2). Table 5.18 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 5.19 compares weights of the various translational features of the LTAG±1 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	354	387	64	122
Sentence-Level TER	303	393	231	0
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	0		0	
Matching Words (%)	69.34		69.86	

Table 5.18: Comparison between translations produced by the context-informed (CI) system (LTAG±1) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0739	0.0323	0.0292	0.0558	0.0418	0.0739	-0.1977	-	-
LTAG±1	0.0734	0.0182	0.0446	0.05279	0.0515	0.0398	-0.2274	0.0434	-0.0421

Table 5.19: Comparison of weights for each translational feature of the two systems (LTAG±1 and Moses baseline) obtained by MERT training.

5.3.2.3 English-to-Chinese

Our next set of experiments were carried out on an English-to-Chinese data set (cf. Section 3.6). Experiments were carried out with two types of supertags (CCG±1, LTAG±1), using both IGTree and TRIBL.

Experimental results are displayed in Table 5.20. When IGTree classifiers were used, the LTAG±1 feature does not show any improvement over the Moses baseline for any of the evaluation metrics, while CCG±1 shows a slight improvement only in BLEU.

On the other hand, when we use TRIBL classifiers, CCG±1 yields a 0.37 BLEU points improvement (3.76% relative increase) over the baseline, a statistically significant improvement at 99.3% level of confidence. LTAG±1 produces the highest BLEU improvement (0.54 BLEU points; 5.48% relative) over the baseline, and the improvement is statistically significant at 99.9% level of confidence.

Experiments	BLEU	NIST	TER	WER	PER
Baseline	*9.85*	*4.87*	*77.13*	*82.03*	*60.29*
IGTree					
CCG±1	9.91	4.83	77.98	82.75	61.31
LTAG±1	9.80	4.82	77.71	82.58	61.12
TRIBL					
CCG±1	10.22 (99.3%)	4.96	76.68	81.65	59.76
LTAG±1	10.39 (99.9%)	4.89	76.92	81.82	60.20

Table 5.20: Experimental results for large-scale English-to-Chinese translation.

In sum, while for large-scale English-to-Japanese translation none of the contextual features showed any improvement over the baseline, for large-scale English-to-Chinese translation a slight improvement in terms of BLEU was observed for the CCG supertag context when IGTree was used for the classification task. We also carried out experiments using TRIBL as the classifier, and achieved statistically significant improvements over the baseline for both the CCG and LTAG supertag features.

In this translation task, we also carried out an additional analysis on the translations produced by the best-performing system (LTAG±1 with TRIBL) and the Moses baseline. Table 5.21 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 5.22 compares weights of the various translational features of the LTAG±1 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	178	117	95	967
Sentence-Level TER	377	341	637	4
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	0.29		0.37	
Matching Words (%)	52.15		51.98	

Table 5.21: Comparison between translations produced by the context-informed (CI) system (LTAG±1) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0805	0.0011	0.0760	0.0704	0.0086	-0.0020	-0.1972	-	-
LTAG±1	0.0821	0.0012	0.0638	0.0454	0.0118	0.0337	-0.2202	0.0332	0.0003

Table 5.22: Comparison of weights for each translational feature of the two systems (LTAG±1 and Moses baseline) obtained by MERT training.

5.3.3 Effect of Small vs Large-Scale Data Sets

First, we compare the effectiveness of supertag contextual features, both collectively and individually, on all small-scale translation tasks. We also provide a contrastive overview in which we compare the impact of supertag contexts with that of basic contexts (cf. Section 4.3.1).

On IWSLT English-to-Chinese translation, the highest evaluation scores over the baseline are achieved employing supertags and their combinations. For English-to-Hindi, supertag as a source-side context gives moderate improvements over the baseline, but neighbouring words appear to be the most effective context. For English-to-Czech, slight improvements over the baseline are seen for supertags, but POS and word contexts do not improve the baseline at all. None of the improvements over the baseline models on both the English-to-Hindi and the English-to-Czech translation tasks are statistically significant in terms of BLEU.

Analyzing the outcomes of small-scale translation tasks, we see that supertags seem to be the most effective context features, as compared to neighbouring words and part-of-speech. Arguably, one can surmise that the differences in word order between the source and target languages in our experiments are best treated by a feature which is not entirely restricted to local context.

We can draw better conclusions from the outcomes of large-scale translation tasks. For large-scale English-to-Dutch translation, improvements provided by supertag-based SMT model are statistically significant in terms of BLEU at the 96% level of confidence. In contrast, improvements for the word context are not statistically significant, and the POS-based model performs below the baseline PB-SMT model.

For large-scale English-to-Japanese translation, none of the contextual features showed any improvement over the baseline. For large-scale English-to-Chinese translation, a slight improvement in BLEU over the baseline was observed for the CCG supertag context, when IGTree was used for the classification task. For this language pair, we also carried out experiments with TRIBL as the classifier, and achieved modest improvements over the baseline for the both CCG and LTAG supertag con-

texts.

Comparing the effectiveness of the classifiers on large-scale translation tasks, we observed that IGTree proved to be useful for English-to-Dutch, but not for English-to-Japanese or English-to-Chinese. In contrast, TRIBL was effective for this latter language direction. In terms of contextual features, overall supertags seemed to be more effective than words and POS tags.

5.3.4 Experiments on Increasing Size of Training Sets

We carried out learning curve experiments considering basic features (words and POS tags) as a source-language context on three different language pairs (English-to-Spanish, English-to-Dutch, and Dutch-to-English), the results of which were reported in Section 4.3.5. We carried out learning curve experiments considering supertags as source-language context on English-to-Spanish and English-to-Dutch language pairs. For that purpose we used the same data sets that we used to perform learning curve experiments with basic contextual features. We report the outcomes of English-to-Spanish and English-to-Dutch learning curve experiments with supertag contextual features in Sections 5.3.4.1 and 5.3.4.2, respectively.

5.3.4.1 English-to-Spanish

As mentioned in Section 4.3.5.1, the English-to-Spanish training data set was divided into eight different training sets ranging from 10K sentence pairs to 1.64M sentence pairs. To perform experiments on this sequence of training sets, we used both IGTree and TRIBL. As stated in Chapter 4 (cf. page 76), we were initially able to use the TRIBL classifier with training sets containing only up to 100K sentences due to TRIBL's relatively high memory requirements.

IGTree as classifier: Experimental results on English-to-Spanish translation employing IGTree as the classifier are displayed in Table 5.23, which shows experimental results obtained on training sets comprising 10K to 1.64M sentence pairs. On the

training set containing 1.64M sentence pairs, we were not able to perform the experiment for CCG-LTAG±1 set-up since the memory requirement for building the IGTree classifier for that exceeded the limit of our computer memory (132G). As can be seen from Table 5.23, for all amounts of training data except the largest one (1.64M training data), supertag-based SMT systems remain below the Moses baseline according to the performance measured by any of the evaluation metrics. On the training set containing 1M sentence pairs, the performance of the PB-SMT models with supertag context is very close to the performance of the Moses baseline. Experimental results on the training set of 1.64M sentence pairs show similar characteristics to the results obtained on the 1M training set with a few exceptions. LTAG±1 and CCG±1 improve the baseline BLEU and METEOR scores, respectively, when the largest amount of training data is used.

TRIBL as Classifier: The experimental results obtained on the training sets containing 10K to 100K sentence pairs using TRIBL as the classifier are shown in Table 5.24. When the 10K training set is used, the CCG±1 system provides slight improvements over the baseline across most evaluation metrics. We also see that the performance of the other context-informed SMT models are very close to that of the Moses baseline.

While using 20K sentence-pairs of training data, all context-informed SMT systems (CCG±1, LTAG±1, CCG-LTAG±1, Super-Pair±1) produce moderate improvements over the PB-SMT baseline across most evaluation metrics. However, none of the BLEU and METEOR improvements with respect to the baseline are statistically significant.

Adding any of the supertag contextual features improves upon the baseline across all the evaluation metrics, when we used the training set of 50K sentence pairs. We see from Table 5.24 that the best BLEU improvement (0.25 BLEU points; 0.92% relative) over the Moses baseline are achieved for CCG±1. The improvement in BLEU for CCG±1 with respect to the baseline is statistically significant. More-

Train. Size	Experiments	BLEU	NIST	METEOR	TER	WER	PER
10K	Baseline	*22.68*	*6.00*	*26.15*	*68.93*	*67.58*	*50.81*
	CCG±1	21.84	5.88	25.5	69.84	68.5	51.44
	LTAG±1	21.76	5.89	25.58	69.78	68.44	51.44
	CCG-LTAG±1	21.51	5.84	25.42	70.01	68.59	51.57
	Super-Pair±1	21.61	5.85	25.51	70.17	68.77	51.69
20K	Baseline	*24.58*	*6.38*	*27.77*	*66.51*	*65.16*	*48.71*
	CCG±1	23.93	6.3	2736	67.27	66.16	49.19
	LTAG±1	23.89	6.27	27.26	67.41	66.18	49.49
	CCG-LTAG±1	23.84	6.25	27.28	67.59	66.38	49.55
	Super-Pair±1	23.85	6.25	27.15	67.67	66.64	49.56
50K	Baseline	*27.33*	*6.84*	*29.93*	*64.15*	*62.97*	*46.63*
	CCG±1	26.73	6.75	29.44	64.50	63.44	46.97
	LTAG±1	26.57	6.73	29.37	64.91	63.83	47.28
	CCG-LTAG±1	26.38	6.70	29.34	65.05	63.87	47.35
	Super-Pair±1	26.65	6.74	29.43	64.56	63.54	47.11
100K	Baseline	*28.64*	*7.09*	*30.91*	*62.30*	*61.26*	*45.1*
	CCG±1	28.43	7.04	30.83	62.8	61.71	45.09
	LTAG±1	28.19	7.04	30.85	62.8	61.66	45.62
	CCG-LTAG±1	28.31	7.03	30.75	62.89	61.93	45.65
	Super-Pair±1	28.38	7.06	30.87	62.58	61.53	45.46
200K	Baseline	*29.96*	*7.32*	*31.90*	*60.97*	*60.01*	*44.06*
	CCG±1	29.57	7.23	31.55	61.62	60.71	44.61
	LTAG±1	29.5	7.25	31.67	61.52	60.50	44.54
	CCG-LTAG±1	29.39	7.23	31.55	6147	60.54	44.62
	Super-Pair±1	29.45	7.23	31.54	6162	60.73	44.61
500K	Baseline	*31.08*	*7.47*	*32.67*	*59.86*	*58.83*	*43.18*
	CCG±1	30.72	7.44	32.53	60.24	59.36	43.56
	LTAG±1	30.80	7.44	32.47	60.20	59.32	43.42
	CCG-LTAG±1	30.61	7.44	32.49	60.18	59.40	43.40
	Super-Pair±1	30.86	7.44	32.57	60.11	59.26	43.38
1M	Baseline	*31.52*	7.54	32.94	59.45	58.50	42.84
	CCG±1	31.35	7.52	32.85	59.41	58.53	42.86
	LTAG±1	31.34	7.50	32.81	59.66	58.67	43.06
	CCG-LTAG±1	31.31	7.51	32.81	59.63	58.67	43.07
	Super-Pair±1	31.37	7.50	32.76	59.71	58.69	43.16
1.64M	Baseline	*31.92*	*7.60*	*33.24*	*59.06*	*58.14*	*42.42*
	CCG±1	31.87	7.58	33.25	59.12	58.17	42.61
	LTAG±1	31.93	7.57	33.21	59.36	58.37	42.77
	Super-Pair±1	31.71	7.56	33.01	59.20	58.30	42.71

Table 5.23: Results of English-to-Spanish learning curve experiments with IGTree classifier.

Train. Size	Experi-ments	BLEU	NIST	METEOR	TER	WER	PER
	Baseline	*22.68*	*6.00*	*26.15*	*68.93*	*67.58*	*50.81*
	CCG±1	22.66	6.02	26.21	68.84	67.41	50.66
10K	LTAG±1	22.67	6.00	26.15	69.06	67.71	50.89
	CCG-LTAG±1	21.51	5.84	25.42	70.01	68.59	51.57
	Super-Pair±1	21.61	5.85	25.51	70.17	68.77	51.69
	Baseline	*24.58*	*6.38*	*27.77*	*66.51*	*65.16*	*48.71*
	CCG±1	24.57	6.39	27.74	66.76	65.70	48.78
20K	LTAG±1	24.57	6.40	27.85	66.57	65.39	48.65
	CCG-LTAG±1	24.66	6.41	27.85	66.54	65.32	48.70
	Super-Pair±1	24.70 (72%)	6.40	27.80	66.57	65.25	48.71
	Baseline	*27.33*	*6.84*	*29.93*	*64.15*	*62.97*	*46.63*
	CCG±1	27.58 (97.7%)	6.90	30.08 (98.1%)	63.53	62.38	46.29
50K	LTAG±1	27.45 (88%)	6.89	29.98 (74.8%)	63.66	62.66	46.18
	CCG-LTAG±1	27.44 (92.7%)	6.89	30.02 (86.1%)	63.57	62.72	46.16
	Super-Pair±1	27.56 (93.9%)	6.87	30.04 (91.1%)	63.77	62.60	46.52
	Baseline	*28.64*	*7.09*	*30.91*	*62.30*	*61.26*	*45.1*
	CCG±1	29.02 (96.9%)	7.14	31.18 (99.9%)	62.25	61.21	45.09
100K	LTAG±1	29.07 (99.5%)	7.15	31.21 (99.9%)	62.06	61.04	44.93
	CCG-LTAG±1	29.11 (99.5%)	7.17	31.32 (99.9%)	61.88	60.89	44.86
	Super-Pair±1	29.11 (99.5%)	7.16	31.32 (99.9%)	61.88	60.84	44.89

Table 5.24: Results of English-to-Spanish learning curve experiments using TRIBL as the classifier.

over, improvements in BLEU for the remaining context-based systems (LTAG±1, CCG-LTAG±1, Super-Pair±1) with respect to the baseline are very close to the significance level. As far as METEOR is concerned, CCG±1 produces a statistically significant improvement (0.15 METEOR points; 0.57% relative) over the baseline.

When the 100K training set is used, all of the supertag features provide modest gains over the Moses baseline across all evaluation metrics. Both Super-Pair±1 and CCG-LTAG±1 produce the highest BLEU (0.47 BLEU points improvement; 1.65% relative) and METEOR (0.41 METEOR points improvement; 1.33% relative) improvements over the baseline, both of which are statistically significant with respect to the baseline.

Learning Curves: With learning curves, here we present a more analytical study of the effect of increasing amounts of training data, for both IGTree and TRIBL classifiers. We plot the BLEU score learning curves of the two best-performing context-informed models (LTAG±1, Super-Pair±1) for TRIBL and IGTree, as well as the Moses baseline in Figure 5.3. The curves of IGTree extend up to the maximum

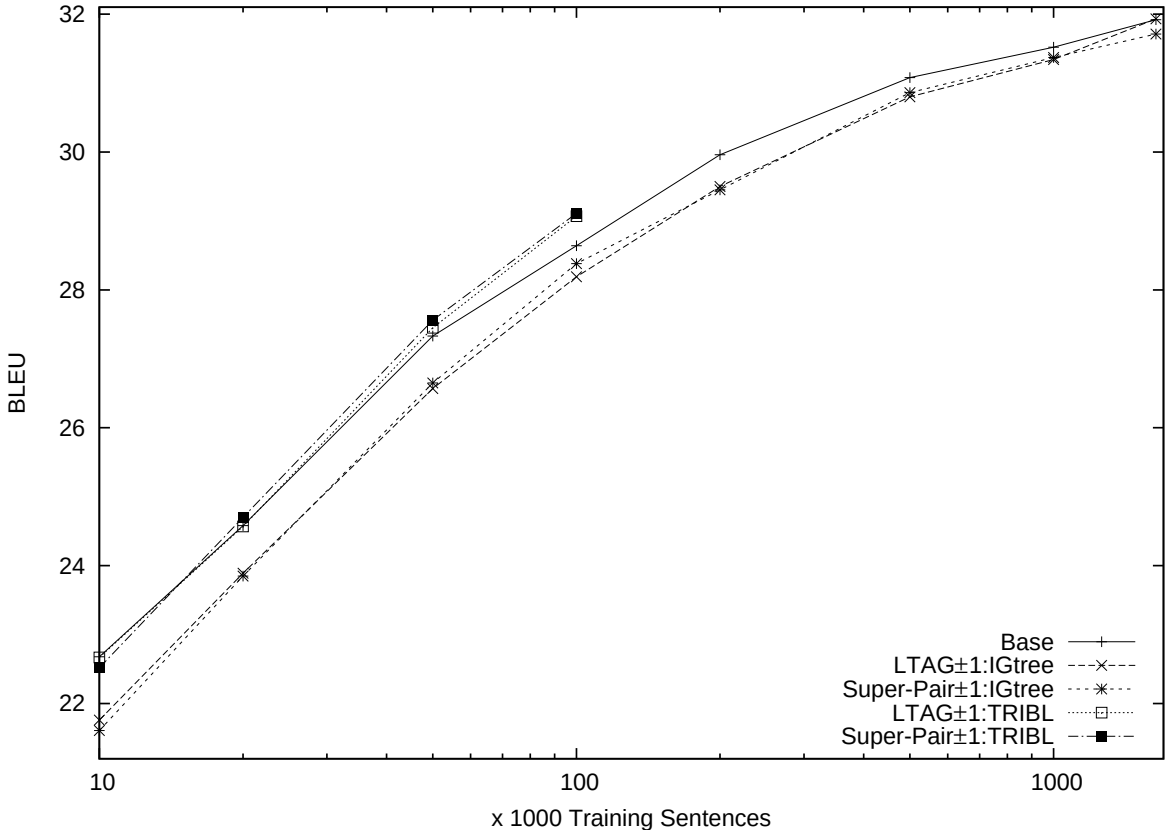

Figure 5.3: BLEU Learning curves comparing the Moses baseline against supertag-based SMT models in English-to-Spanish translation task.

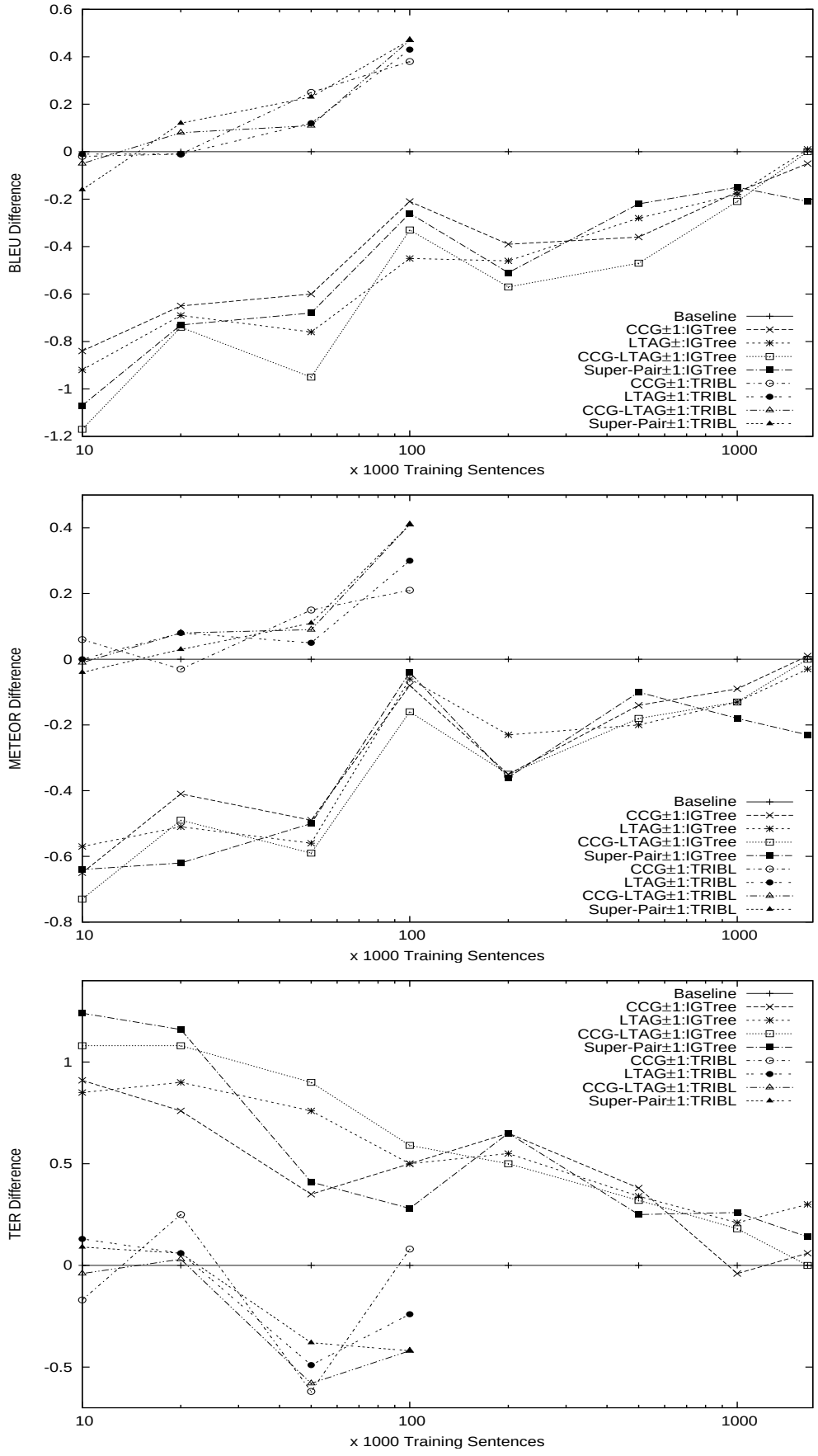

Figure 5.4: BLEU (top), METEOR (centre) and TER (bottom) score-difference curves comparing the Moses baseline against supertag-based SMT models in English-to-Spanish translation task.

of 1.64M training sentences; as noted, due to limitations in memory, the TRIBL experiments extend only up to 100K training sentences.

In addition, Figure 5.4 shows the BLEU (top), METEOR (centre) and TER (bottom) difference curves of the supertag-based experiments against the baseline, highlighting the gains and losses against the baseline. In addition to the two context-based models (LTAG±1, Super-Pair±1) and the Moses baseline, this figure displays the performances of the CCG±1 and the CCG-LTAG±1 against the baseline.

Figure 5.3 shows that the LTAG±1 and Super-Pair±1 curves of TRIBL start just below the baseline curve, then improve over the baseline curve. The figure also illustrates that the LTAG±1 and Super-Pair±1 curves of IGTree start at a lower level than the baseline curve, and end at the same level as the baseline curve at the largest training set size.

We summarize the outcomes of the English-to-Spanish translation task. TRIBL appears to be effective on both small and moderately large-scale data sets. In contrast, IGTree does not offer improvements over the baseline either with the small or the large-scale context-informed models. The performance of the large-scale context-informed models with the IGTree classifier are comparable to that of the Moses baseline.

As an additional point of analysis, Figure 5.5 compares the Moses baseline with both TRIBL and IGTree using the CCG±1 feature, in terms of the average number of target phrases considered for a source phrase for varying training data sizes.

The graph in Figure 5.5 shows that the TRIBL curve lies between the IGTree curve and the Moses baseline curve; the Moses baseline uses an increasing number of target phrases with more training data, reaching an average of several hundreds of phrases at the maximal training set sizes. The TRIBL curve starts close to the IGTree curve, but rises at 100K training sentences; nevertheless, the TRIBL curve remains below the baseline curve. Thus, both the TRIBL and IGTree classifiers produce smaller, more constrained distributions of the target phrases given a source phrase and its context information.

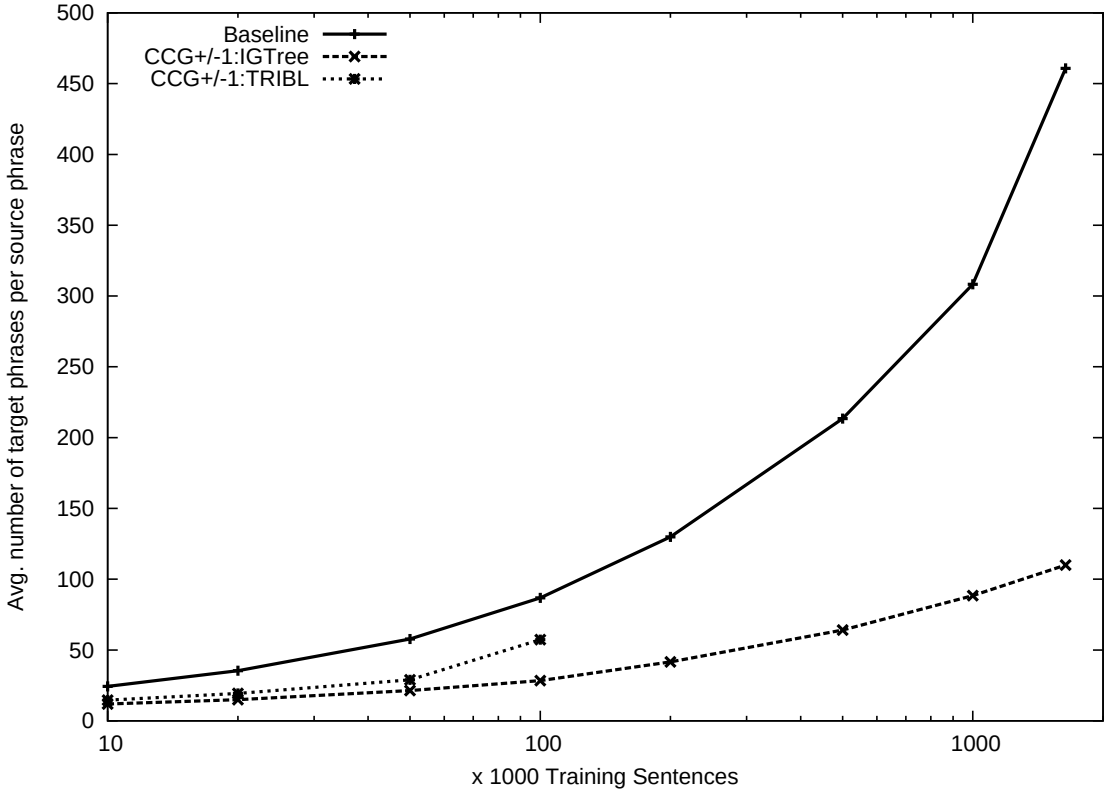

Figure 5.5: Average number of target phrase distribution sizes for source phrases for TRIBL and IGTree compared to the Moses baseline.

In this translation task, we carried out an additional analysis on the translations produced by our best-performing context-informed (CI) system (CCG-LTAG±1)[6] and the Moses baseline. Table 5.25 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 5.26 compares weights of the various translational features of the CCG-LTAG±1 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	603	538	356	503
Sentence-Level TER	595	540	828	37
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	2.1		2	
Matching Words (%)	60.01		59.65	

Table 5.25: Comparison between translations produced by the best-performing context-informed (CI) system (CCG-LTAG±1) and the Moses baseline.

[6]The best-performing CCG-LTAG±1 system which we used for this analysis was built on 100K training set with TRIBL (cf. Table 5.24).

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0795	0.0448	0.0091	0.0449	0.0527	0.1151	-0.1451	-	-
CCG-LTAG±1	0.1053	0.0504	0.0695	0.0012	0.0489	0.0880	-0.1333	0.0296	0.0001

Table 5.26: Comparison of weights for each translational feature of the two systems (CCG-LTAG±1 and Moses baseline) obtained by MERT training.

5.3.4.2 English-to-Dutch

In this section, we report the outcomes of the English-to-Dutch learning curve experiments. In this translation task, we adopted our previously best-performing set-ups in order to carry out experiments with supertag contextual models. We took in total four experimental set-ups, two (CCG±1, LTAG±1) representing individual supertag features, and the remaining two (CCG-LTAG±1, Super-Pair±1) combinations of supertag features.

In order to perform learning curve experiments with supertag contextual features, we used the same English-to-Dutch data set which we used to conduct the learning-curve experiments with basic contextual features (cf. Section 4.3.5.3). TRIBL is used as the classifier, as in the learning-curve experiments with basic contextual features.

The experimental results obtained on the training sets containing 10K to 1.31M sentence pairs are shown in Table 5.27, where statistically significant improvements are observed when the training data contains 100K sentence pairs or more. When larger amounts (1M and 1.33M) of training data are used, both CCG and LTAG supertags as individual context features provide statistically significant improvements in BLEU over the Moses baseline. Supertag combinations (CCG-LTAG±1, Super-Pair±1) produce moderate gains over the baseline, which are very close to the significance level.

We observe the effect of increasing amounts of training data by drawing learning curves. Figure 5.6 illustrates BLEU learning curves comparing the Moses baseline against the two context-informed SMT models that are based on two different types of supertag features: the individual contextual feature (LTAG±1) and combined contextual feature (Super-Pair±1). We see from Figure 5.6 that the BLEU learning

Train. Size	Experiments	BLEU	NIST	METEOR	TER	WER	PER
	Baseline	*17.20*	*4.99*	*43.74*	*72.40*	*75.39*	*57.17*
	CCG±1	16.47	4.98	43.08	72.19	75.49	57.05
10K	LATG±1	16.92	4.99	43.49	72.09	75.23	57.26
	CCG-LTAG±1	16.96	4.96	43.47	72.60	75.59	57.81
	Super-Pair±1	16.75	4.95	43.47	72.78	75.81	57.46
	Baseline	*19.03*	*5.31*	*46.30*	*70.33*	*73.69*	*55.08*
	CCG±1	18.73	5.35	45.82	70.1	73.52	54.91
20K	LATG±1	19.02	5.35	46.16	69.83	73.35	55.12
	CCG-LTAG±1	19.09 (46%)	5.31	46.20	70.30	73.60	55.39
	Super-Pair±1	18.76	5.29	46.23	70.67	74.09	55.21
	Baseline	*21.70*	*5.74*	*49.31*	*67.32*	*71.19*	*52.48*
	CCG±1	21.22	5.77	48.61	67.15	70.82	52.45
50K	LATG±1	21.62	5.80	49.16	66.74	70.57	52.23
	CCG-LTAG±1	21.57	5.74	49.21	67.40	71.24	52.80
	Super-Pair±1	21.57	5.75	49.27	67.48	71.27	52.53
	Baseline	*22.53*	*5.88*	*50.30*	*66.42*	*70.48*	*51.84*
	CCG±1	22.20	5.96	50.04	65.51	69.82	51.22
100K	LATG±1	22.95 (99.7%)	6.01	50.70 (99.48%)	65.00	69.33	50.87
	CCG-LTAG±1	22.61 (68%)	5.93	50.55 (98.8%)	65.80	70.00	51.44
	Super-Pair±1	22.58 (41%)	5.92	50.60 (94.9%)	66.00	69.95	51.28
	Baseline	*23.47*	*6.04*	*51.46*	*65.30*	*69.40*	*50.90*
	CCG±1	23.60 (98.7%)	6.16	51.50 (70.8%)	64.35	68.28	50.05
200K	LATG±1	23.45	6.11	51.33	64.55	68.64	50.36
	CCG-LTAG±1	23.63 (52%)	6.08	51.55 (67%)	65.14	68.93	50.64
	Super-Pair±1	23.60 (58.1%)	6.07	51.57 (77.7%)	65.12	69.16	50.71
	Baseline	*24.06*	*6.11*	*52.06*	*64.59*	*68.58*	*50.36*
	CCG±1	23.96	6.22	52.00	63.63	67.84	49.67
500K	LATG±1	23.97	6.19	51.94	63.87	67.99	49.94
	CCG-LTAG±1	23.83	6.12	52.16 (73.8%)	64.65	68.82	50.29
	Super-Pair±1	24.14 (53%)	6.15	52.30 (95.6%)	64.33	68.52	50.05
	Baseline	*24.26*	*6.17*	*52.39*	*64.55*	*68.72*	*50.12*
	CCG±1	24.50 (99.5%)	6.29	52.36	63.25	67.59	49.23
1M	LATG±1	24.40 (96.7%)	6.25	52.37	63.52	67.77	49.35
	CCG-LTAG±1	24.30 (40%)	6.19	52.41 (59.4%)	64.19	68.34	49.72
	Super-Pair±1	24.31 (58%)	6.2	52.57 (92.3%)	63.95	68.32	49.71
	Baseline	*24.26*	*6.17*	*52.68*	*64.36*	*68.80*	*50.02*
	CCG±1	24.65 (99.9%)	6.34	52.74 (80.5%)	62.86	67.26	48.80
1.31M	LATG±1	24.80 (99.9%)	6.3	52.67	63.2	67.6	49.17
	CCG-LTAG±1	24.49 (80.7%)	6.22	52.73 (64%)	63.92	68.34	49.63
	Super-Pair±1	24.56 (93.8%)	6.23	52.79 (79.11%)	63.92	68.25	49.71

Table 5.27: Results of the English-to-Dutch learning curve experiments with TRIBL classifier comparing the effect of supertag context and Moses baseline.

curves of the supertag-based SMT systems (LTAG±1, Super-Pair±1) start close to the baseline curve, go upward and cross the baseline curve when more training data is added.

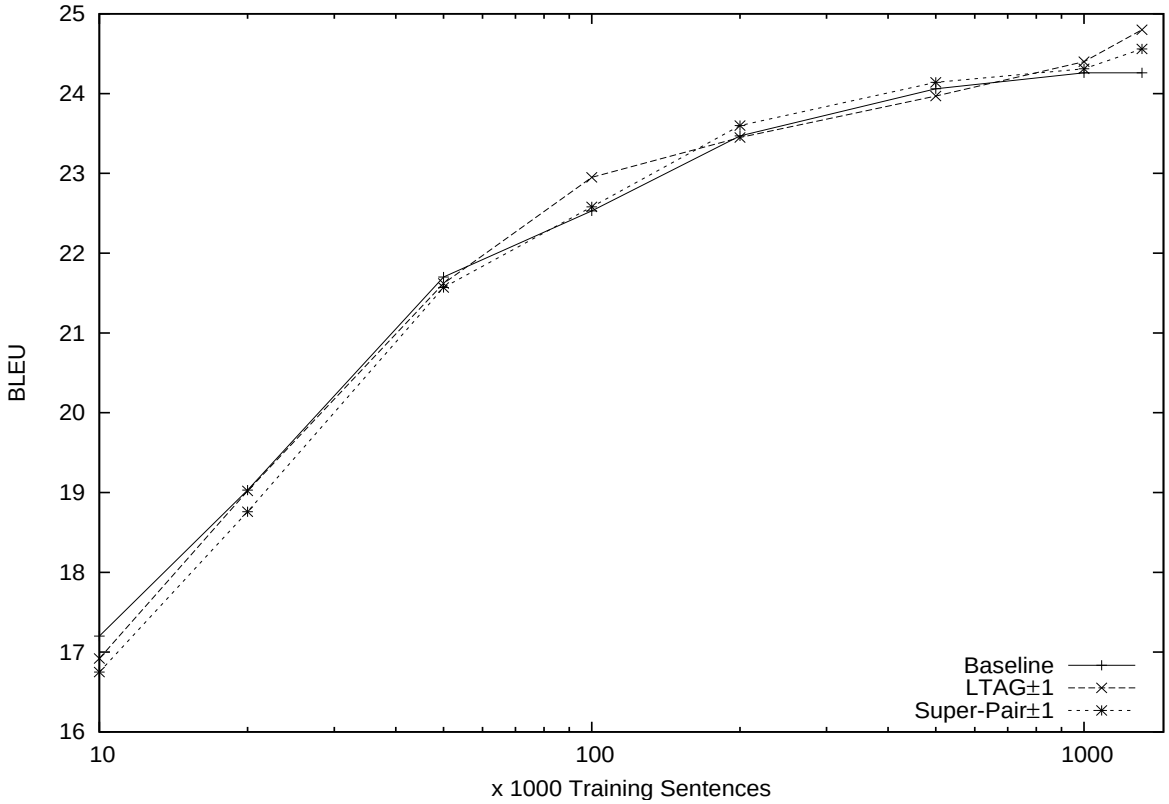

Figure 5.6: BLEU Learning curves comparing the Moses baseline against the supertag-based SMT models in English-to-Dutch translation task.

In addition to the BLEU learning curve shown in Figure 5.6, Figure 5.7 shows BLEU (top graph), METEOR (centre graph) and TER (bottom graph) score-difference curves of four supertag-based SMT systems (CCG±1, LTAG±1, CCG-LTAG±1, Super-Pair±1), showing the gains and losses against the Moses baseline. LTAG±1 and CCG±1 produced respectively the highest and second highest BLEU scores for the larger amounts of training data. We found that most of the BLEU improvements with respect to the baseline are statistically significant. From the central graph in Figure 5.7 which shows METEOR score-difference curves of four supertag-based SMT systems against the Moses baseline, we see that most of the curves do not resemble the BLEU score-difference curves (top graph of Figure 5.7). Interestingly, METEOR score-difference curves of the CCG±1 and the LTAG±1

systems reside mostly beneath the baseline curve.

The bottom graph in Figure 5.7 shows the TER score-difference curves where we see that most such curves show consistency in residing below the baseline. Both LTAG±1 and CCG±1 produce lower TER scores than other context-informed SMT models (i.e. CCG-LTAG±1, Super-Pair±1) and the Moses baseline.

For the English-to-Dutch translation task, we compare the effectiveness of the basic contextual features (Section 4.3.5.3) and the supertag contextual features. In summary, the BLEU and TER metrics indicate supertags (CCG and LTAG) to be the most effective context features, while the METEOR evaluation metric suggests otherwise. As the METEOR metric does not support the Dutch language (Lavie and Agarwal, 2007), we had to use it with its default English settings. We strongly suspect this might be the reason why METEOR shows inconsistencies while evaluating the Dutch sentences.

As an additional point of analysis, we compare translations produced by our best-performing context-informed (CI) system (LTAG±1)[7] with those by the Moses baseline. Table 5.28 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 5.29 compares weights of the various translational features of the LTAG±1 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	250	208	220	322
Sentence-Level TER	325	210	427	38
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	3.8		3.8	
Matching Words (%)	61.50		61.74	

Table 5.28: Comparison between translations produced by the best-performing context-informed (CI) system (LTAG±1) and the Moses baseline.

[7]The best-performing LTAG±1 system which we used for this analysis was built on the largest size of training data (1.31M) (cf. Table 5.27).

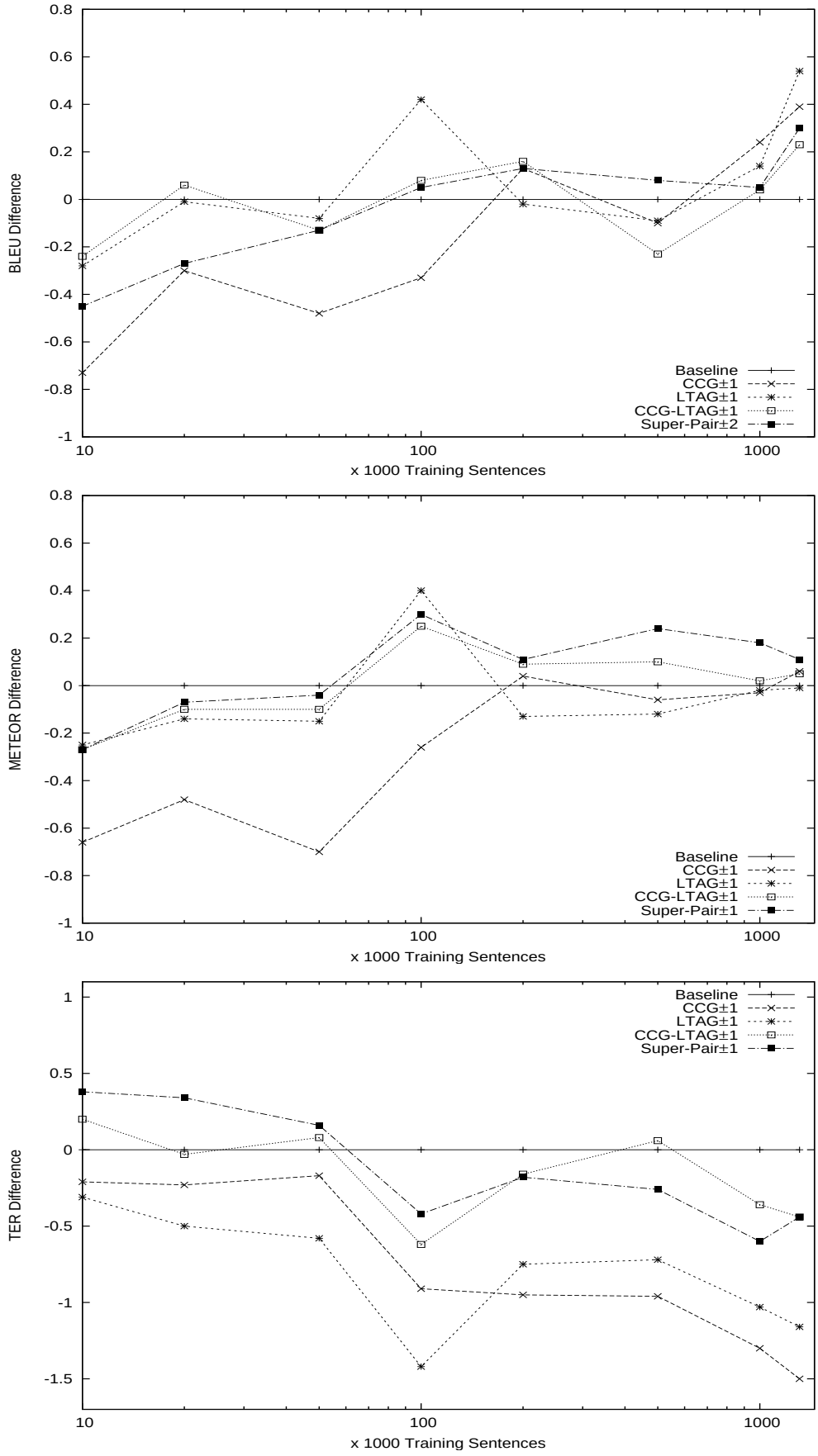

Figure 5.7: BLEU (top), METEOR (centre) and TER (bottom) score-difference curves comparing the Moses baseline against the supertag-based SMT models in English-to-Dutch translation task.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1072	0.0102	0.0509	0.1103	0.0468	0.0936	-0.2689	-	-
LTAG±1	0.1168	0.0081	0.0710	0.0032	0.0510	0.168	-0.2408	0.0697	0.0049

Table 5.29: Comparison of weights for each translational feature of the two systems (LTAG±1 and Moses baseline) obtained by MERT training.

5.3.5 Translation Analysis

We performed manual qualitative analysis comparing the translated output of the best-performing system with that of the Moses baseline system. In order to carry out the manual evaluation, we randomly sampled 50 test set sentences.

We analyzed the translated output of our best-performing system (LTAG±1) against that of the Moses baseline in the English-to-Dutch translation task (cf. Section 5.3.4.2). We observed that the baseline Moses system frequently mistranslates English function words. Moreover, the baseline prefers to generate certain phrases that are collocationally strong n-grams in the Dutch language. We conjecture that during the translation process the Dutch language model might overpower other SMT models for selecting such candidate translations (i.e. those containing collocationally strong n-grams) despite the fact that those are incorrect Dutch equivalents of the English phrases for the particular input sentence. The following two translation examples illustrate how our best-performing system (LTAG±1) surpasses the Moses baseline on this translation task:

(1) English: European agriculture is not uniform .
Reference: De Europese landbouw is verre van eenvormig .
LTAG±1: De Europese landbouw is niet uniform .
Baseline: Europese landbouw niet uniform is .

(2) Baseline: Apart from a limited budget , the European Union has little political interest in Tajikistan .

Reference: Naast een beperkte begroting heeft de Europese Unie politiek gezien weinig in Tadzjikistan te zoeken .

LTAG±1: Afgezien van een beperkte begroting heeft de Europese Unie weinig politieke belang in Tadzjikistan .

Baseline: Afgezien van een beperkte begroting , de Europese Unie heeft weinig politieke belang in Tadzjikistan .

In translation example (1), the translation produced by LTAG±1 is fluent and roughly synonymous to the reference translation, while the baseline generates a translation with the wrong word order, and also misses the initial article 'De'. Translation example (2) resembles (1) in that the LTAG±1 system generates a fluent and grammatical translation except for one agreement issue (the adjective *politieke* should be *politiek* as the noun *belang* has neuter gender), while the baseline system also generates a faulty word order.

5.3.6 Analysis of Learning Curve Experiments

In this section we summarize the outcomes of the learning curve experiments reported in the above sections. When varying the amounts of English-to-Spanish Europarl training data from 10,000 to 1.64 million sentences in a learning curve experiment, the resulting curves demonstrate that our source-language contextual models cannot surpass the baseline PB-SMT model for any amount of training data used. We discover that the TRIBL classifier obtains gains at small training set sizes, though not at the smallest size. In contrast, IGTree requires the maximal amount of training data (1.64 million sentences) to equal the baseline.

Furthermore, learning curve experiments on the English-to-Dutch language direction show that rich and complex syntactic features surpass basic features (words and POS tags) as useful source-language context on small-scale as well as large-scale translation tasks. Moreover, outcomes of the manual analysis conducted on the MT

outputs of the best-performing context-informed model against the respective Moses baseline resemble the findings of several automatic evaluation measures. In general, learning curve experiments give a more accurate overview of relative gains when more data is available.

5.3.7 Context-Dependent vs Context-Independent Phrase Translation

In this section, we compare the effectiveness of the context-dependent phrase translation with that of the context-independent (PB-SMT) phrase translation. First, we give an example to illustrate how a source phrase with additional context information is classified into a distribution over possible target phrases. We consider a particular contextual feature (CCG$\pm$1) to illustrate the classification task. We select a source English sentence ('*let me make a suggestion to the commission as to how this problem could be tackled*') from the development set of the English-to-Spanish translation task (cf. Section 5.3.4.1). This sentence contains the ambiguous single-word phrase '*make*', the contextually appropriate Spanish translation of which should be '*hacer*'.

Following Equation (5.1), we take the CCG supertags of the neighbouring ($\pm$1) words around the source phrase '*make*' in order to form its context information, namely: CI(*make*) = {st(*me*), st(*make*), st(*a*)} = {(S\NP)/NP, NP, NP/N}. Thus, we form a test example $\langle make, \text{CI} \rangle$ which is given to the classifier for classification. As part of the earlier offline training phase, millions of training examples are generated that take the same form as the test examples, labeled with classes (aligned target phrases), with one example for each alignment in the training data. A memory-based classifier is then trained on these millions of training examples.

During decoding, we generate all possible test examples from the test set and give them to the classifier in order to obtain possible translations for the source phrases. For the above test example ($\langle make, \text{CI} \rangle$), we classify it with the rele-

vant classifier, which gives us a class distribution in the form of a list of target phrases which are context-sensitive translations of the source phrase '*make*'. A weight is associated with each class. From these weights we estimate the probabilities of translation into target phrases ($\hat{e}_k$) from the source phrase '*make*' with its additional context information, which are $\mathrm{P}(\hat{e}_k|\mathrm{make}, \mathrm{CI}(\mathrm{make}))$. Table 5.30 shows some of the possible Spanish translations of the English phrase '*make*' including '*hacer*' with their memory-based context-dependent translation probabilities (i.e. memory-based scores: $\mathrm{P}(\hat{e}_k|\hat{f}_k, \mathrm{CI}(\hat{f}_k))$) compared with context-independent translation probabilities (i.e. baseline scores: $\mathrm{P}(\hat{e}_k|\hat{f}_k)$).

	Baseline	CCG±1
	$\mathrm{P}(\hat{e}_k\|\hat{f}_k)$	$\mathrm{P}(\hat{e}_k\|\hat{f}_k, \mathrm{CI}\hat{f}_k)$
hacer	0.2353	0.3412
hagan	0.0851	0.0057
realizar	0.0277	0.0305
hacen	0.0245	0.0073
haga	0.0142	0.0113
...	...	...
TPDS	388	181

Table 5.30: Some of the possible Spanish translations of the English phrase '*make*' with their memory-based context-dependent translation probabilities (rightmost column) compared against context-independent translation probabilities of the baseline system. TPDS: target phrase distribution size.

Table 5.30 shows that the memory-based classifier assigns a relatively higher score to the most suitable Spanish phrase '*hacer*', while it assigns lower scores to three of the four alternative translations listed. The baseline phrase translation probability is estimated using the relative frequency counts of source and target phrases. Additionally we report the target phrase distribution size (TPDS, bottom line in Table 5.30) for the source phrase '*make*' in the baseline system, 388 phrases, as well as in our memory-based model, 181 phrases. This illustrates how the memory-based classifier typically produces a reduced set of target phrases for a given source phrase in context.

As an additional point of analysis, we also compared the log-linear weights (λ_i) of the context-informed memory-based features with those of a baseline features for

the above experiment (CCG±1, an English-to-Spanish translation task described further in Section 5.3.4.1).

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0795	0.0448	0.0091	0.0449	0.0527	0.1151	-0.1451	-	-
CCG±1	0.1075	0.0347	0.0551	0.02639	0.0247	0.0974	-0.1322	0.04633	0.0051

Table 5.31: Weights of different log-linear features of the CCG±1 system with Moses baseline.

Table 5.31 indicates that our context-informed models ($\hat{h}_{\text{mbl}}$, $\hat{h}_{\text{best}}$) contribute positively to the phrase-scoring process during translation. Moreover, MERT (Och, 2003) assigns a notably higher weight to the context-informed feature $\hat{h}_{\text{mbl}}$ than the baseline feature $\hat{h}_{\text{ftp}}$, directly indicating the importance of the memory-based context-informed models.

5.4 Context-Informed Hierarchical PB-SMT

In Section 4.4.1, we reported the experimental results obtained by integrating basic contextual features into the Hiero model. In this section, we demonstrate the experimental results obtained by employing supertags as contextual features into the Hiero model.

Section 5.4.1 reports the outcomes of the English-to-Hindi and English-to-Dutch translations. Section 5.4.2 provides some analysis of the results obtained from the two translation tasks.[8]

5.4.1 Experimental Results

5.4.1.1 English-to-Hindi

We carried out English-to-Hindi translation with the EILMT corpus (cf. Section 3.6). Experimental results for this translation task are displayed in Table 5.32.

[8]The experiments reported in this section have partly been published, albeit in a different form, in Haque et al. (2010a).

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*11.08*	*4.44*	*29.19*	*75.30*	*82.09*	*57.98*
CCG±1	11.40 (75.1%)	4.49	28.05	74.35	80.52	57.98
LTAG±1	11.54 (87.9%)	4.51	29.32	74.49	80.40	57.25
CCG±2	11.63 (98.1%)	4.52	29.00	74.72	81.01	57.68
LTAG±2	11.56 (90.08%)	4.54	29.60	74.31	80.57	57.30

Table 5.32: Results on English-to-Hindi translation obtained integrating supertag contexts into Hiero.

We see from Table 5.32 that adding any type of supertag feature as context improves upon the Hiero baseline regardless of the context width used. CCG±2 produces the best BLEU improvement (0.55 BLEU points; 4.96% relative) over the baseline, which is statistically significant. LTAG±2 gives the second-best BLEU improvement (0.46 BLEU points; 4.16% relative) over the Hiero baseline, which is very close to the significance level. Other evaluation metrics tend to follow a similar trend to the BLEU metric. In summary, supertag-based SMT systems produce slightly better scores when a context width of 2 is considered.

As an additional point of analysis, we compare translations produced by our best-performing context-informed (CI) system (CCG±2) (cf. Table 5.32) with those by the Hiero baseline. Table 5.33 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 5.34 compares weights of the various translational features of the CCG±2 and the baseline systems obtained by MERT training.

	CI>Hiero	CI<Hiero	CI=Hiero	Zero
Sentence-Level BLEU	85	68	21	321
Sentence-Level TER	190	155	150	0
Closeness to Reference Set	CI		Hiero	
Matching Translations (%)	0		0	
Matching Words (%)	51.54		51.10	

Table 5.33: Comparison between translations produced by the best-performing context-informed (CI) system (CCG±2) and the Hiero baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{glue}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Hiero	0.1885	0.0817	-0.0093	0.0914	0.0312	0.1227	-0.0598	-0.4151	-	-
CCG±2	0.1544	0.0104	0.0657	0.0214	0.0367	-0.0832	-0.0519	-0.5522	0.0149	0.0086

Table 5.34: Comparison of weights for each translational feature of the two systems (CCG±2 and Hiero baseline) obtained by MERT training.

5.4.1.2 English-to-Dutch

English-to-Dutch translation was carried out on the Open Subtitles corpus (cf. Section 3.6). Although our main focus was to observe the effect of incorporating supertags as a source contextual feature on translation quality, we also carried out experiments combining supertags with lexical contextual features.

Exp.	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*21.92*	*5.29*	*43.06*	*56.72*	*55.43*	*48.60*
CCG±2	22.65 (90.3%)	5.37	43.83 (99.4%)	56.27	55.08	48.08
LTAG±2	22.55 (91.1%)	5.34	43.99 (99.1%)	56.47	55.22	48.15

Table 5.35: Experimental results of English-to-Dutch translation with individual features, compared against a Hiero baseline.

The results obtained with the individual context features, compared to the baseline, are shown in Table 5.35. Moderate improvements over the Hiero baseline are observed with the addition of CCG supertags (0.73 BLEU points; 3.33% relative increase), and LTAG supertags (0.63 BLEU points; 2.88% relative). Thus, among the individual contextual features, CCG±2 produces the highest BLEU improvements over the baseline. However, none of the improvements are statistically significant (although close) with respect to the baseline.

When focusing on the METEOR evaluation metric, we see that among the individual features, LTAG±2 produces the biggest improvements (0.93 points; 2.16% relative increase) over the baseline. Moderate improvements in METEOR are also observed for the CCG±2 feature (0.77 METEOR points; 1.79% relative increase). In contrast to the BLEU comparisons, all the METEOR improvements with respect to the baseline are statistically significant. Improvements in TER for CCG±2 (a reduction of 0.45 TER points) and LTAG±2 (0.25 TER points) features are quite reasonable and comparable to the improvements in METEOR and BLEU. As far as

other evaluation metrics (NIST, PER and WER) are concerned, improvements (see Table 5.35) measured by them are quite similar to the improvements measured on BLEU, METEOR and TER.

Exp.	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*21.92*	*5.29*	*43.06*	*56.72*	*55.43*	*48.60*
Word±2+CCG±2	22.90 (95.1%)	5.38	44.00 (98.2%)	56.12	54.90	48.24
Word±2+LTAG±2	23.30 (99.5%)	5.37	44.08 (99.6%)	56.37	55.11	47.87
Word±2+CCG±2+LTAG±2	23.00 (99.8%)	5.37	43.89 (98.5%)	55.87	55.27	48.05

Table 5.36: Experimental results with combined features, compared against Hiero baseline.

Subsequently, we performed experiments in which we combined the lexical features with the supertag-based features. The results of these experiments are shown in Table 5.36. Combining LTAG supertags with Word features causes system performance to improve to 23.30 BLEU score, 1.38 points (a relative improvement of 6.3%) over the HPB-SMT baseline. CCG supertags combined with Word features produces an improvement of 0.98 absolute BLEU points (4.48% relative increase). Improvements on both combinations are statistically significant at 99.5% and 95.1% levels of confidence, respectively. Furthermore, we combine lexical features with two types of supertags (Word±2+CCG±2+LTAG±2), which gives a statistically significant 1.08 BLEU points improvement (4.93% relative) over the baseline.

The METEOR evaluation scores show similar trends for the combined set-ups. The best METEOR score (an improvement of 1.02 METEOR points; 2.37% relative) is obtained when words are combined with LTAG supertags. Moderate improvements over the baseline are observed when Word±2+CCG±2 and Word±2+CCG±2+ LTAG±2 are used. The improvements on Word±2+CCG±2, Word±2+LTAG±2 and Word±2+CCG±2+LTAG±2 with respect to the baseline are statistically significant in terms of METEOR.

On the TER evaluation metric, the best-performing combination, Word±2+CCG±2 +LTAG±2, yields an absolute reduction of 0.85 TER points below the Hiero baseline. Reductions of 0.35 and 0.60 TER points below the baseline are seen with the Word±2+CCG±2 and Word±2+LTAG±2 combinations, respectively. As far as

other evaluation metrics (NIST, PER and WER) are concerned, They follow similar trends.

As an additional point of analysis, we compare translations produced by our best-performing context-informed (CI) system (Word±2+LTAG±2) (cf. Table 5.36) with those by the Hiero baseline. Table 5.37 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 5.38 compares weights of the various translational features of the Word±2+LTAG±2 and the baseline systems obtained by MERT training.

	CI>Hiero	CI<Hiero	CI=Hiero	Zero
Sentence-Level BLEU	46	40	129	785
Sentence-Level TER	103	106	679	112
Closeness to Reference Set	CI		Hiero	
Matching Translations (%)	12.4		12.4	
Matching Words (%)	58.18		57.66	

Table 5.37: Comparison between translations produced by the best-performing context-informed (CI) system (Word±2+LTAG±2) and the Hiero baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{glue}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Hiero	0.0823	0.0205	0.0217	0.1702	0.0409	-0.3477	0.3421	0.095	-	-
Word±2+ LTAG±2	0.0801	0.0260	0.005	0.113	0.060	-0.3157	0.3249	0.1306	0.0120	0.0349

Table 5.38: Comparison of weights for each translational feature of the two systems (Word±2+LTAG±2 and Hiero baseline) obtained by MERT training.

5.4.1.3 Translation Analysis

For the English-to-Dutch translation task, we performed a manual qualitative analysis of differences between the translations produced by our best-performing context-informed system (Word±2+LTAG±2) and those by the Hiero baseline. Among the 1,000 test sentences, the Word±2+LTAG±2 system obtains a higher BLEU score than the baseline for 56 sentences, among which in 32 cases the improvement is due to better lexical choice. The Word±2+LTAG±2 system generates a more fluent output in 17 sentences. These two types of improvements overlap on 10 occasions

(i.e. in 10 sentences, the improvement involves both better lexical choice and better fluency). The following are two such translation examples which show how our context-informed system improves over the baseline:

(3) input:	i appreciate your help .
reference:	ik waardeer je hulp .
Word±2+LTAG±2:	ik waardeer je hulp .
baseline:	ik waardeer je helpen .
(4) input:	we' re taking the girl now .
reference:	we halen het meisje nu .
Word±2+LTAG±2:	we nemen het meisje nu .
baseline:	nemen we de meisje nu .

In the example (3), the word 'help' in the source English sentence is ambiguous as it can translate to the noun 'hulp' or the verb 'helpen'. The Word±2+LTAG±2 system conveys a meaning more similar to the input sentence by choosing the correct Dutch word 'hulp'. In the example (4), the translation of the Word±2+LTAG±2 system is more fluent than the baseline Hiero translation, as it generates a correct word order while the baseline does not, and the Word±2+LTAG±2 system chooses the correct neuter article 'het' instead of the incorrect non-neuter article 'de' selected by the baseline.

As an additional analysis, we examined the decoding process to discover why the Word±2+LTAG±2 system generates better output than the baseline. In the example (3), to translate the source sentence, 5,354 candidate phrases are used by the baseline system, while only 460 candidate phrases (IGTree classes) are used by the Word±2+LTAG±2 system (see Table 5.39). As a result, during decoding, 9,654 hypotheses are generated in the Word±2+LTAG±2 system compared to 20,371 hypotheses in the baseline. We also identified details regarding what candidate phrases along with source spans are used for the best translation hypothesis. A source span

for each candidate phrase is represented by word positions in the source sentence ([1..n]; where n: sentence length). In the Word±2+LTAG±2 system, candidate phrases used in the best translation hypothesis are: 'ik':[1..1], 'waardeer':[2..2], 'je hulp':[3..4] and '.':[5..5]. In contrast, the baseline uses two candidate phrases ('ik waardeer je':[1..3] and 'helpen .':[4..5]) to generate the best translation hypothesis, and the usage of the last phrase ('helpen .') in this translation is incorrect.

		Word±2+LTAG±2	Hiero
Example (3)	Candidate Phrases	460	5,354
	Hypotheses	9,654	20,371
Example (4)	Candidate Phrases	1,577	8,518
	Hypotheses	24,092	35,659

Table 5.39: Number of candidate phrases used and hypotheses generated by Word±2+LTAG±2 and Hiero models during decoding.

In the example (4), to translate the source sentence, 8,518 candidate phrases are used by the baseline system, while only 1,577 candidate phrases are used by the Word±2+LTAG±2 system (see Table 5.39). As a result, during decoding, 24,092 hypotheses are generated in the Word±2+LTAG±2 system, as opposed to 35,659 hypotheses in the baseline. In the Word±2+LTAG±2 system, the candidate phrases used to generate the best translation hypothesis are: 'we nemen':[1..2], 'het meisje':[2..4], 'nu':[5..5] and '.':[6..6]. The baseline uses the following candidate phrases: 'nemen we de':[1..3], 'meisje':[4..4], 'nu':[5..5] and '.':[6..6]. The baseline system chooses an incorrect candidate phrase ('nemen we de') to generate the best translation hypothesis.

The above analysis reveals that in addition to the context-dependent translation features, context-informed models use reduced but more fine-grained sets of candidate phrases, which in turn force the model to weed out bad hypotheses during decoding, and thereby improve translation quality.

5.4.1.4 Numbers of Rules and Examples

Hiero usually generates a massive number of rules compared to the phrase-based approach. The first row in Table 5.40 shows that the number of distinct rules (rule

table size) generated by Hiero for the English-to-Dutch data set is almost three times larger than the number of distinct source-target phrase-pairs (phrase table size) generated by Moses on the same data set. The last row in Table 5.40 shows a similar trend in the case of all rules (non-distinct) generated from the parallel training data during the rule extraction process of Hiero. IGTree classifiers are built on the set of examples formed by the source phrase (α), target phrase (γ), and the contextual information (CI) of the source phrase obtained during the rule extraction process in Hiero. In other words, the number of training examples equals the number of times Hiero's rules apply to the training source sentences. Although IGTree scales roughly linearly to larger numbers of examples, it would be a challenge on present-day computers to train IGTree with large-scale training data.

	Hiero	Moses
Distinct	6,761,376	1,988,504
Non-distinct	11,603,617	3,817,252

Table 5.40: Numbers of rules in Hiero or phrase-pairs in Moses.

5.4.2 Discussion

We demonstrated that supertags can also be successfully integrated as source-language contextual features into the state-of-the-art Hiero system (Chiang, 2007).

In the English-to-Hindi translation task, adding supertags as source-language context improves a Hiero baseline despite working with a tiny training set. In the English-to-Dutch translation task, considering only individual contextual features, the system produces moderate gains for supertags (3.33% and 2.88% relative gains in BLEU for CCG±2 and LTAG±2, respectively). Furthermore, we observed the best improvement over the baseline when supertags are combined with word contexts (4.48%, 6.3% and 4.93% relative improvements in BLEU for Word±2+CCG±2, Word±2+LTAG±2 and Word±2+CCG±2+LTAG±2 respectively). If we compare the integration of supertag features as opposed to that of contextual words and POS tags in the Hiero system (cf. Section 4.4.1), the system produces better gains for

supertags than on words and POS tags in the English-to-Dutch translation task.

On the English-to-Hindi translation task, the POS-based system performs slightly better than the supertag-based systems, although the performance differences between them are not statistically significant.

The relative lack of efficiency of the combination of POS tags and supertags lies in the fact that POS information is already present in the supertags. POS tags are therefore redundant when supertags are available. Words, on the other hand, remain relevant as they appear to contain complementary information not carried by supertags.

5.5 Sentence-Similarity Based PB-SMT

In Chapter 4, we demonstrated a series of experimental results obtained by integrating basic contextual features (neighbouring words and POS tags) into the state-of-the-art PB-SMT (Koehn et al., 2003) and Hiero (Chiang, 2007) models. In this chapter, we have illustrated how we integrate lexical syntactic descriptions in the form of supertags into the PB-SMT and Hiero models, and presented a range of experimental results obtained employing supertag contextual features. Among the different types of lexical and syntactic features, lexical syntactic descriptions in the form of supertags (Bangalore and Joshi, 1999; Clark and Curran, 2004) that preserve long-range word-to-word dependencies in a sentence have proven to be more effective than neighbouring words and POS contexts. These rich contextual features are able to disambiguate a source phrase, on the basis of the local syntactic behaviour of that phrase. In addition to local contextual information, global contextual information such as the grammatical structure of a sentence, sentence length and n-gram word sequences could provide additional important information to enhance this phrase-sense disambiguation. In other words, similarity between an input source sentence to be translated with the tranining sentences that were used to create the SMT system can be useful means to weight candidate phrases which are used form the

most likely translation.

In section 5.5.1, we explore various sentence similarity features by measuring the similarity between a source sentence to be translated with the source-side of the bilingual training sentences and integrate them directly into the PB-SMT model. We performed experiments on an English-to-Chinese translation task by applying sentence-similarity features, both individually and collectively with supertag-based features. In Section 5.5.2, we illustrate the results obtained from the experiments we carried out by adding those features.[9]

5.5.1 Sentence-Similarity as Context Feature

Among the source-language contexts, supertags have been shown to perform better than words and POS tags in target phrase selection. Supertags include rich knowledge sources to disambiguate a source phrase, although they provide only local syntactic behaviour of a source phrase. However, global contextual features, such as the similarity measure between an input source sentence to be translated with the source-side of the bilingual training sentences, could sometimes provide useful evidence to choose more appropriate candidate phrases. In addition to local contextual information, global contextual information could be an additional important source of information to enhance the sense disambiguation task.

5.5.1.1 Sentence-Similarity Features

Costa-Jussà and Banchs (2010) integrated source context information into the PB-SMT model by incorporating a feature function estimated using a cosine distance similarity metric. Their feature function is computed for each of the phrase-pairs of the t-table by measuring cosine distance between the input sentence to be translated and the source sentences of the bilingual training corpus from which those phrase-pairs were extracted. A slight improvement was reported over the Moses PB-SMT

[9]The contents in this section have been partly published, albeit in a different form, in Haque et al. (2010b).

baseline system on an English-to-Spanish experimental corpus. Following (Costa-Jussà and Banchs, 2010), we explore various similarity features including cosine distance by measuring the similarity between a source sentence to be translated with the source-side of the parallel training sentences and integrate them directly into the state-of-the-art log-linear PB-SMT (Koehn et al., 2003) model. Furthermore, we conduct experiments by combining sentence-similarity features (global contexts) with supertag-based features (local contexts).

A PB-SMT t-table contains a list of source phrases and their corresponding translations (target phrases) with associated translation probabilities. Source and target phrases are extracted from a large bilingual training corpus to build the t-table. During training, we also keep track of the source sentences of the training corpus from which phrase pairs are extracted.

To translate a source test sentence, first we generate all possible source phrases and gather their corresponding candidate translations (target phrases). We then collect the source training sentences that are linked to the source phrase of each of the phrase-pairs. We measure the similarity between the source test sentence to be translated with the source training sentences. There could be two possible cases: (i) a phrase pair could be extracted from only one training sentence pair, in which case we calculate the similarity score between the source test sentence and only that particular training sentence, and (ii) phrase pairs could be extracted from many training sentence pairs, in which case we calculate the similarity score between the source test sentence and each of the training sentences separately, and then take the average of the scores. The results of sentence-similarity measures are scores which are not probabilities. Finally, we normalize these similarity scores to convert them into probabilities ($\mathrm{P}_{\mathrm{sim}}$). Thus, we derive a log-linear feature $\hat{h}_{\mathrm{sim}}$ to represent global context information as in (5.3):

$$\hat{h}_{\mathrm{SIM}} = \log \mathrm{P}_{\mathrm{SIM}} \tag{5.3}$$

We considered different similarity functions to measure the similarity between two sentences, including: (i) cosine distance as used in (Costa-Jussà and Banchs, 2010), (ii) Dice coefficient (Dice, 1945), and (iii) the METEOR automatic MT evaluation metric (Lavie and Agarwal, 2007). We employed three variations of dice coefficient in our experiments on the basis of the order of n-gram word sequence match between two sentences: monogram overlap Dice coefficient (MODC), bigram overlap Dice coefficient (BODC), trigram overlap Dice coefficient (TODC).

5.5.1.2 Employing Sentence Similarity-Based Feature with Supertag-Based Features

In addition to the sentence-similarity features (cf. Equation 5.3), we also carried out experiments by integrating supertag context into the PB-SMT model (Koehn et al., 2003), both individually and in collaboration with sentence-similarity features. In order to carry out experiments with the supertag features, as mentioned earlier, we derive the context-informed feature $\hat{h}_{\text{mbl}}$ and a binary feature $\hat{h}_{\text{best}}$ (defined in Equation 3.7 and Equation (3.8), respectively). We performed experiments by integrating these log-linear features ($\hat{h}_{\text{sim}}$, $\hat{h}_{\text{mbl}}$, $\hat{h}_{\text{best}}$) directly into the PB-SMT model.

5.5.2 Results and Analysis

Experiments were carried out on an English-to-Chinese task. We used the English-to-Chinese NIST-08 data set (cf. Section 3.6) in order to carry out our experiments. The training set contains 500,000 sentence pairs (henceforth referred to as 'LargeSet') of newswire translation genres from the LDC. We had another training set which is a subset of 'LargeSet' and contains the first 100,000 sentence pairs (henceforth referred as 'SmallSet') of the 'LargeSet'. We used the NIST'05 1,082-sentence test set for tuning and the NIST'08 1,357-sentence 'current' test set for evaluation (cf. Section 3.6).

Our intention to combine sentence similarity-based features with supertag-based features forced us to choose English as the source language, given that supertaggers

are readily available only for English.

5.5.2.1 Automatic Evaluation

The results obtained with the similarity features, compared to the Moses baseline, are shown in Table 5.41. The top rows of Table 5.41 displays the results obtained on 'SmallSet'. In small-scale translation, the cosine distance and METEOR similarity features are unable to show any improvement over the baseline. However, monogram overlap dice coefficient (MODC) and bigram overlap dice coefficient (BODC) similarity features produce 0.17 BLEU points (1.79% relative increase) and 0.18 BLEU points (1.89% relative increase) improvements respectively over the baseline, and these improvements are very close to the significance level; by contrast, the trigram overlap dice coefficient (TODC) similarity feature fails to show any BLEU improvement over the baseline.

	Experiments	BLEU	NIST	TER	WER	PER
SmallSet	Baseline	*8.52*	*4.37*	*80.31*	*84.79*	*64.08*
	Cosine	8.50	4.41	80.06	84.63	63.71
	MODC	8.69 (94%)	4.44	79.62	84.21	63.38
	BODC	8.70 (88%)	4.42	80.09	84.65	63.72
	TODC	8.50	4.43	80.41	85.03	63.82
	METEOR	8.52	4.42	80.40	85.06	63.94
LargeSet	Baseline	*9.85*	*4.87*	*77.13*	*82.03*	*60.29*
	Cosine	10.16 (99.6%)	4.90	77.03	81.97	60.07
	MODC	10.19 (98.9%)	4.91	77.37	82.41	60.44
	BODC	10.06 (92.3%)	4.88	77.29	82.29	60.53
	TODC	10.16 (98.8%)	4.92	76.60	81.70	59.73
	METEOR	10.30 (99.9%)	4.94	76.22	81.06	59.62

Table 5.41: Experimental results applying sentence-similarity features. MODC: Monogram Overlap Dice Coefficient, BODC: Bigram Overlap Dice Coefficient, TODC: Trigram Overlap Dice Coefficient.

Contrary to the small-scale translation, all sentence-similarity based SMT systems in large-scale translation produce moderate improvements in BLEU over the baseline, and most of the improvements are statistically significant with respect to the baseline. The NIST evaluation metric tends to produce similar improvements to those observed on BLEU. As far as edit-distance-based evaluation metrics (TER,

WER, PER) are concerned, moderate improvements are seen for the cosine, TODC, and METEOR similarity features. However, we do not see any improvement across any of the edit-distance-based metrics for the MODC and BODC similarity functions.

	Experiments	BLEU	NIST	TER	WER	PER
SmallSet	Baseline	*8.52*	*4.37*	*80.31*	*84.79*	*64.08*
	CCG±1	8.74 (88%)	4.49	79.87	84.54	63.45
	LTAG±1	8.71 (80%)	4.49	79.94	84.61	63.39
LargeSet	Baseline	*9.85*	*4.87*	*77.13*	*82.03*	*60.29*
	CCG±1	10.22 (99.3%)	4.96	76.68	81.65	59.76
	LTAG±1	10.39 (99.9%)	4.89	76.92	81.82	60.20

Table 5.42: Experimental results applying supertag-based features.

The results obtained with the supertag-based features, compared to the Moses baseline, are shown in Table 5.42. The top three and bottom three rows of Table 5.42 display the results (of Moses baseline and supertag-based models (CCG±1 and LTAG±1)) obtained on 'SmallSet' and 'LargeSet', respectively. Where small-scale translation is concerned, we see moderate improvements in BLEU over the baseline for both CCG (0.22 BLEU points; 2.31% relative) and LTAG (0.19 BLEU points; 2% relative) supertag contexts. Improvements are quite close to the significance level. In large-scale translation, statistically significant improvements in BLEU are to be observed for both the CCG±1 (0.37 BLEU points; 3.75% relative) and the LTAG±1 (0.54 BLEU points; 5.48% relative) features.

Our main intention was to perform experiments by integrating sentence-similarity features and supertag-based features together into the PB-SMT model as different log-linear features to see whether further improvements could be achieved. Hence, the best-performing sentence-similarity set-ups (MODC & BODC for small-scale translation; METEOR for large-scale translation) are combined with CCG and LTAG supertag-based features. Experimental results obtained combining both types of features are shown in Table 5.43.

On small-scale translation, the improvement obtained on MODC and LTAG feature combination is the highest (a 0.50 BLEU point improvement; 5.25% relative)

	Experiments	BLEU	NIST	TER	WER	PER
	Baseline	*8.52*	*4.37*	*80.31*	*84.79*	*64.08*
SmallSet	MODC + CCG±1	8.48	4.34	80.59	85.19	64.32
	MODC + LTAG±1	9.02 (99.9%)	4.49	79.69	84.41	63.31
	BODC + CCG±1	8.71 (88%)	4.44	79.92	84.45	63.67
	BODC + LTAG±1	8.73 (90%)	4.43	80.05	84.69	63.75
	Baseline	*9.85*	*4.87*	*77.13*	*82.03*	*60.29*
LargeSet	METEOR + CCG±1	10.16 (98.1%)	4.92	76.60	81.70	59.73
	METEOR + LTAG±1	10.44 (99.9%)	4.93	76.43	81.41	59.62

Table 5.43: Experimental results applying combined features.

and is statistically significant; in contrast, adding the MODC feature to CCG feature does not produce any improvement over the baseline. Adding the BODC feature to the CCG and LTAG features adds 0.19 BLEU points (2% relative) and 0.21 BLEU points (2.2% relative) respectively to the baseline score.

On the other hand, on large-scale translation, CCG and LTAG supertag features combined with the METEOR similarity feature as source-side contexts produce statistically significant improvements in BLEU (METEOR+CCG±1: 0.31 BLEU points; 3.15% relative; METEOR+LTAG±1: 0.59 BLEU points; 5.98% relative) over the Moses baseline.

Additionally, we carried out an analysis on the translations produced by the best-performing system (METEOR + LTAG±1) and the Moses baseline in the large-scale task. Table 5.44 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 5.45 compares weights of the various translational features of the METEOR + LTAG±1 and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	178	117	95	967
Sentence-Level TER	377	341	637	4
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	0.29		0.37	
Matching Words (%)	52.15		51.98	

Table 5.44: Comparison between translations produced by the best-performing context-informed (CI) system (METEOR + LTAG±1) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}	λ_{sim}
Moses	0.0805	0.0011	0.0760	0.0704	0.0086	-0.0020	-0.1972	-	-	-
METEOR+ LTAG±1	0.0898	0.0355	0.0398	0.0015	0.0359	0.1106	-0.2094	0.0484	0.0007	0.0052

Table 5.45: Comparison of weights for each translational feature of the two systems (METEOR + LTAG±1 and Moses baseline) obtained by MERT training.

5.5.2.2 Translation Analysis

We also performed a sentence-level automatic evaluation of the translations produced by our best-performing context-informed (CI) system (MODC+LTAG±1) on the small-scale translation, compared to those of the Moses baseline. Among the 1,357 test set sentences, the best system obtains a higher BLEU score than the baseline in 177 sentences, while the baseline obtains a higher BLEU score than the best system in 133 sentences. We performed a manual qualitative analysis of the translations of the two systems by randomly sampling a few (25 sentences) of those translations (i.e. 177 sentences). Figure 5.8 shows two such translation examples which illustrate how our context-informed system produces better translations than the baseline.

(5) input: he called for intensive talks in the coming weeks .
reference: 他 呼吁 在 未来 几 周 进行 密集 谈判 。
MODC + LTAG±1: 他 呼吁 **在 未来 数 周** 密集 会谈 。
baseline: 他 呼吁 密集 会谈 **在 未来 数 周** 。

(6) input: i have too many things waiting for me that i cannot do now .
reference: 我 有 太 多 现在 无法 做 的 事情 等待 着 我 去 做 。
MODC + LTAG±1: 我 有 太 多 的 事 等待 我 ， 我 **现在** 不 **做** 。
baseline: 我 有 太 多 的 事 等待 我 ， **我 不 能 在** 。

Figure 5.8: Translation examples comparing the best-performing system (MODC + LTAG±1) and the Moses baseline.

We observe that our best-performing system produces more fluent translations (as in example (5) and (6) in Figure 5.8) than the baseline Moses translations. The improvements of our system over the baseline system are two-fold: better word reordering (as in example (5) in Figure 5.8) and better lexical selection (as in example (6) in Figure 5.8).

We also looked at a few (25 sentences) of those translations (133 sentences) where the baseline system generates better translations than our system. We observe that both the baseline and our best-performing CI system (MODC+LTAG±1) show poor lexical choice and bad word order for most of these translations, due perhaps to the fact that the small training set does not contain the correct translation. Manual evaluation also reveals that MODC+LTAG±1 tends to generate additional words (i.e. preposition, verb) to form more fluent and grammatical translations than the baseline.

As an additional point of analysis, we found that the average number of candidate phrases that are used per source phrase to translate the development set is 97.54 in the baseline PB-SMT model (small-scale). In contrast, the average number of candidate phrases per source phrase that are used to translate the same set are 57.01 and 53.74 in the CCG and LTAG supertag-based models (small-scale) respectively (the combined models also use the same set of target phrases as the supertag-based models). Hence, the supertag-based models use reduced but more fine-grained sets of candidate phrases and employ a memory-based context-dependent weighting in translation. Moreover, integration of sentence-similarity features into the PB-SMT model helps the model to choose more appropriate candidate phrases for a source phrase during translation. Therefore, integration of global (sentence-similarity) and local (supertag) contextual features jointly into the PB-SMT model forces the model to weed out bad hypotheses during decoding, which improves translation quality.

In this translation task, we observed that the baseline Moses systems (with 100K or 500K training sets) produced very low BLEU scores. There might be few reasons for this anomaly: (i) Englisht-to-Chinese is a difficult translation pair since both are morphologically divergent langauges, (ii) we had only one set of reference translations in Chinese, and (iii) unlike IWSLT data, the NEWS data is quite noisy.

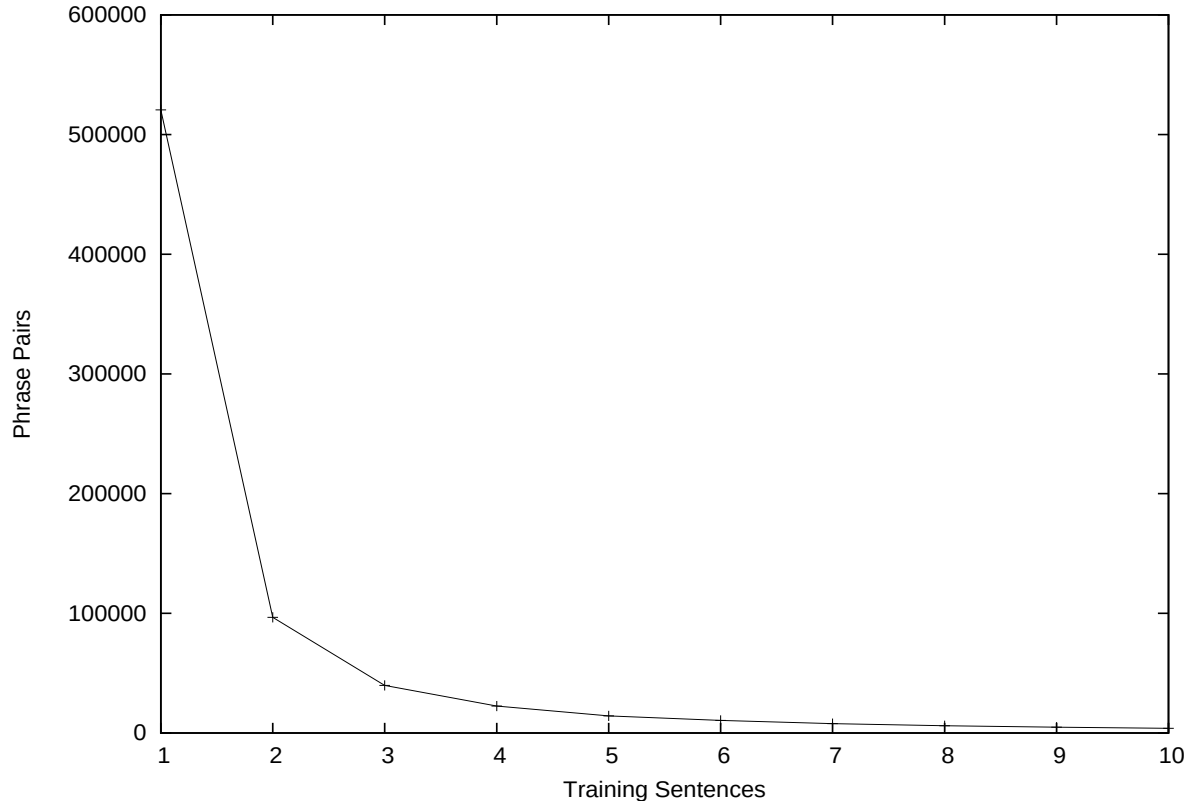

Figure 5.9: Distribution of the source-target phrase-pairs over the number of training sentences from which those phrase-pairs are extracted.

5.5.2.3 Distribution of t-table Entries over Number of Training Sentences

Measuring similarity between a test sentence and a set of source sentences of the training corpus is a time-consuming process. As an additional point of analysis, we measured the distribution of source-target phrase-pairs (t-table entries) over the number of training sentence pairs from which those phrase-pairs are extracted. To perform this analysis, we considered the phrase-table created on the large-scale training data (i.e. 'LargeSet') which contains 500,000 sentence pairs. We extract those phrase-pairs from the original phrase table that are used to translate the development set. Thus, we create a filtered phrase table that contains 773,703 source-target phrase-pairs.[10] We plot the distribution of these source-target phrase-pairs of the filtered phrase table against the number of training sentences from which those phrase-pairs are extracted in Figure 5.9. As can be seen from the graph in

[10]Original phrase table contains 5,190,748 source-target phrase-pairs.

Figure 5.9, 67.92% of the phrase pairs in the phrase-table (filtered) are extracted from only single training sentence pair of the training set. We also observed that only 6.01% and 3.55% phrase pairs of the phrase-table (filtered) are extracted from more than 10 and 20 training sentence pairs, respectively. The above statistics indicate that the similarity measurement process is an issue only for a small number of phrase-table entries.

5.6 Summary

In this chapter, first we provided a short overview of two kinds of supertags: CCG and LTAG. Then, we illustrated how we define contextual information with supertags for a PB-SMT source phrase as well as for a HPB-SMT source phrase. Use of supertag context forced us to use English as source-language of translation pairs since supertag information is readily available only for English.

We demonstrated a series of experimental results obtained by employing various supertag contextual features in the PB-SMT model. We employed three memory-based classification algorithms: IB1, IGTree and TRIBL. However, we used IB1 thereafter only for one small-scale data set (English-to-Chinese IWSLT), and avoided use of IB1 due to its large computing complexity. The experiments were carried out on a series of small- and large-scale data sets, which involves various language pairs. We discovered that supertags appear to be an effective source-language context for both small- and large-scale translations. We also observed that TRIBL seems to be more effective than IGTree since the latter does not contribute much in large-scale translations. We compare the effectiveness of various contextual features, and by this we conclude that supertags appear to be more effective source-side contextual features than words and POS tags.

This chapter also introduced supertags as source-context in the Hiero model, and reported a series of experimental results. We found that adding supertags as well as a combination of supertags and words as source-language contexts improves

upon the Hiero baseline significantly.

Furthermore, we introduced various sentence similarity-based contextual features into PB-SMT, which were deployed individually and collectively with supertag-based features. We found that the combined set-up provided the highest improvements in BLEU over a PB-SMT baseline on both small- and large-scale translation tasks.

In the following chapter, we investigate the incorporation of deep syntactic and semantic contextual information into the two state-of-the-art SMT models (Koehn et al., 2003; Chiang, 2007).

Chapter 6

Deep Syntactic and Semantic Features

In the previous chapter, we have deployed neighbouring words, POS tags and supertags as source language contexts in the state-of-the-art PB-SMT (Koehn et al., 2003) and Hiero (Chiang, 2007) models. Thus far, the context information of a source phrase is modeled as a sequence of features immediately before and after the focus phrase. Although it can be argued that they offer a rich source of information to disambiguate the translation of a source phrase, they remain position-specific and local, and might, therefore, not provide all information needed for disambiguation. In order to compensate for this, we model position-independent contextual features related to the focus phrase. We choose to model the grammatical dependencies linking from and to the head word of a focus phrase with words occurring elsewhere in the sentence. We even go a step further by moving from deep syntactic dependencies to semantic dependencies that a focus phrase has.

The remainder of the Chapter is organized as follows. In Section 6.1, we give an overview of deep syntactic and semantic information which are the focus of this chapter. In Section 6.2, we describe how we make use of the deep syntactic and semantic features in order to form contextual information (CI) of a source phrase. We carry out a range of experiments by incorporating those features into the state-

of-the-art PB-SMT (Koehn et al., 2003) and hierarchical PB-SMT (Chiang, 2007) models. The experimental results obtained adding deep syntactic and semantic contexts into the PB-SMT and the HPB-SMT models are reported in Section 6.3 and Section 6.4, respectively.

6.1 Overview of Deep Syntactic and Semantic Information

6.1.1 Grammatical Dependency Relations

Dependency parsers have become increasingly popular in recent years. They produce grammatical syntactic structures that consist of lexical elements linked by binary asymmetrical relations called *dependencies* (Nivre, 2005). By definition, dependency relations capture long range dependencies among the words in a sentence. Data-driven dependency parsing, a popular approach, relies on a formal dependency grammar and uses corpus data to induce a probabilistic model for disambiguation. The formal dependency grammar has largely developed as a form for syntactic representation used by traditional grammarians (Nivre, 2005).

Figure 6.1 shows the dependency parse tree of the English sentence '*Can you play my favourite old record?*'. It represents a syntactic structure based on the predicate-argument relationships among the words in the sentence. Each relation (i.e. dependency label) expresses a child-parent relationship between a pair of words in the sentence.

MT researchers have made use of dependency parse information at different stages of the SMT model in order to improve overall MT quality (Quirk et al., 2005; Max et al., 2008; Shen et al., 2008). Importantly, Shen et al. (2008) embed a target dependency language model during decoding to exploit long distance word relations, which are unavailable with a traditional n-gram language model. Inspired by these approaches, we tried to explore grammatical dependency information as

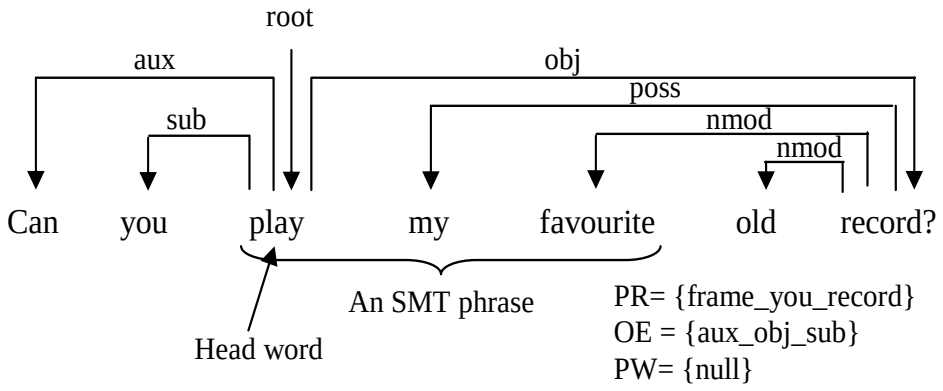

Figure 6.1: The dependency parse tree of the English sentence '*Can you play my favourite old record?*' and the dependency features extracted from it for the SMT phrase '*play my favourite*' (cf. page 163).

source-language context features that could disambiguate a source phrase on the basis of the distant contextual dependencies. In this chapter, we show that position-independent syntactic dependency relations emanating from or pointing to the head of a source phrase can be modeled as a useful source context feature in PB-SMT (Koehn et al., 2003) and HPB-SMT (Chiang, 2007) models to improve target phrase selection.

In our experiments, we used dependency parsers for two different languages: English and Dutch. In order to obtain the dependency parse information for English sentences, we employ the Malt dependency parser[1] (Nivre et al., 2006). To generate dependency features, the Dutch sentences were parsed using Frog,[2] a robust morphosyntactic analyzer and dependency parser (Van den Bosch et al., 2007).

We compare the dependency features with words, part-of-speech tags and supertags as context, in order to observe the relative effects of position-independent and position-dependent features. While supertags represent an abstract view upwards the tree or graph, excluding other lexical nodes and anything below the lowest common ancestors in the tree between other lexical nodes and the words captured in the contextual features, dependency relations directly encode relations between tokens. One can follow a dependency and retrieve the lexical modifier or head at

[1]http://maltparser.org/download.html
[2]http://ilk.uvt.nl/frog/

any distance.

6.1.2 Semantic Role Labeling

Semantic role labeling is an established benchmark task in NLP research since the CoNLL-2004 shared task (Carreras and Márquez, 2004). The task is to identify the semantic arguments associated with a clause's predicate, and their classification into specific semantic roles with respect to the predicate. Typical roles include agent, theme, temporal, locative modifiers. The CoNLL 2008 shared task (Surdeanu et al., 2008) introduced a unified dependency-based formalism that models both syntactic dependencies and semantic roles for English. A dependency-based semantic role labeler (SRL) (Johansson and Nugues, 2008) finds all semantic graphs around each predicate verb or noun in an input sentence in addition to the dependency parse tree. The SRL task begins with the identification of semantic predicates and their arguments in a sentence. Then, SRL identifies all predicate roles, each of which represents a relation between a pair of words in the sentence. For example, Figure 6.2 shows an English sentence ('*Can you play my favourite old record?*') and two semantic graphs around its two predicates (verbal: '*play*' and nominal: '*record*') identified by SRL. In each graph, a set of roles express semantic dependencies of the predicate word with other words (arguments) in the sentence.

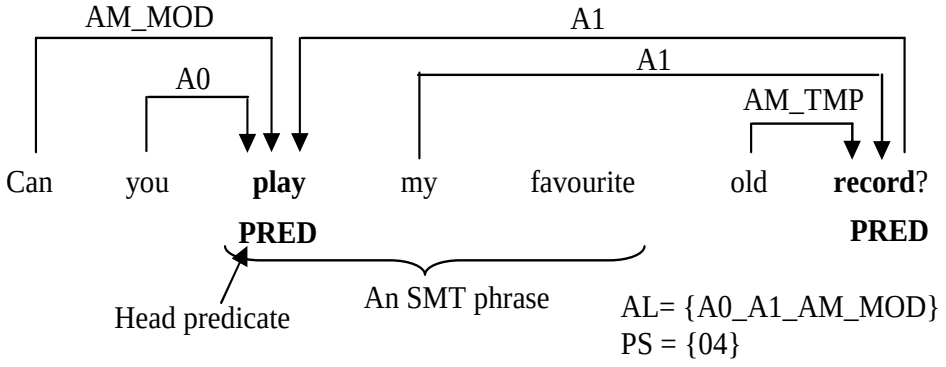

Figure 6.2: The semantic graph of an English sentence and the semantic features extracted from it for an SMT phrase (cf. page 166).

Recently, Wu and Fung (2009) utilized semantic roles in improving SMT accuracy

by enforcing consistency between the semantic predicates and their arguments across both the input sentence and the translation output. Inspired by Wu and Fung (2009), we introduce semantic information as a new contextual feature in PB-SMT for an English-to-Dutch translation task. In order to obtain the semantic graph information of English sentences, we used the LTH semantic parser[3] (Johansson and Nugues, 2008), which assigns both predicative (PropBank-based) and nominative (NomBank-based) graphs.

6.2 Deep Syntactic and Semantic Information as Context

In this section, we describe how we utilize the deep syntactic and semantic parse information in order to derive contextual information (CI) of a source phrase. In Section 6.2.1, we show how CI with the grammatical dependency parse information is derived for a PB-SMT source phrase as well as for a Hiero source phrase. In Section 6.2.2, we illustrate how CI with the semantic information is formed for a PB-SMT source phrase.

6.2.1 Dependency Relations as Context Information

6.2.1.1 Dependency Relations as Context Information for PB-SMT

We model grammatical dependencies linking to and from the head word of a PB-SMT source phrase ($\hat{f}_k$) with words occurring elsewhere in the sentence. The identification of the head word of a phrase is non-trivial, as SMT phrases are not restricted to linguistically coherent phrases, so the identification of a head word cannot be done with linguistic rules of thumb (Magerman, 1995) (e.g. select the head noun from the noun phrase). In our work, we identify head words of the SMT phrases from the dependency tree generated for each sentence. For all words in a given source phrase,

[3]http://nlp.cs.lth.se/software/

the word that hierarchically occupies the highest position in the dependency tree is chosen as the head word. In Figure 6.1 which shows the dependency parse tree of the English sentence '*Can you play my favourite old record?*', we see that the head word of the PB-SMT English phrase '*play my favourite*' is '*play*' according to the tree structure.

We consider the following dependency features, drawing on the syntactic dependencies emanating from or pointing to the head word of the source focus phrase (see also Figure 6.1):

OE (outgoing edges) — For the head word of the focus phrase, we extract a list of zero or more relations with other words of which the word is the parent (i.e. the dependency type labels on all modifying dependency relations). The list of relations is concatenated and sorted uniquely and alphabetically into a single feature. This feature is denoted as OE, for 'outgoing edges'. For example, the head word (*'play'*) of the focus phrase (*'play my favourite'*) has three outgoing edges: auxiliary, subject and object (see Figure 6.1). Therefore, the OE feature is formed as: OE = {aux_obj_sub}.

PR (parent relation) — For the head word of the focus phrase we extract the relation it has with its parent. If the head word is a verb, then the subcategorization frame information is extracted and used as this feature. This feature is denoted as PR, for 'parent relation'. For example, the head word *'play'* is a verb, and so we extract its subcategorization frame information from the tree, namely {frame_you_record}.

PW (parent word) — Extending the PR feature, we encode the identity of the parent word of the head word of the focus phrase. This feature is denoted as PW, for 'parent word'. For example, the head word *'play'* is root according to the tree structure, and has no parent word. So we set the PW feature to *null*.

Together we refer to these dependency features as the grammatical dependency information ($\mathrm{CI}_{\mathrm{di}}(\hat{f}_k)$) of the focus phrase ($\hat{f}_k$). These dependency features can be

applied both individually and jointly. For instance, a combination of three dependency features (OE, PR and PW) defines the contextual information $\mathrm{CI}_{\mathrm{di}}(\hat{f}_k)$ as in (6.1):

$$\mathrm{CI}_{\mathrm{di}}(\hat{f}_k) = \{\mathrm{PR}, \mathrm{OE}, \mathrm{PW}\} \tag{6.1}$$

We derive the context-sensitive log-linear feature $\hat{h}_{\mathrm{mbl}}$ (defined in Equation (3.7)) with the dependency contextual information (i.e. $\mathrm{CI}_{\mathrm{di}}$). In order to carry out experiments with the dependency features, we incorporate the context-sensitive feature $\hat{h}_{\mathrm{mbl}}$ and a binary feature $\hat{h}_{\mathrm{best}}$ (defined in Equation (3.8)) in the log-linear framework of Moses. The experimental results obtained employing the dependency features are reported in Section 6.3.

Two published studies are closely related to our work on integrating dependency features. Carpuat and Wu (2007) mention in passing that their WSD system uses basic dependency relations, but the nature of this information is not further described, nor is its effect. Max et al. (2008) exploit grammatical dependency information, in addition to information extracted from the immediate context of a source phrase. Our approach differs with Max et al. (2008) at least in three respects:

1. Max et al. (2008) select a set of the 16 most informative dependency relations for their experiments. Dependencies that link any of the tokens in the given source phrase to tokens outside the phrase are considered. Each dependency type is represented in the vector by the outside word it involves, or by the symbol 'nil', which indicates that this type of dependency does not occur in the phrase under consideration. In contrast to this approach, we used all (26) dependency relations in our experiments, while only extracting features from the head words of the SMT phrases.

2. They filter out phrases from the phrase table for which $\mathrm{P}(\hat{e}_k|\hat{f}_k) < 0.0002$. In contrast, we keep all phrase pairs.[4]

[4]Filtering out translations of a source phrase is always a risk since any of the discarded target

3. Their experimental data contains 95K English-to-French training sentence pairs, while we carried out a range of experiments considering different data sizes, domains, and language pairs, elaborated further in Section 6.3.

6.2.1.2 Dependency Relations as Context Information for HPB-SMT

Like the head-word identification of the source phrases in PB-SMT, we identify the head word of a Hiero source phrase (α) with the use of a dependency tree generated for the sentence from which the Hiero phrase is extracted. A Hiero phrase may contain a combination of terminal and nonterminals in the source language. Accordingly, the head word of a source phrase (α) may appear in the nonterminal.

We derive two dependency features – OE and PR – for a Hiero phrase following the similar approach to the one we adopted to derive those features for a PB-SMT source phrase (cf. Section 6.2.1.1). We refer to these dependency features as the grammatical dependency information $\text{CI}_{\text{di}}(\alpha)$ of the source phrase (α). As in PB-SMT, these dependency features can be applied either individually or jointly. For instance, a combination of the two dependency features (OE, PR) defines the contextual information $\text{CI}_{\text{di}}(\alpha)$ as in (6.2):

$$\text{CI}_{\text{di}}(\alpha) = \{\text{PR}, \text{OE}\} \tag{6.2}$$

In order to derive the context-sensitive log-linear feature ϕ_{mbl} (defined in Equation (3.9)), we use the dependency feature-based context information (i.e. $\text{CI}_{\text{di}}(\alpha)$) for each source phrase (α). In order to carry out experiments with the dependency contextual features, we incorporate the context-sensitive feature ϕ_{mbl} and a binary feature ϕ_{best} (defined in Equation (3.10)) in the log-linear framework of Hiero. The experimental results obtained integrating dependency features into the Hiero model are reported in Section 6.4.

phrases could be the most acceptable translation of that source phrase under a particular contextual environment.

6.2.2 Semantic Roles as Context Information

We introduce semantic information as a source-side contextual feature in PB-SMT for an English-to-Dutch translation task. The semantic parsing is computationally very expensive, so we carried out experiments with semantic contextual features only on the English-to-Dutch Europarl data set (cf. Section 3.6). Note that we did not explore semantic context in the state-of-the-art Hiero system (Chiang, 2007).

The semantic information (CI_{si}) of a PB-SMT source phrase ($\hat{f}_k$) originates from the semantic (verbal or nominal) predicate captured in that phrase. This introduces three possible cases: (a) there is no predicate in the source phrase, (b) there is only one predicate in the source phrase, in which case this is chosen as the head predicate to define (CI_{si}), and (c) more than one predicate occurs in the source phrase; for such cases, the predicate that occupies hierarchically a superior position in the dependency parse tree is chosen as the head predicate. Figure 6.2 shows an English phrase '*play my favourite*' identified by our baseline PB-SMT system, Moses, which contains only the verbal predicate '*play*'.

In our experiments, two semantic features were considered for the head predicate:

AL (argument labels) — We extract the list of one or more predicate roles (argument labels) of the head predicate of a source phrase. The roles are concatenated and sorted uniquely and alphabetically into a single feature. This feature is denoted as AL, for 'argument labels'. For example, Figure 6.2 illustrates that the head predicate ('*play*') of the focus phrase ('*play my favourite*') has three semantic dependencies (argument labels): acceptor (A0), thing accepted (A1) and modal (AM_MOD).

PS (predicate sense) — In addition to the semantic roles of a predicate, SRL attempts to disambiguate the sense of the predicate in the source sentence. We extract the sense of the head predicate of the source phrase. This feature is denoted as PS, for 'predicate sense'. For example, the sense of the head

predicate (*'play'*) in the sentence in Figure 6.2 is {04}.[5]

The two features AL and PS are applied both individually and jointly. For instance, a combination of the two semantic features defines the contextual information $\mathrm{CI}_{\mathrm{si}}(\hat{f}_k)$ of the source phrase ($\hat{f}_k$) as in (6.3):

$$\mathrm{CI}_{\mathrm{si}}(\hat{f}_k) = \{\mathrm{AL}, \mathrm{PS}\} \tag{6.3}$$

In order to carry out experiments by integrating semantic contextual features into the PB-SMT model (Koehn et al., 2003), we followed the approach that we adopted in order to integrate dependency contextual features into the PB-SMT model (illustrated in the previous section). The experimental results obtained employing semantic features are reported in Section 6.3.

6.3 Experiments with Context-Informed PB-SMT

In this section, we report the experimental results obtained by integrating deep syntactic and semantic contextual features into the PB-SMT model. This section is divided into six subsections. Section 6.3.1 reports experimental results on small-scale data sets representing the language pairs Dutch-to-English, English-to-Hindi, and English-to-Czech. Section 6.3.2 reports experimental results on large-scale data sets representing the language pairs Dutch-to-English and English-to-Dutch. Section 6.3.3 provides some analysis of the effectiveness of various contextual features in small- and large-scale translations. In Section 6.3.4, we present the experimental results of the learning curve experiments which we carried out on three different language pairs: English-to-Spanish, Dutch-to-English and English-to-Dutch. In Section 6.3.5, we provide some analysis of the outcomes of the learning curve experiments.[6]

[5]Figure 3.1 (cf. page 40) shows four English sentences, each containing the word '*play*'. The fourth sentence in Figure 3.1 is used in Figure 6.2 to illustrate the semantic features.

[6]Parts of the experiments carried out with our context-informed PB-SMT model including learning curve experiments have been reported, albeit in a different form, in Haque et al. (2011).

6.3.1 Experiments on Small-Scale Data Sets

6.3.1.1 Dutch-to-English

Small-scale Dutch-to-English translation was performed on the Dutch-to-English Open Subtitles corpus (cf. Section 3.6).[7] Five experiments were performed combining dependency features (outgoing edges: OE, parent relation: PR, and parent word: PW), the results of which are shown in Table 6.1. The combination of PR and OE produces the best results in terms of BLEU and NIST: we observe a 0.67 absolute improvement corresponding to 2.07% relative improvement in terms of BLEU, which is not statistically significant.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*32.39*	*6.11*	*55.39*	*50.15*	*49.67*	*43.12*
PR	32.69	6.08	55.08	50.48	50.11	43.58
OE	32.61	6.00	55.53	52.40	51.56	45.09
PR+PW	32.74	6.06	55.98	51.15	50.75	43.61
PR+OE	33.06 (60%)	6.20	55.70	49.45	48.83	42.44
PR+OE+PW	32.79	6.18	55.37	49.51	49.03	42.43

Table 6.1: Experiments with dependency relations.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*32.39*	*6.11*	*55.39*	*50.15*	*49.67*	*43.12*
PR+OE+Word±2	33.05	6.11	56.02	50.62	49.82	43.68
PR+OE+POS±2	33.30	6.09	56.57	50.52	50.17	43.81
PR+OE+POS±2†	33.39 (98.7%)	6.11	56.30	50.43	50.34	43.54

Table 6.2: Experiments combining dependency relations, words and part-of-speech.

In the second series of experiments, we combined the position-independent PR+OE dependency feature with the position-dependent word and part-of-speech features. The combined experimental results are reported in Table 6.2. We observe that combining POS±2† with PR+OE yields the highest BLEU improvement (1.0 BLEU point; 3.08% relative) over the baseline, which is statistically significant at the 98.7% level of confidence. The best METEOR score (an improvement of 1.18 METEOR

[7]Experiments reported in this section have been summarized, albeit in different form, in Haque et al. (2009b).

points over the baseline; 2.14% relative) is obtained when PR+OE is combined with POS±2.

In this translation task, we compare the effectiveness of the dependency relations with that of the basic features (experimental results with basic contextual features were reported in Section 4.3.1.2) as source-language context in PB-SMT. In sum, the small-scale Dutch-to-English translation task shows the POS contextual feature to produce the largest single-feature improvement over the baseline, while the difference in score between POS-based and dependency-based contextual models is negligible. Moreover, the highest improvement (statistically significant) in BLEU over the baseline is obtained employing the combination of the POS- and dependency-based features.

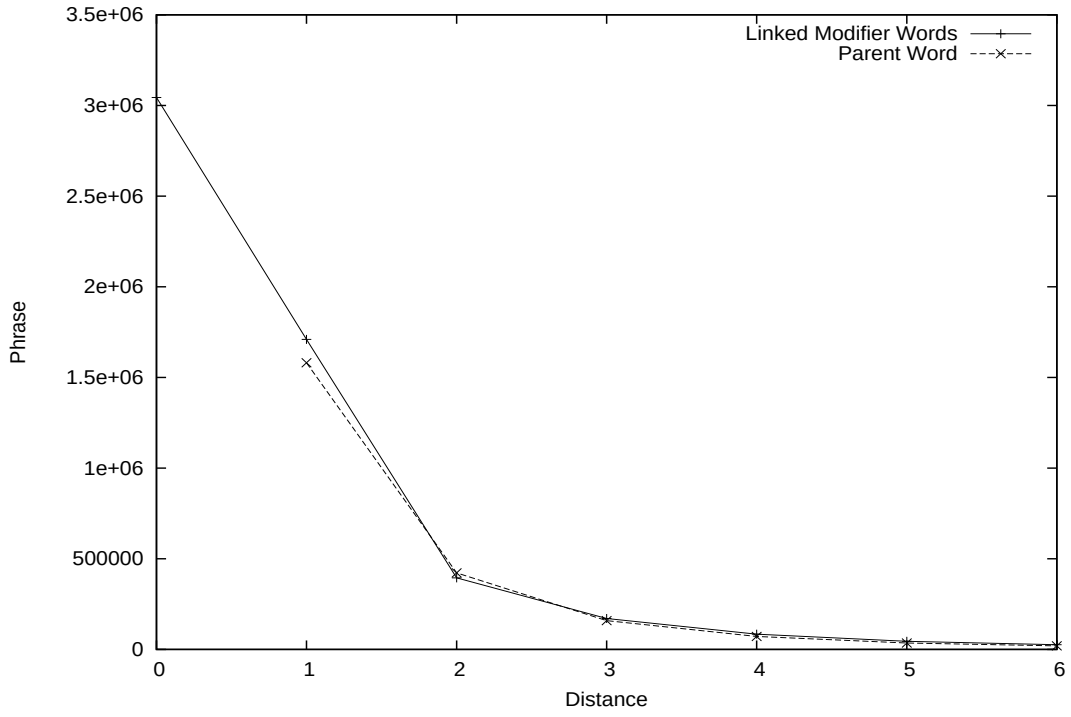

Figure 6.3: Distances found between phrase boundaries with linked modifier words and parent word.

As an additional analysis, Figure 6.3 displays the distribution of distances (number of tokens) between the source phrase boundary and the words outside the phrase linked via a dependency relation. There are about twice as many outgoing edge dependency relations linking to modifier words outside the focus phrase than to

phrase-internal modifiers. About half the phrases have the root of the dependency graph as the parent, i.e. they are the main verbs. For the remaining phrases, the parent of the head-word is a phrase-external word. From the distance distribution statistics we find that the average distance of head-modifying words to the phrase boundary is only 0.75 tokens when including phrase-internal relations, indicating that modifiers of the phrases are usually not too far away, and are mostly immediate neighbours. In contrast, parent words of the phrases' head words are found relatively further away, at an average distance of 1.69 tokens outside the phrase boundary.

In this translation task, we carried out an analysis on the translations produced by our best-performing context-informed (CI) system (PR+OE+POS±2†) and the Moses baseline. Table 6.3 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 6.4 compares weights of the various translational features of the PR+OE+POS±2† and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	62	54	246	638
Sentence-Level TER	107	117	613	163
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	17.5		17.2	
Matching Words (%)	62.42		61.72	

Table 6.3: Comparison between translations produced by the best-performing context-informed (CI) system (PR+OE+POS±2†) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1065	0.0583	0.0078	0.0579	0.0650	0.0998	-0.2357	-	-
PR+OE+POS±2†	0.0907	0.0086	0.0047	0.0211	0.0253	0.1798	-0.2184	0.0692	-0.0018

Table 6.4: Comparison of weights for each translational feature of the two systems (PR+OE+POS±2† and Moses baseline) obtained by MERT training.

6.3.1.2 English-to-Hindi

For the English-to-Hindi translation task, we use the previously best-performing set-up (PR+OE) obtained from the Dutch-to-English translation task. We employ dependency contextual features jointly with lexical, POS, and supertag contextual features. Experiments were carried out using the small EILMT tourism corpus (cf. Section 3.6). The experimental results for individual and joint features, using the TRIBL classifier, are displayed in Table 6.5.

Experimental results in Table 6.5 show that dependency features (i.e. PR+OE) produce an improvement of just 0.09 BLEU points (0.82% relative increase) over the baseline, which is not statistically significant. The other evaluation metrics show similar levels of improvement. The results of combining the dependency features with various contextual features are also shown in Table 6.5. Combining PR+OE with POS±2 and the concatenation of CCG and LTAG supertag features, referred to as PR+OE+POS±2+CCG+LTAG±1† (last row in Table 6.5), we achieve the overall best improvement (0.41 BLEU points; 3.7% relative) over the baseline. Again, the improvement is not statistically significant, yet is close to the significance level. Similar trends are observed on other evaluation metrics for the combined features.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*10.93*	*4.54*	*28.59*	*74.87*	*82.06*	*56.67*
PR+OE	11.02	4.58	28.28	74.65	81.75	56.55
PR+OE+POS±2	11.08	4.56	28.59	75.39	82.67	56.65
PR+OE+Word±2	11.02	4.55	28.59	75.13	82.27	56.85
PR+OE+CCG±1	11.02	4.57	28.27	74.77	82.08	56.56
PR+OE+LTAG±1	11.08	4.56	28.27	75.39	82.67	56.65
PR+OE+CCG±1+LTAG±1†	11.02	4.57	28.27	74.77	82.08	56.56
PR+OE+Super-pair±1†	11.02	4.54	28.27	75.00	82.00	56.82
PR+OE+Super-pair±2†	11.23	4.57	28.59	74.74	81.92	56.55
PR+OE+POS±2+CCG+LTAG±1†	11.34 (89%)	4.58	27.94	74.70	82.00	56.44

Table 6.5: Experiments applying dependency features features in English-to-Hindi translation.

In this translation task, we compare the effectiveness of various contextual features (experimental results with basic contextual features and supertag contextual features were reported in Sections 4.3.1.3 (on page 67) and 5.3.1.2 (on page 112), respectively) as a source-language context in PB-SMT. To summarize, the word con-

textual model produces the biggest single-feature improvement over the baseline in terms of BLEU. The combination of dependency, supertag, and POS features brings about the highest BLEU score in this translation task, although the improvements are not statistically significant.

We also carried out an analysis on the translations produced by our best-performing context-informed (CI) system (PR+OE+POS±2+CCG+LTAG±1†) and the Moses baseline. Table 6.6 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 6.7 compares weights of the various translational features of the PR+OE+POS±2+CCG+LTAG±1† and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	72	54	29	358
Sentence-Level TER	167	160	168	0
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	0		0	
Matching Words (%)	51.78		51.86	

Table 6.6: Comparison between translations produced by the best-performing context-informed (CI) system (PR+OE+POS±2+CCG+LTAG±1†) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0980	0.0453	0.0652	0.0510	0.1948	0.0910	-0.1683	-	-
PR+OE+POS±2+ CCG+LTAG±1†	0.0719	0.0268	0.0313	0.0237	0.0589	0.0021	-0.2066	0.0447	-0.006

Table 6.7: Comparison of weights for each translational feature of the two systems (PR+OE+POS±2+CCG+LTAG±1† and Moses baseline) obtained by MERT training.

6.3.1.3 English-to-Czech

For the English-to-Czech translation task (Penkale et al., 2010)[8] we employed the previously best performing experimental set-up. The evaluation results on the WMT 2009 test set are reported in Table 6.8. We observe that small improvements over

[8]Parts of experimental results in the English-to-Czech translation task have been reported in Penkale et al. (2010).

the Moses baseline are achieved for PR and PR+OE features in terms of BLEU. Moderate improvements are observed for PR and PR+OE in METEOR and TER. The highest METEOR score over the baseline is obtained for the dependency parent relation (PR: 0.31 METEOR points improvement; 0.91% relative). On TER, PR yields an absolute reduction of 0.13 TER points below the baseline.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*7.83*	*3.90*	*34.13*	*87.66*	*80.53*	*67.88*
PR	7.85	3.92	34.44	87.53	80.57	67.88
PR+OE	7.86	3.92	34.29	87.55	80.53	67.80

Table 6.8: Experimental results on the WMT 2009 test set.

Experimental results on the WMT 2010 test set are shown in Table 6.9. We observe that the improvements with this test set are similar to the improvements obtained with the WMT 2009 test set. PR yields a slight improvement in BLEU over the Moses baseline. As far as METEOR is concerned, the PR feature produces the best improvement (a gain of 0.28 METEOR points, 0.81% relative) over the baseline. However, PR+OE produces moderate improvement in METEOR (a gain of 0.20 METEOR points; 0.57% relative) above the baseline.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*8.05*	*3.97*	*34.61*	*86.01*	*78.54*	*67.48*
PR	8.06	4.00	34.89	85.98	78.62	67.43
PR+OE	8.03	3.99	34.81	85.97	78.63	67.44

Table 6.9: Experimental results on the WMT 2010 test set.

To summarize the effectiveness of the different contextual features (experimental results with basic contextual features and supertag contextual features were reported in Sections 4.3.1.4 (on page 68) and 5.3.1.3 (on page 114), respectively) as a source-language context in PB-SMT, for this translation task slight improvements over the baseline are seen for supertag and dependency features, while POS and word contexts do not improve the baseline at all. None of the improvements over the baseline models are statistically significant in terms of BLEU.

We also carried out an analysis on the translations produced by the PR sys-

tem and the Moses baseline (with WMT 2010 test set). Table 6.10 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 6.11 compares weights of the various translational features of the PR and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	205	190	106	1988
Sentence-Level TER	584	591	1308	6
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	0.241		0.281	
Matching Words (%)	30.53		30.73	

Table 6.10: Comparison between translations produced by the best-performing context-informed (CI) system (PR) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1017	0.0405	0.0265	0.0550	0.0222	0.2377	-0.1147	-	-
PR	0.1108	0.0080	0.0905	-0.0965	0.0234	0.0134	-0.1165	0.1273	0.0022

Table 6.11: Comparison of weights for each translational feature of the two systems (PR and Moses baseline) obtained by MERT training.

6.3.2 Experiments on Large-Scale Data Sets

6.3.2.1 Dutch-to-English

We carried out a similar series of experiments with large-scale data sets to the ones reported in the previous section. Our first experimental data set is the Dutch-to-English Europarl corpus (cf. Section 3.6). Like the classification approaches employed in the previous large-scale translation tasks, we used the IGTree classifier to carry out this set of experiments, as TRIBL's memory needs become too demanding with data sets of this size.

We used similar experimental settings to that used with the small-scale Open Subtitles data set reported in Section 6.3.1.1. Experimental results are reported in Table 6.12, where we see that some of the dependency features produce small improvements over the baseline. Of these, PR+OE produces the largest improvement

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*27.29*	*6.686*	*56.81*	*58.65*	*63.97*	*45.18*
Dependency Relations						
PR	27.47 (66%)	6.726	57.02	58.5	63.59	45.10
OE	27.40	6.737	56.95	58.3	63.56	44.92
PR+OE	27.53 (63%)	6.721	57.15	58.64	63.93	45.08
PR+PW	27.17	6.690	56.86	58.94	64.09	45.34
PR+OE+PW	27.29	6.725	56.89	58.68	63.82	45.18
Combinations of Words, Part-of-Speech tags and Dependency Relations						
PR+OE+Word±2	27.02	6.69	56.7	59.00	64.23	45.44
PR+OE+POS±2	27.14	6.64	56.68	59.39	64.56	45.76
PR+OE+POS±2†	27.16	6.66	56.58	59.00	64.17	45.53

Table 6.12: Results on large-scale Dutch-to-English translation.

(0.24 BLEU points; 0.88% relative) over the baseline, but, none of the improvements are statistically significant. Furthermore, we combine the best-performing dependency feature combination (PR+OE) with Word±2, POS±2 and POS±2†, the results of which are shown in the last rows of Table 6.12. Nevertheless, none of the combinations are able to produce any improvement over the baseline. The other evaluation metrics tend to follow the same trends as with BLEU.

We summarize the Dutch-to-English translation task by comparing the effectiveness of the basic contextual features (experimental results with basic contextual features were reported in Section 4.3.2.1 (on page 70)) with that of the dependency contextual features. In sum, word- and POS-based models do not show any improvements over the baseline PB-SMT model. In contrast, we achieve small but consistent improvements over the baseline across all evaluation metrics when dependency relations are employed as source-language contextual features.

Additionally, we carried out an analysis on the translations produced by the PR and the Moses baseline. Table 6.13 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Table 6.14 compares weights of the various translational features of the PR and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	258	224	240	278
Sentence-Level TER	249	249	261	41
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	4.3		4.2	
Matching Words (%)	65.43		65.23	

Table 6.13: Comparison between translations produced by the best-performing context-informed (CI) system (PR) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1086	0.0641	0.0107	0.0569	0.0867	0.0977	-0.2459	-	-
PR	0.1068	0.0259	0.0731	0.0642	0.0545	0.0503	-0.2100	0.0343	0.0013

Table 6.14: Comparison of weights for each translational feature of the two systems (PR and Moses baseline) obtained by MERT training.

6.3.2.2 English-to-Dutch

We conducted experiments by incorporating dependency features on the same Dutch-to-English Europarl data set described in the above section (Section 6.3.2.1), but in the reverse direction. We introduce semantic roles as a new contextual feature for this translation task, and in addition we tried different combinations of lexical, syntactic and semantic features.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*24.26*	*6.177*	*52.68*	*64.37*	*68.81*	*50.02*
Dependency Relations						
PR	24.72 (99.9%)	6.245	52.75	63.87	68.36	49.76
OE	24.32	6.219	52.62	64.00	68.52	49.87
PR+OE	24.62	6.235	52.82	63.95	68.26	49.81
PR+PW	24.58	6.260	52.80	63.62	68.33	49.49
PR+PW+OE	24.26	6.204	52.46	64.24	68.70	50.03
Semantic Roles						
AL	24.56 (90.9%)	6.237	52.66	63.95	68.09	49.54
AL+PS	24.50 (91.6%)	6.221	52.50	64.15	68.33	49.79

Table 6.15: Results on English-to-Dutch Translation employing deep syntactic and semantic features.

Experimental results for individual contextual features are displayed in Table 6.15. Among the dependency features, PR produces the highest improvement (0.46 BLEU points; 1.90% relative) over the baseline, which is statistically significant at a 99.9% level of confidence. Among the semantic features, AL yields the highest

score (a 0.30 BLEU point improvement; 1.24% relative) over the baseline, but this is not statistically significant, although it is close to the significance level. Overall, PR remains the best performing feature among the individual context features.

Similar to the previous approaches, the best-performing settings were combined to see whether further improvements could be achieved. Experimental results for the combined features are reported in Table 6.16. We see from Table 6.16 that a combined set-up (AL+PS+PR) equals the BLEU score obtained with the PR feature (cf. Table 6.15), and this improvement is statistically significant at the 98.2% level of confidence. Among the other combinations, Word±2 combined with PR produces the second highest improvement (0.40 BLEU points, 1.64% relative increase) over the baseline, although this is not a statistically significant increase.

Experiments	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*24.26*	*6.177*	*52.68*	*64.37*	*68.81*	*50.02*
PR+Word±2	24.66 (92.9%)	6.302	52.89	63.36	67.95	49.09
PR+CCG-LTAG±1	24.51	6.301	52.55	63.14	67.32	49.04
PR+CCG-LTAG±1†	24.55	6.232	52.58	63.72	68.01	49.48
AL+PR	24.70 (92.9%)	6.258	52.79	63.70	68.10	49.55
AL+CCG-LTAG±1	24.55	6.236	52.59	63.99	68.26	49.81
AL+PS+PR	24.72 (98.7%)	6.254	52.64	63.89	68.14	49.53
AL+PS+CCG-LTAG±1	24.50	6.218	52.77	64.23	68.35	49.73

Table 6.16: Results on English-to-Dutch translation combining best performing features.

Similar to the previous translation tasks reported in the above sections, for this translation task we compare the effectiveness of the various contextual features in the PB-SMT model (experimental results with basic contextual features and supertag contextual features were reported in Sections 4.3.2.2 (on page 71) and Section 5.3.2.1 (on page 116), respectively). We see that improvements over the baseline for the dependency and supertag-based context-informed models are statistically significant in terms of BLEU at 99.9% and 96% levels of confidence respectively. In contrast, improvements for the word context are not statistically significant, and the POS-based model performs below the baseline PB-SMT model. The semantic role contextual feature achieved modest gains over the baseline, both when used individually and

in collaboration with other features. While this is encouraging, as noted earlier, semantic parsing is computationally expensive, so any gains in translation accuracy need to be offset against slower processing speeds.

In this translation task, we carried out an additional analysis on the translations produced by our best-performing context-informed (CI) system (PR) and the Moses baseline. Table 6.17 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Table 6.18 compares weights of the various translational features of the PR and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	226	189	256	329
Sentence-Level TER	274	219	469	38
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	3.8		3.8	
Matching Words (%)	61.72		61.74	

Table 6.17: Comparison between translations produced by the context-informed (CI) system (PR) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1072	0.0102	0.0509	0.1103	0.0468	0.0936	-0.2689	-	-
PR	0.0916	0.0059	0.0440	0.0324	0.0357	0.049	-0.2123	0.0808	0.0203

Table 6.18: Comparison of weights for each translational feature of the two systems (PR and Moses baseline) obtained by MERT training.

6.3.3 Effect of Different Contextual Features

In this section, we report a comparison of the different contextual features in small- and large-scale translations. We compare the effectiveness of deep syntactic and semantic contextual features with that of the basic and supertag contextual features reported in previous chapters.

The small-scale Dutch-to-English translation task showed that the POS contextual feature produces the biggest improvement over the baseline, but the difference in score between POS-based and dependency-based contextual models is negligible. Moreover, in this translation task, the highest improvement over the baseline is

obtained by employing a combination of the POS- and dependency-based features, which is statistically significant in terms of BLEU.

For English-to-Hindi, the word contextual model produces the biggest improvement over the baseline in terms of BLEU when we look at individual features. However, the combination of dependency, supertag, and POS features brings about the highest BLEU score in this translation task, although the results are not statistically significant.

For English-to-Czech, slight improvements over the baseline are seen for supertags and dependency features, but POS and word contexts do not contribute at all. None of the improvements over the baseline models in both English-to-Hindi and English-to-Czech translation tasks are statistically significant in terms of BLEU.

If we roughly compare the effectiveness of the various contextual features both collectively and individually, for small-scale translation tasks we see that supertags and dependency relations seem to be more effective contexts than neighbouring words and part-of-speech contexts.

For large-scale Dutch-to-English translation, word- and POS-based models do not show any improvements over the baseline PB-SMT model. In contrast, we achieve small but consistent improvements over the baseline for all evaluation metrics when dependency relations are employed as the source-language contextual features.

For the reverse language direction, the dependency and supertag-based context-informed models produced statistically significant improvements over a PB-SMT baseline. In contrast, improvements for the word context are not statistically significant, and the POS contextual features do not improve the baseline PB-SMT model. Our semantic role contextual feature produced moderate improvements over the baseline, both when used individually and in collaboration with other features.

In sum, for large-scale translation, in terms of contextual features, overall supertags, dependency relations and semantic roles seem to be more effective than word- and POS-based models.

6.3.4 Experiments on Increasing Size of Training Sets

We carried out learning curve experiments considering basic features (words and POS tags) as source-language contexts on three different language pairs (English-to-Spanish, English-to-Dutch, and Dutch-to-English), the results of which were reported in Sections 4.3.5. Section 5.3.4 reported the results of learning curve experiments obtained incorporating supertags as source-language contexts into the PB-SMT model. Like basic and supertag contextual features, we carried out learning curve experiments on the English-to-Spanish, Dutch-to-English and English-to-Dutch language pairs by employing deep syntactic and semantic features as source-language contexts. In order to carry out the learning curve experiments with deep syntactic and semantic contextual features, we used the same data sets as used in the previous learning curve experiments.

We report the outcomes of English-to-Spanish and Dutch-to-English learning curve experiments deploying deep syntactic contextual features in the PB-SMT model in Sections 6.3.4.1 and 6.3.4.2, respectively. In Section 6.3.4.3, we report the learning curve experiments which we performed on English-to-Dutch language pair employing deep syntactic and semantic contextual features in the PB-SMT model.

6.3.4.1 English-to-Spanish

As mentioned in Sections 4.3.5 and 5.3.4, the English-to-Spanish training data set was divided into eight different training sets ranging from 10K sentence pairs to 1.64M sentence pairs. To perform experiments on this sequence of training sets, we used both IGTree and TRIBL. As noted earlier, we were able to use only the TRIBL classifier with training sets containing up to 100K sentences due to TRIBL's relatively high memory requirements.

IGTree as classifier: Table 6.19 shows experimental results obtained on English-to-Spanish training sets comprising 10K to 1.64M sentence pairs by employing

IGTree as the classifier. As can be seen from Table 6.19, for all amounts of training sets except two (100K and 1.64M training sets), dependency relation-based systems remain below the Moses baseline according to the performance measured by any of the evaluation metrics. On the 500K and the 1M training sets, the performance of the dependency relation-based SMT models are close to that of the Moses baseline. On the 100K training set, adding the parent relation context feature improves upon the baseline in terms of BLEU, NIST, and METEOR. Moreover, on the 1.64M training set, the PR system produces a slight improvement in BLEU over the Moses baseline.

TRIBL as Classifier: The experimental results obtained using TRIBL as the classifier on the training sets containing 10K to 100K sentence pairs are shown in Table 6.20. When the 10K training set is used, we see that the performance of all context-informed SMT systems (PR, OE, PR+OE) are much closer to the performance of the Moses baseline across all evaluation metrics.

The PR and OE systems produce slight improvements over the Moses baseline at the 20K amount of training data across most evaluation metrics. However, none of the BLEU and METEOR improvements are statistically significant with respect to the baseline.

Adding any of the dependency contextual features improves upon the baseline across all evaluation metrics, when we used 50K amount of training data. We see from Table 6.20 that most of the improvements in METEOR over the Moses baseline are statistically significant, and improvements in BLEU are very close to significance level with respect to the baseline.

When 100K sentence-pairs of training data are used, all dependency contextual features produce modest gains over the Moses baseline across all the evaluation metrics. The PR system produces a statistically significant and the highest improvement in BLEU (PR: 0.33 BLEU points; 1.15% relative) as well as a statistically significant improvement in METEOR (PR: 0.25 METEOR points; 0.81% relative) over

Train. Size	Experiments	BLEU	NIST	METEOR	TER	WER	PER
10K	Baseline	*22.68*	*6.00*	*26.15*	*68.93*	*67.58*	*50.81*
	PR	22.29	5.94	25.83	69.49	68.04	51.21
	OE	22.22	5.94	25.88	69.31	67.84	51.03
	PR+OE	22.01	5.91	25.76	69.51	68.09	51.14
20K	Baseline	*24.58*	*6.38*	*27.77*	*66.51*	*65.16*	*48.71*
	PR	24.45	6.36	27.63	66.92	65.65	49.00
	OE	24.32	6.36	27.64	66.89	65.58	48.93
	PR+OE	24.02	6.32	27.52	67.14	65.83	49.14
50K	Baseline	*27.33*	*6.84*	*29.93*	*64.15*	*62.97*	*46.63*
	PR	27.12	6.82	29.77	64.06	63.09	46.63
	OE	27.22	6.81	29.65	64.22	63.13	46.82
	PR+OE	26.81	6.8	29.65	64.27	63.28	46.76
100K	Baseline	*28.64*	*7.09*	*30.91*	*62.30*	*61.26*	*45.1*
	PR	28.71	7.11	31.06	62.28	61.39	45.02
	OE	28.64	7.06	30.76	62.77	61.73	45.47
	PR+OE	28.49	7.05	30.84	62.6	61.63	45.49
200K	Baseline	*29.96*	*7.32*	*31.90*	*60.97*	*60.01*	*44.06*
	PR	29.56	7.27	31.67	61.41	60.44	44.45
	OE	29.65	7.25	31.70	61.52	60.56	44.51
	PR+OE	29.68	7.28	31.73	61.24	60.28	44.31
500K	Baseline	*31.08*	*7.47*	*32.67*	*59.86*	*58.83*	*43.18*
	PR	30.94	7.45	32.54	60.13	59.19	43.38
	OE	31.07	7.48	32.64	59.93	59.02	43.24
	PR+OE	30.87	7.45	32.52	60.21	59.27	43.39
1M	Baseline	*31.52*	*7.54*	*32.94*	*59.45*	*58.50*	*42.84*
	PR	31.37	7.53	3288	59.46	58.56	42.85
	OE	31.47	7.51	32.86	59.63	58.71	43.03
	PR+OE	31.42	7.52	32.86	59.61	58.65	42.88
1.64M	Baseline	*31.92*	*7.60*	*33.24*	*59.06*	*58.14*	*42.42*
	PR	31.94	7.566	33.19	59.42	58.38	42.65
	OE	31.8	7.59	33.10	59.2	58.22	42.54
	PR+OE	31.76	7.56	32.98	59.44	58.51	42.87

Table 6.19: Results of English-to-Spanish learning curve experiments with deep syntactic and semantic contextual features while employing IGTree as the classifier.

Train. Size	Experi-ments	BLEU	NIST	METEOR	TER	WER	PER
10K	Baseline	*22.68*	*6.00*	*26.15*	*68.93*	*67.58*	*50.81*
	PR	22.56	5.99	26.05	69.09	67.72	50.92
	OE	22.54	5.97	26.07	69.26	67.80	51.02
	PR+OE	22.39	5.98	25.97	69.31	67.97	50.78
20K	Baseline	*24.58*	*6.38*	*27.77*	*66.51*	*65.16*	*48.71*
	PR	24.69	6.402	27.8	66.48	65.16	48.63
	OE	24.64	6.39	27.8	66.62	65.38	48.82
	PR+OE	24.54	6.39	27.78	66.54	65.35	48.76
50K	Baseline	*27.33*	*6.84*	*29.93*	*64.15*	*62.97*	*46.63*
	PR	27.49 (76%)	6.89	30.11 (99.28%)	63.62	62.67	46.16
	OE	27.44 (83%)	6.877	30.01 (87.10%)	63.57	62.64	46.20
	PR+OE	27.49 (81%)	6.891	30.11 (99.27%)	63.64	62.76	46.06
100K	Baseline	*28.64*	*7.09*	*30.91*	*62.30*	*61.26*	*45.1*
	PR	28.97 (94.1%)	7.13	31.16 (99.91%)	62.16	61.18	45.00
	OE	28.81 (91.5%)	7.12	31.10 (99.44%)	62.20	61.17	44.96
	PR+OE	28.86 (53%)	7.11	31.12 (99.45%)	62.33	61.31	45.12

Table 6.20: Results of English-to-Spanish learning curve experiments with deep syntactic and semantic contextual features while employing TRIBL as the classifier.

the Moses baseline. For the 100K training set, we observed statistically significant improvements in METEOR also for other experimental set-ups (OE and PR+OE) of the dependency features.

Learning Curves: We draw learning curves in order to examine the effect of increasing amounts of training data, for both IGTree and TRIBL classifiers. We plot the BLEU score learning curves of the two best-performing context-informed SMT models (PR, OE) for TRIBL and IGTree, as well as for the Moses baseline in Figure 6.4.

In addition, Figure 6.5 shows the BLEU (top), METEOR (centre) and TER (bottom) difference curves of the dependency relation-based SMT systems against the baseline, highlighting the gains and losses achieved. In addition to the PR, OE and the Moses baseline, Figure 6.5 displays the performance of the PR+OE system against the baseline.

Figure 6.4 shows that the PR and OE curves for TRIBL start just below the baseline curve, then cross the baseline curve when more training data is added. The

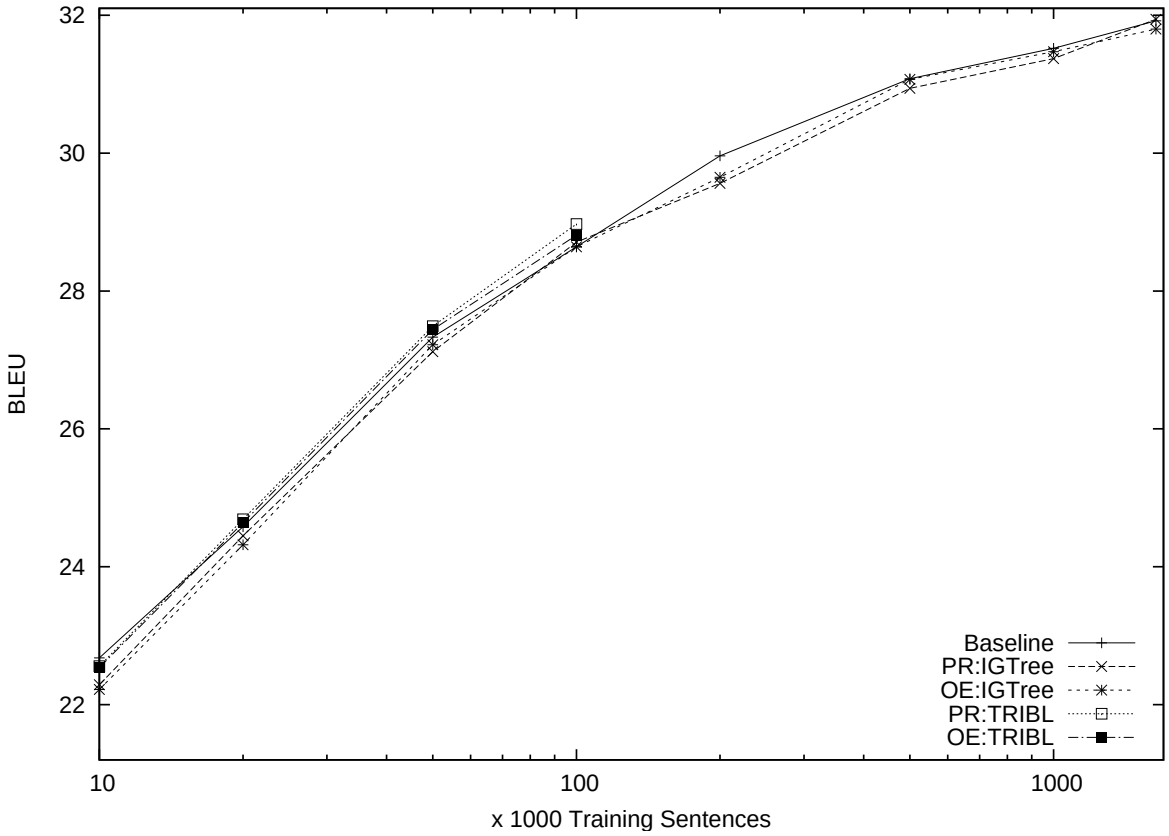

Figure 6.4: BLEU Learning curves comparing the Moses baseline against the two context-informed SMT models (PR, OE) in English-to-Spanish translation task.

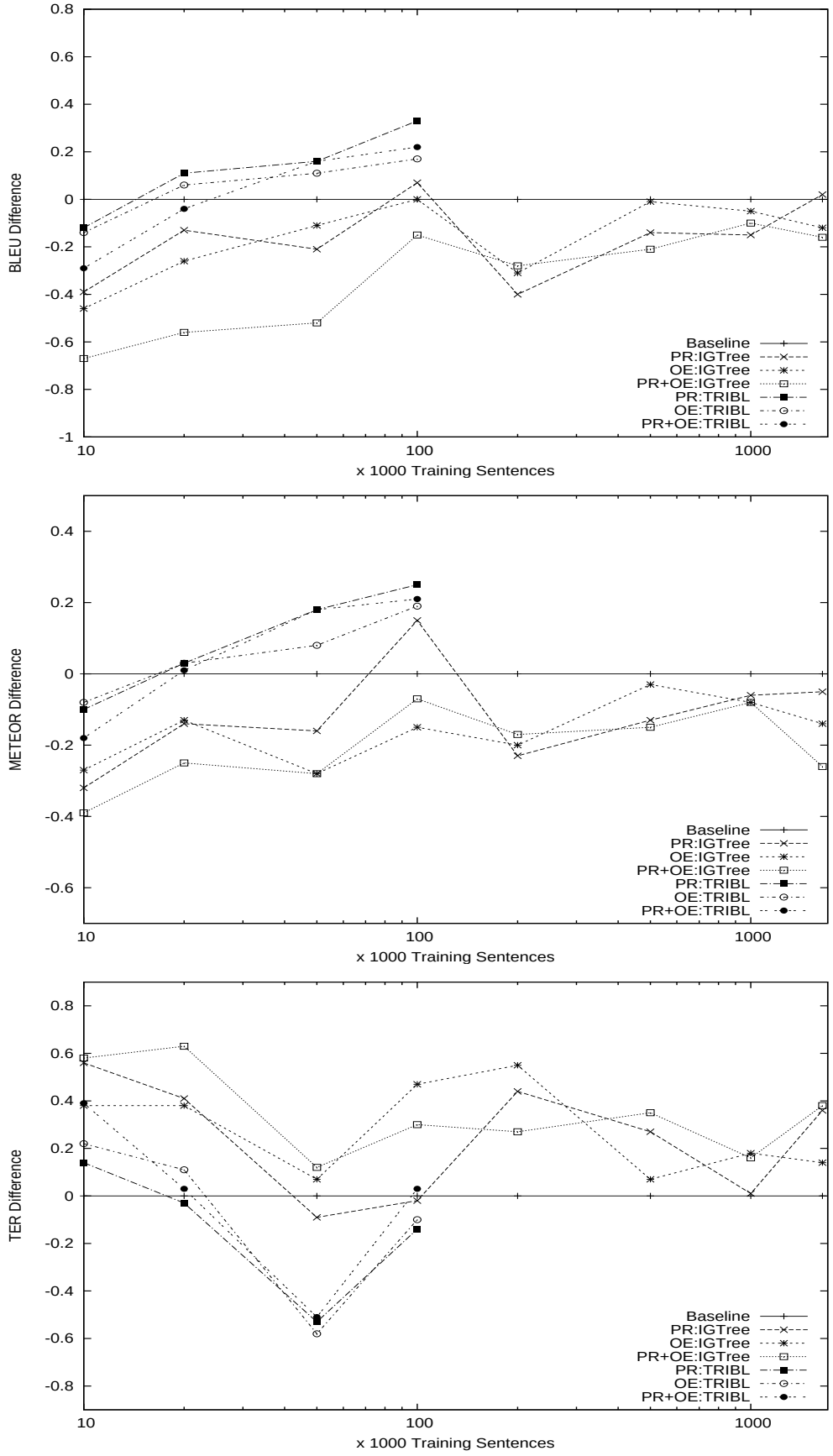

Figure 6.5: BLEU (top), METEOR (centre) and TER (bottom) score-difference curves comparing the Moses baseline against the context-informed SMT models (PR, OE, PR+OE) in English-to-Spanish translation task.

figure also illustrates that the PR and OE curves for IGTree start at a lower level than the baseline curve, and end at the same level as the baseline curve at the largest training set size.

In sum, TRIBL appears to be effective on both small and moderately large-scale data sets. In contrast, IGTree does not offer improvements over the baseline either with the small or the large-scale context-informed models; the performance of the large-scale context-informed models with the IGTree classifier are close or equal to the performance of the Moses baseline.

In this translation task, we carried out an additional analysis on the translations produced by our best-performing context-informed (CI) system (PR)[9] and the Moses baseline. Table 6.21 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 6.22 compares weights of the various translational features of the PR and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	529	493	462	516
Sentence-Level TER	514	499	949	38
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	2.05		2	
Matching Words (%)	59.90		59.65	

Table 6.21: Comparison between translations produced by the best-performing context-informed (CI) system (PR) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.0795	0.0448	0.0091	0.0449	0.0527	0.1151	-0.1451	-	-
PR	0.0963	0.0508	0.0383	0.0009	0.0417	0.1193	-0.1280	0.0401	-0.0010

Table 6.22: Comparison of weights for each translational feature of the two systems (PR and Moses baseline) obtained by MERT training.

6.3.4.2 Dutch-to-English

In this section, we report the outcomes of the Dutch-to-English learning curve experiments. On the Dutch-to-English translation task, we adopted our previously

[9]The best-performing PR system which we used for this analysis was built on 100K training set with TRIBL (cf. Table 6.20).

best-performing experimental set-ups (PR, OE, PR+OE) in order to integrate dependency relation-based contextual features into the PB-SMT model. On this translation task, we used TRIBL as the classifier as we did for the learning-curve experiments with basic contextual features (cf. Section 4.3.5.2 (on page 81)).

Train. Size	Experiments	BLEU	NIST	METEOR	TER	WER	PER
10K	Baseline	*18.31*	*5.14*	*46.73*	*69.13*	*74.06*	*54.77*
	PR	18.39 (46%)	5.20	46.67	68.65	73.31	54.37
	OE	18.35 (38%)	5.18	46.65	68.74	73.35	54.49
	PR+OE	18.34 (25%)	5.16	46.72	68.94	73.52	54.69
20K	Baseline	*20.13*	*5.47*	*49.14*	*67.22*	*72.24*	52.79
	PR	20.28 (30%)	5.52	49.29 (84.9%)	66.87	71.64	52.48
	OE	20.34 (76%)	5.52	49.31 (89.5%)	66.69	71.45	52.45
	PR+OE	20.05	5.47	49.09	67.07	71.85	52.76
50K	Baseline	*23.35*	*5.96*	*52.54*	*63.85*	*68.91*	49.76
	PR	23.56 (93.7%)	6.03	52.45	63.65	68.49	49.68
	OE	23.59 (66.67%)	6.01	52.56 (52%)	63.61	68.66	49.64
	PR+OE	23.48 (27%)	5.99	52.58 (56%)	63.59	68.63	49.68
100K	Baseline	24.72	6.19	54.18	62.30	67.54	48.35
	PR	24.92 (20%)	6.27	54.29 (73.7%)	61.92	67.13	48.10
	OE	24.88 (18%)	6.24	54.36 (87.5%)	61.99	67.08	48.08
	PR+OE	24.98 (66%)	6.24	54.42 (90.2%)	61.99	67.20	48.13
200K	Baseline	25.89	6.36	55.30	61.02	66.34	47.15
	PR	25.96 (20%)	6.46	55.38 (53%)	60.51	65.74	46.86
	OE	25.96 (20%)	6.42	55.33 (53.1%)	60.70	65.95	47.04
	PR+OE	26.04 (53%)	6.42	55.38 (57.8%)	60.73	65.82	47.10
500K	Baseline	26.56	6.53	56.23	59.89	65.02	46.25
	PR	26.82 (81%)	6.61	56.48 (93.7%)	59.63	64.89	46.03
	OE	26.97 (98.5%)	6.61	56.63 (99.48%)	59.48	64.67	46.04
	PR+OE	27.00 (95.1%)	6.59	56.58 (98.14%)	59.57	64.70	46.08
1M	Baseline	27.06	6.63	56.60	59.08	64.29	45.54
	PR	27.41 (91.5%)	6.70	56.98 (97.30%)	58.88	64.08	45.56
	OE	27.26 (86%)	6.68	56.92 (96.9%)	58.87	64.11	45.56
	PR+OE	27.30 (74%)	6.67	56.94 (95.02%)	59.04	64.25	45.60
1.31M	Baseline	27.29	6.68	56.81	58.65	63.96	45.17
	PR	27.60 (84%)	6.74	57.12 (95.17%)	58.71	63.85	45.25
	OE	27.56 (82%)	6.72	57.23 (99.53%)	58.66	63.86	45.27
	PR+OE	27.59 (79%)	6.71	57.17 (97.24%)	58.64	63.89	45.38

Table 6.23: Results of the Dutch-to-English learning curve experiments with deep syntactic contextual features and TRIBL classifier.

The experimental results obtained on the eight training sets containing 10K to 1.31M amounts of sentence pairs are shown in Table 6.23. As can be seen from Table 6.23, moderate improvements are to be observed for dependency relation-based

contextual features for any amount of training set used. Statistically significant improvements in BLEU are seen when the 500K training set is used. As far as other training sets are concerned, most of the improvements in BLEU are close to the significance level with respect to the baseline.

If we look at the performance of the dependency relation-based SMT systems as measured by the METEOR evaluation metric in Table 6.23, we see that most of the improvements in METEOR over the baseline are statistically significant. Moreover, performance measured by other evaluation metrics seem to be quite similar to those measured by BLEU and METEOR.

We draw learning curves of our context-informed SMT systems together with the baseline in order to observe the effect of increasing amounts of training data. Figure 6.6 shows the BLEU learning curves comparing the Dutch-to-English Moses baseline against the two context-based models (PR, OE). Additionally, Figure 6.7 shows respectively BLEU (top), METEOR (centre) and TER (bottom) score-difference curves, highlighting the gains and losses of the context-based models (PR, OE, PR+OE) against the baseline.

In Figure 6.6, we observe that the BLEU learning curves of all the context-informed models (PR, OE) always remain above the baseline curve from the starting point (10K training data) to the end point (1.31M training data).

The top two graphs in Figure 6.7 show the BLEU and METEOR score-difference curves, respectively. We see that the BLEU and METEOR score-difference curves for all the context-informed models (PR, OE, PR+OE) always remain above the baseline curve from the starting point (10K training data) to the end point (1.31M training data). The bottom graph of Figure 6.7 show that the TER score-difference curves of all the context-informed models (PR, OE, PR+OE) mostly remain below the baseline curve, which illustrates the effectiveness of source-language context in this translation task also in terms of TER.

In summary, in the Dutch-to-English translation task, the dependency relations appear to be effective source-language context features in PB-SMT according to all

evaluation metrics.

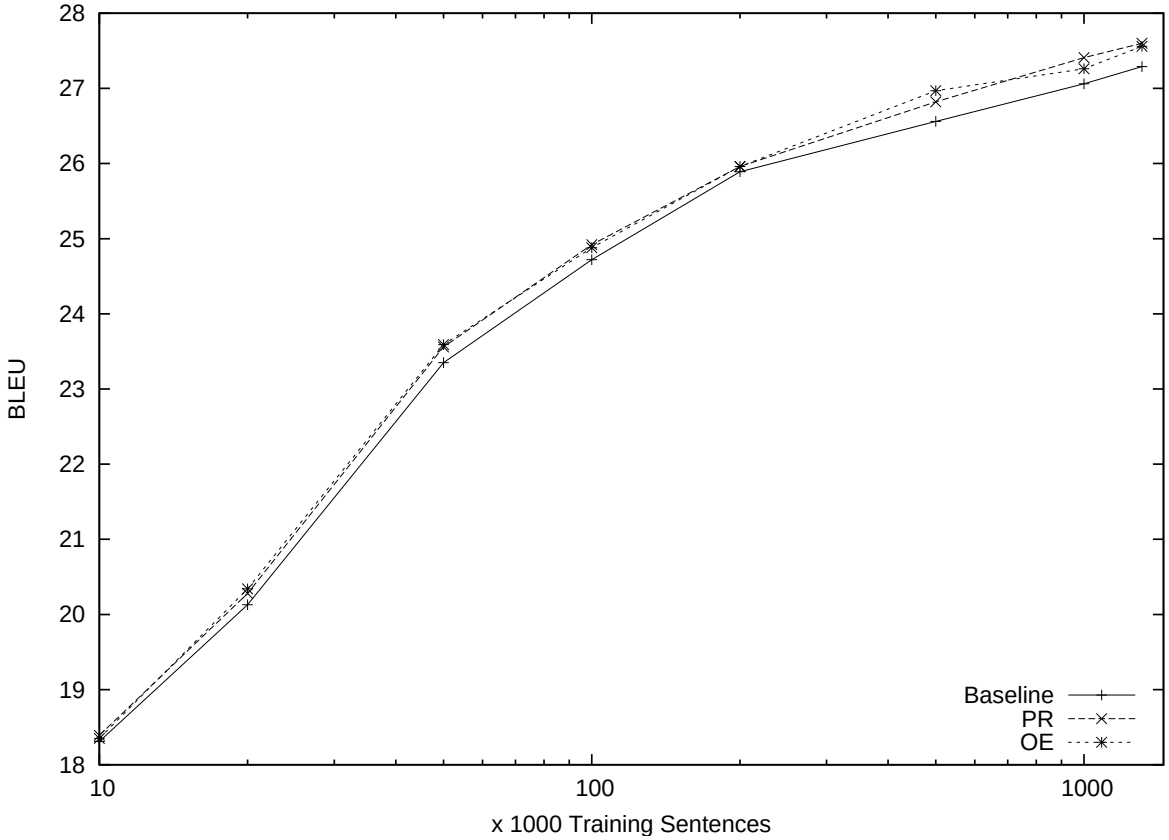

Figure 6.6: BLEU Learning curves comparing the Moses baseline against the two context-informed SMT models (PR, OE) in Dutch-to-English translation task.

As an additional point of analysis, we compare translations produced by our best-performing context-informed (CI) system (PR)[10] with those by the Moses baseline. Table 6.24 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 6.25 compares weights of the various translational features of the Word±2 and the baseline systems obtained by MERT training.

6.3.4.3 English-to-Dutch

In this section, we report the outcomes of the English-to-Dutch learning curve experiments. In order to conduct English-to-Dutch learning curve experiments, we

[10]The best-performing PR system which we used for this analysis was built on the largest available training data (1.31M sentences) with TRIBL classifier (cf. Table 6.23).

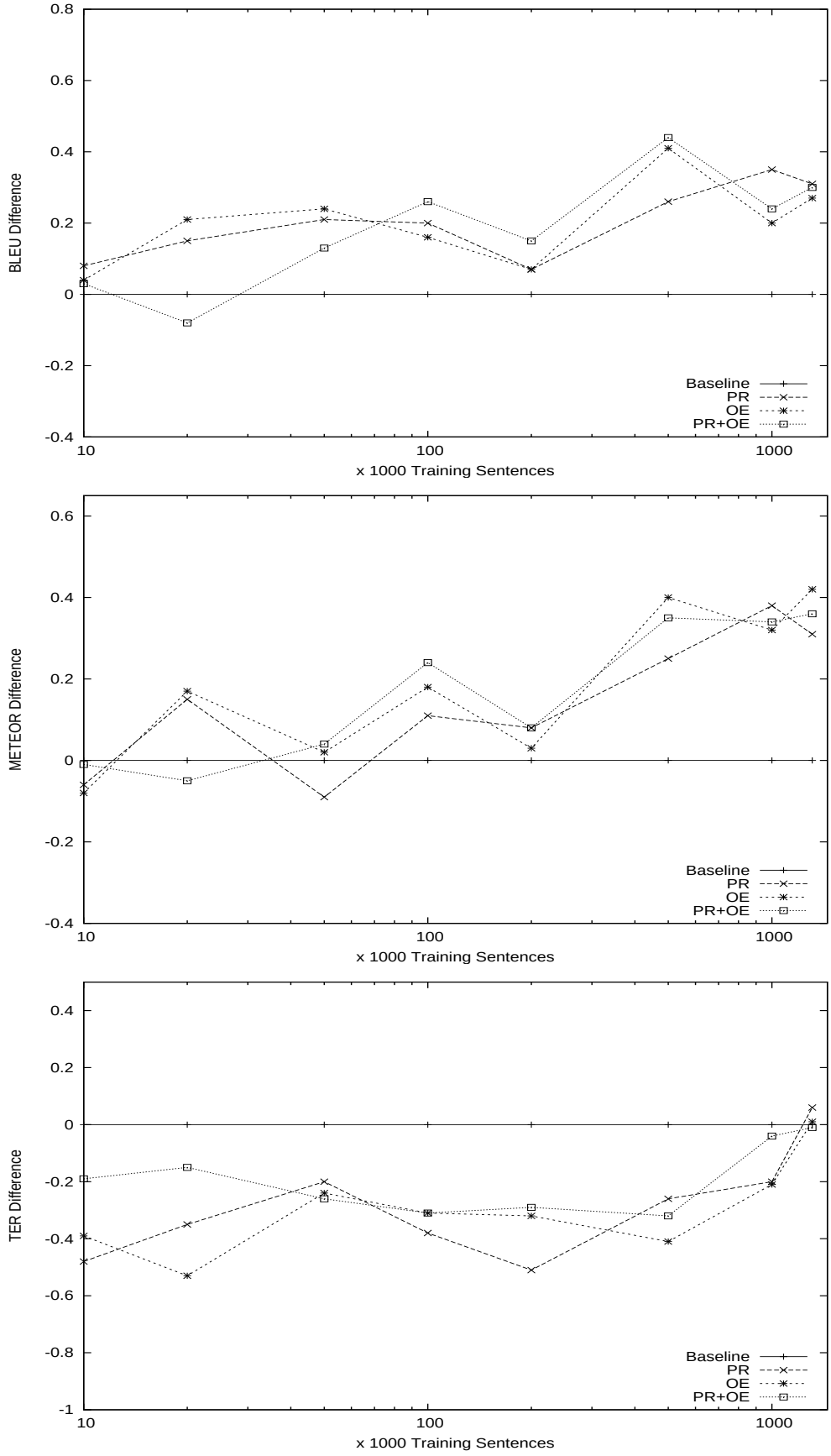

Figure 6.7: BLEU (top), METEOR (centre) and TER (bottom) score-difference curves comparing the Moses baseline against the context-informed SMT models (PR, OE, PR+OE) in Dutch-to-English translation task.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	278	247	202	273
Sentence-Level TER	279	278	402	41
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	4.3		4.2	
Matching Words (%)	65.41		65.23	

Table 6.24: Comparison between translations produced by the best-performing context-informed (CI) system (PR) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1086	0.0641	0.0107	0.0569	0.0867	0.0977	-0.2459	-	-
PR	0.1018	0.0292	0.0838	0.0106	0.0620	-0.1129	-0.1958	0.0467	0.0062

Table 6.25: Comparison of weights for each translational feature of the two systems (PR and Moses baseline) obtained by MERT training.

consider the previously best-performing experimental set-ups comprising each feature type (dependency relation (PR, OE, PR+OE), semantic role (AL, PS-AL)).

In order to perform the deep syntactic and semantic feature-based learning curve experiments, we used the same English-to-Dutch data set which we used to conduct the learning-curve experiments with the basic contextual features (cf. Section 4.3.5.3) as well as with the supertag-based contextual features (cf. Section 5.3.4.2).

The experimental results obtained on the increasing amounts of training sets are shown in Table 6.26. As can be seen from the table, adding deep syntactic and semantic contextual features (individually or jointly) improves upon the Moses baseline when the training data used contains 20K or more sentence pairs.

The dependency relation-based PR system produces statistically significant improvements in BLEU (0.25 BLEU points; 1.06% relative) when the training set contains 200K sentence pairs. As far as the other training sets are concerned, PR produces consistent improvements in BLEU over the baseline, some of which are very close to the significance level. Like PR, other dependency feature-based systems (OE and PR+OE) consistently outperform the Moses baseline.

When adding semantic features as source language context, the PB-SMT system gives statistically significant improvements in BLEU (PS-AL: 0.39 BLEU points; 2.04% relative) over the Moses baseline when 20K sentence-pairs training data are used. On 100K, 200K, and 1.31M training sets, improvements in BLEU for the

PS-AL over the baseline are very close to the significance level. Table 6.26 shows that the AL system performs slightly worse than the PS-AL, but shows consistency in improving over the baseline. As far as the other evaluation metrics are concerned, we see from Table 6.26 that adding semantic features (AL and PS-AL) as source-language contexts consistently improves over the PB-SMT baseline.

Figure 6.8 illustrates BLEU learning curves comparing the Moses baseline against the best-performing context-informed SMT models (Dependency relations: PR, Semantic roles: PS-AL) for each feature type.

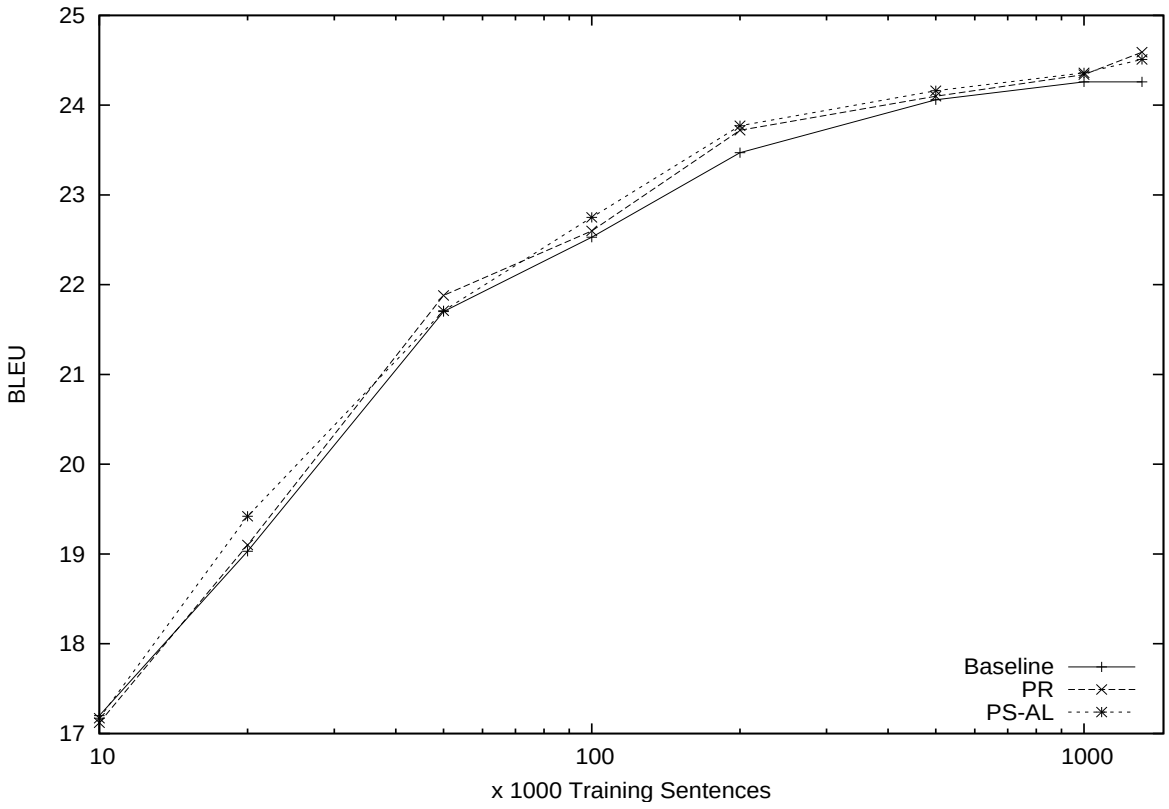

Figure 6.8: BLEU Learning curves comparing the Moses baseline against SMT models added with deep syntactic and semantic contextual features in the English-to-Dutch translation task.

We see from the graph in Figure 6.8 that the dependency and semantic feature-based BLEU curves (PR, PS-AL) show consistency in residing mostly above the baseline BLEU curve from the starting point (10K training data) to the end point (1.31M training data).

In addition to the BLEU learning curves shown in Figure 6.8, the three graphs

Train. Size	Experiments	BLEU	NIST	METEOR	TER	WER	PER
	Baseline	*17.20*	*4.99*	*43.74*	*72.40*	*75.39*	*57.17*
	PR	17.12	5.01	43.69	71.95	75.06	57.11
10K	OE	16.90	4.97	43.65	72.62	75.68	57.38
	PR+OE	17.05	5.00	43.83 (80.6%)	72.18	75.21	57.15
	AL	16.95	5.00	43.51	72.19	75.31	57.17
	PS-AL	17.17	5.01	43.62	71.93	74.90	57.18
	Baseline	*19.03*	*5.31*	*46.30*	*70.33*	*73.69*	*55.08*
	PR	19.10 (56.6%)	5.34	46.26	70.16	73.54	54.96
20K	OE	19.20 (83.1%)	5.34	46.54 (94.4%)	70.34	73.78	55.03
	PR+OE	19.05	5.33	46.39 (73.5%)	70.23	73.59	55.05
	AL	19.05	5.33	46.11	70.45	73.86	55.37
	PS-AL	19.42 (99.7%)	5.36	46.39 (70.42%)	69.92	73.23	54.90
	Baseline	*21.70*	*5.74*	*49.31*	*67.32*	*71.19*	*52.48*
	PR	21.88 (80%)	5.78	49.33	67.15	70.82	52.41
50K	OE	21.71	5.73	49.34	67.58	71.32	52.72
	PR+OE	21.73 (34%)	5.76	49.37 (67%)	67.46	71.21	52.54
	AL	21.53	5.75	48.97	67.37	71.12	52.58
	PS-AL	21.71	5.74	49.15	67.34	71.16	52.72
	Baseline	*22.53*	*5.88*	*50.30*	*66.42*	*70.48*	*51.84*
	PR	22.60 (66%)	5.92	50.36 (68%)	65.96	70.12	51.65
100K	OE	22.52	5.88	50.44 (81.8%)	66.39	70.64	51.91
	PR+OE	22.65 (67%)	5.92	50.55 (95.6%)	66.13	70.20	51.63
	AL	22.50	5.90	50.26	66.12	70.27	51.78
	PS-AL	22.75 (91.9%)	5.91	50.36 (61%)	66.04	70.21	51.71
	Baseline	*23.47*	*6.04*	*51.46*	*65.30*	*69.40*	*50.90*
	PR	23.72 (99%)	6.08	51.65 (89.70%)	64.96	69.12	50.74
200K	OE	23.55 (40%)	6.05	51.51 (62.8%)	65.17	69.28	50.84
	PR+OE	23.51 (89%)	6.05	51.53 (63.4%)	65.31	69.28	50.88
	AL	23.63 (93.4%)	6.07	51.49 (53%)	65.12	69.26	50.95
	PS-AL	23.77 (94.1%)	6.06	51.49 (48%)	65.16	69.28	50.99
	Baseline	*24.06*	*6.11*	*52.06*	*64.59*	*68.58*	*50.36*
	PR	24.10 (47%)	6.15	52.19 (84.7%)	64.46	68.51	50.22
500K	OE	24.02	6.12	52.13 (78.9%)	64.35	68.60	50.17
	PR+OE	23.91	6.13	52.06	64.46	68.68	50.22
	AL	24.01	6.15	52.22 (86.41%)	64.37	68.58	50.10
	PS-AL	24.16 (52%)	6.14	52.19 (74.9%)	64.41	68.52	50.20
	Baseline	*24.26*	*6.17*	*52.39*	*64.55*	*68.72*	*50.12*
	PR	24.34 (43%)	6.23	52.46 (72.35%)	63.99	68.09	49.64
1M	OE	24.08	6.15	52.19	64.32	68.54	50.16
	PR+OE	24.29 (54%)	6.18	52.50	64.24	68.43	50.10
	AL	24.17	6.14	52.29	63.83	68.37	49.63
	PS-AL	24.36 (53%)	6.19	52.42 (54%)	64.23	68.46	50.00
	Baseline	*24.26*	*6.17*	*52.68*	*64.36*	*68.80*	*50.02*
	PR	24.59 (89.7%)	6.27	52.79 (83.8%)	63.78	68.27	49.54
1.31M	OE	24.41 (88.9%)	6.19	52.56	63.99	68.35	49.84
	PR+OE	24.41 (82.3%)	6.23	52.67	64.00	68.45	49.81
	AL	24.34 (44%)	6.23	52.64	63.83	68.37	49.63
	PS-AL	24.51 (91.8%)	6.21	52.72 (58%)	64.23	68.69	49.92

Table 6.26: Results of the English-to-Dutch learning curve experiments with TRIBL classifier comparing the effect of supertag context and Moses baseline.

in Figure 6.9 show respectively BLEU (top), METEOR (centre) and TER (bottom) score-difference curves, highlighting the gains and losses against the Moses baseline.

The top graph in Figure 6.9 shows BLEU score-difference curves of five context-informed SMT systems (PR, OE, PR+OE, AL, PS-AL) against the Moses baseline. We see from the graph that most semantic and dependency feature-based BLEU curves reside above the baseline BLEU curve for the most training set sizes. The central graph of Figure 6.9 shows METEOR score-difference curves. We see that most of the METEOR score-difference curves show consistency in residing mostly above the baseline curve for all amounts of training data used.

The bottom graph in Figure 6.9 displays TER score-difference curves, from which we see that most of the TER score-difference curves show consistency in residing below the baseline.

In summary, in this translation task, all the metrics (BLEU, METEOR, TER) suggest that deep syntactic and semantic features are effective source-language contexts in PB-SMT.

As an additional point of analysis, we compare translations produced by our best-performing context-informed (CI) system (PR)[11] with those by the Moses baseline. Table 6.27 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 6.28 compares weights of the various translational features of the PR and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	231	228	211	330
Sentence-Level TER	293	232	438	37
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	3.8		3.8	
Matching Words (%)	61.98		61.74	

Table 6.27: Comparison between translations produced by the best-performing context-informed (CI) system (PR) and the Moses baseline.

[11]The best-performing PR system which we used for this analysis was built on the largest available training data (1.31M sentence pairs) (cf. Table 6.26).

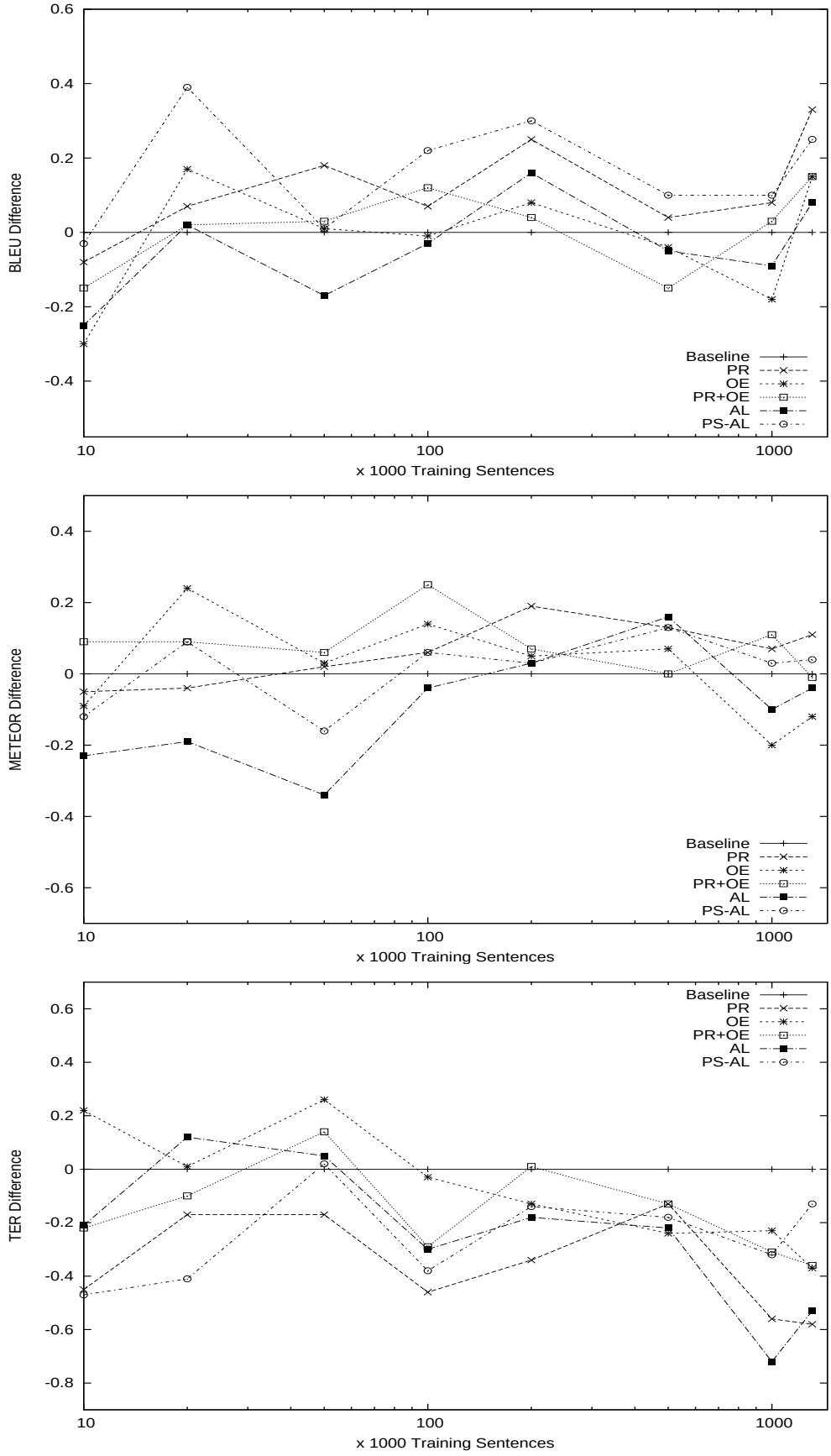

Figure 6.9: BLEU (top), METEOR (centre) and TER (bottom) score-difference curves comparing the Moses baseline against dependency and semantic feature-based SMT models in English-to-Dutch translation task.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Moses	0.1072	0.0102	0.0509	0.1103	0.0468	0.0936	-0.2689	-	-
PR	0.0859	0.0122	0.0433	0.0746	0.0673	0.0415	-0.2198	0.016	0.0004

Table 6.28: Comparison of weights for each translational feature of the two systems (PR and Moses baseline) obtained by MERT training.

6.3.5 Analysis of Learning Curve Experiments

In this section we summarize the outcomes of the learning curve experiments reported in the above sections. We report a comparative overview of the effectiveness of the different contextual features in the learning-curve experiments on three different language pairs (English-to-Spanish, Dutch-to-English, and English-to-Dutch). Moreover, we compare the effectiveness of deep syntactic and semantic contextual features with that of the basic and supertag-based contextual features reported in the previous chapters (cf. Sections 4.3.5 (on page 76) and 5.3.4 (on page 121)).

The learning curve experiments on the English-to-Spanish language pair show that supertags and dependency relations are more effective source-language contextual features than neighbouring words and POS tags. On the English-to-Spanish translation task, the TRIBL classifier produces gains at smaller training sets, though not at the smallest sizes (10K training sentences). When using IGTree as the classifier, the context-informed SMT systems equal the baseline at the largest amount of training data.

The learning curve experiments on the Dutch-to-English and English-to-Dutch language pairs show that rich and complex syntactic contexts (supertags, dependency relations and semantic roles) outperform basic contexts (words and POS tags) in terms of effectiveness in the small- and large-scale translations.

For the Dutch-to-English translation task, we see that all types of contextual features improve upon the PB-SMT baseline across all sizes of training data. On this language pair, deep syntactic and semantic feature-based SMT systems outperform the basic contextual feature-based SMT systems when larger amounts of training data are used.

We see from the learning curve experiments on the English-to-Dutch language

pair that the deep syntactic and semantic features show consistency in improving over the baseline for most sizes of training data used. In contrast, the supertag-based contextual features give the highest improvements over the PB-SMT baseline while using larger amounts of training data, but do not show consistency in improving over the baseline when smaller amounts of training data are used.

6.3.6 Translation Analysis

We performed manual qualitative analysis comparing the translation output of the best-performing systems with those of the Moses baseline systems. In order to carry out the manual evaluation, we randomly sampled 50 test set sentences.

First, we looked at the translated output of our best-performing system (PR+OE+POS±2†) against that of the Moses baseline in the small-scale Dutch-to-English translation task (cf. Section 6.3.1.1). We observed that the (PR+OE+POS±2†) system generates a more fluent and adequate output than the baseline for 11.7% of the test set sentences. The following are two such translation examples:

(7)	Dutch:	heeft mijn vader je gestuurd ?
	Reference:	did my father send you ?
	PR+OE+POS±2†:	*did* my father *send* you ?
	Baseline:	my father *has sent* you ?
(8)	Dutch:	daarna vraag je om informatie .
	Reference:	then you call for information .
	PR+OE+POS±2†:	then *you ask* for information .
	Baseline:	then *ask you* to get information .

In Example (7) we observe that the baseline does not select the proper word order of an interrogative sentence, and selects a less optimal (more literal) translation of the Dutch auxiliary verb *heeft* (*has*), while the PR+OE+POS system generates a translation identical to the reference translation. In example (8), the baseline system again selects a less appropriate (Dutch) word order.

We carried out a manual analysis on the translation output of the best-performing

system (PR) against the Moses baseline in the large-scale Dutch-to-English translation task (cf. Section 6.3.4.2). Examples (9) and (10) show how our best-performing system (PR) improves over the Moses baseline in this task.[12] Again, the baseline makes sub-standard choices, leaving out the main verb in (9) and failing to approximate the intended meaning, as it also mangles the negation in example (10). However, on that example the PR system also fails to generate a fluent and adequate output.

(9)	Dutch:	De afschaffing van de doodstraf maakt als het ware deel uit van onze cultuur .
	Reference:	The abolition of the death penalty belongs , as it were , to our culture .
	PR:	The abolition of the death penalty , as it were , is part of our culture .
	Baseline:	The abolition of the death penalty , as part of our culture .
(10)	Dutch:	Alleen Koeweit heeft het stemrecht enkel verleend aan mannen en niet aan vrouwen .
	Reference:	Only Kuwait extended the vote to men and not to women .
	PR:	Only Kuwait has the right to vote to men and not only granted to women .
	Baseline:	Only Kuwait has just granted the right to vote , not to men and women .

In translation example (9), we see that PR generates a grammatical translation, and the translation conveys the same meaning as the source sentence. In contrast, the Moses baseline neither generates a grammatical translation, nor does the translation convey a meaning similar to the input sentence. The translations in example (10) tend to follow a similar trend to the translations in example (9).

6.4 Experiments with Context-Informed Hierarchical PB-SMT

In addition to the basic and supertag contextual features, we integrate deep syntactic information as a source language context into the Hiero model (Chiang, 2007).

[12]We see from examples that the translations produced by the PR system are slightly semnatically distant from the reference transaltions but far better than those produuced by the baseline in terms of fluency and adequency.

In Section 4.4 (cf. page 89), we reported the experimental results obtained by integrating basic contextual features into the Hiero model. Section 5.4 (cf. page 138) demonstrated the experimental results obtained employing supertags as source-side contextual features in the Hiero model. In the following section, we report the experimental results obtained employing dependency relations as source-language context into the Hiero system.

6.4.1 Experimental Results

Exp.	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*21.92*	*5.29*	*43.06*	*56.72*	*55.43*	*48.60*
PR	22.44 (90%)	5.32	43.61 (96.19%)	56.64	55.67	48.28
OE	22.47 (94.9%)	5.32	43.54 (96.94%)	56.47	55.37	48.3
PR+OE	22.29 (52%)	5.31	43.33 (84%)	56.68	55.56	48.20

Table 6.29: Experimental results with dependency features, compared against Hiero baseline.

English-to-Dutch translation was carried out on the Open Subtitles corpus (cf. Section 3.6). Table 6.29 shows the experimental results obtained by applying dependency relation-based contextual features into the Hiero model. Moderate improvements over the Hiero baseline are observed with the addition of PR (0.52 BLEU points; 2.37% relative increase), OE (0.55 BLEU points; 2.51% relative), and PR+OE (0.37 BLEU points; 1.69% relative). As can be seen from Table 6.29, OE and PR produce respectively the highest and the second highest improvements in BLEU over the baseline, which are very close to the significance level.

As far as the METEOR evaluation metric is concerned, PR produces the highest improvement (0.55 METEOR points; 1.26% relative) over the baseline. Moderate improvements in METEOR are observed for the OE (0.48 METEOR points; 1.13% relative) as well. Contrary to the improvements in BLEU, these improvements in METEOR with respect to the baseline are statistically significant.

Improvements in TER for PR (a reduction of 0.08 TER points), OE (0.25 TER points), and PR+OE (0.04 TER points) features are somewhat small compared to improvements in METEOR and BLEU.

Exp.	BLEU	NIST	METEOR	TER	WER	PER
Baseline	*21.92*	*5.29*	*43.06*	*56.72*	*55.43*	*48.60*
Word±2+PR	22.24 (80%)	5.32	43.84 (93.6%)	56.55	55.41	48.36
Word±2+OE	22.16 (57%)	5.34	43.40 (84.15%)	56.08	54.98	48.27
Word±2+PR+OE	22.41 (81%)	5.33	43.34 (71.41%)	56.3	55.19	48.32

Table 6.30: Experimental results with combined features, compared against Hiero baseline.

We also performed experiments in which we combined the lexical features with the dependency relation-based features. The results of these experiments are shown in Table 6.30. Combining dependency features with Word±2 features causes the system performance to deteriorate. The Hiero systems with combined features give smaller improvements than those with individual features (Word±2+PR: 0.32 BLEU points improvement, 1.46% relative; Word±2+OE: 0.24 BLEU points improvement, 1.09% relative). However, we also see from Table 6.30 that the Word±2+PR+OE model produces a 0.49 point improvement in BLEU (2.23% relative) over the Hiero baseline, which is slightly better than that produced by the PR+OE system (the result of the PR+OE system was displayed in Table 6.29).

The best METEOR score (an improvement of 0.78 METEOR points; 1.81% relative increase) is obtained when words are combined with the PR dependency feature. Moderate improvements in METEOR over the Hiero baseline are observed for the Word±2+OE and the Word±2+PR+OE features. On the TER evaluation metric, Word±2+PR, Word±2+OE, and Word±2+PR+OE produce reductions of 0.17, 0.64, and 0.42 TER points below the Hiero baseline, respectively.

As an additional point of analysis, we compare translations produced by our best-performing context-informed (CI) system (OE) (cf. Table 6.30) with those by the Hiero baseline. Table 6.31 shows how two systems differ from each other in terms of sentence-level automatic evaluation measures (BLEU and TER) and closeness to the reference set. Additionally, Table 6.32 compares weights of the various translational features of the OE and the baseline systems obtained by MERT training.

	CI>Moses	CI<Moses	CI=Moses	Zero
Sentence-Level BLEU	32	19	161	788
Sentence-Level TER	65	57	759	119
Closeness to Reference Set	CI		Moses	
Matching Translations (%)	13		12.4	
Matching Words (%)	58.30		57.66	

Table 6.31: Comparison between translations produced by the best-performing context-informed (CI) system (OE) and the Moses baseline.

System	λ_{lm}	λ_{btp}	λ_{blexp}	λ_{ftp}	λ_{flexp}	λ_{phrpty}	λ_{glue}	λ_{wrdpty}	λ_{mbl}	λ_{best}
Hiero	0.0823	0.0205	0.0217	0.1702	0.0409	-0.3477	0.3421	0.095	-	-
OE	0.1005	0.0091	0.0036	0.1128	0.2983	-0.3796	0.3715	0.0635	0.0034	0.0672

Table 6.32: Comparison of weights for each translational feature of the two systems (OE and Hiero baseline) obtained by MERT training.

6.4.2 Discussion

We have successfully incorporated deep syntactic features (dependency relations) as source-language contexts into the state-of-the-art Hiero system (Chiang, 2007) for English-to-Dutch translation. In this translation task, we compare the effectiveness of the deep syntactic contextual features with that of the basic and supertag-based contextual features, which we reported in the previous chapters (cf. Sections 4.4.1.2 (on page 91) and 5.4.1.2 (on page 140)). We observe that both the supertags and the dependency features appear to be more effective as a source-language context than the basic contextual features (words and POS tags). When the supertag- and dependency relation-based contextual features are employed individually, both yield quite similar improvements. However, supertags together with lexical features outperform both the combined dependency relation and lexical feature-based SMT systems.

6.5 Summary

In this chapter, we introduced deep syntactic and semantic parse information as source-language contexts into the state-of-the-art SMT models (Koehn et al., 2003; Chiang, 2007). We used the same data sets and language pairs which were used for the studies with basic and supertag contextual features in Chapters 4 and 5.

We reported a series of experimental results obtained by integrating deep syntactic features (grammatical dependency relations) into the PB-SMT model (Koehn et al., 2003). We found that the deep syntactic features appear to be useful source-language contexts for both small- and large-scale translations. In order to obtain context-sensitive phrase translations, we employed two different memory-based classification algorithms: IGTree and TRIBL. As observed in previous chapters, we also saw in this chapter that TRIBL is more effective than IGTree in improving MT quality. In addition to dependency relations, the chapter introduced semantic roles as a new contextual feature in PB-SMT. In a large-scale English-to-Dutch translation task, semantic roles as a contextual feature significantly improved over the PB-SMT baseline, both when used individually and in collaboration with other features. Moreover, this chapter provided a comparative overview of the utility of various contextual features in PB-SMT. We found that while integrating deep syntactic and semantic contexts into the PB-SMT model, the system shows more consistency compared to the supertag-based context-informed systems. Nevertheless, supertag contexts prove to be more useful than deep syntactic and semantic contexts when using larger amounts of training data.

Finally, we showed that dependency relations can also be modelled as a useful source-language context in the Hiero model. Comparing the usefulness of various contextual features in the Hiero model in an English-to-Dutch translation task, we observed that both supertags and dependency relations produce similar improvements over a Hiero baseline. We discovered that a combination of supertag and lexical features proved to be most effective contexts in the Hiero model; in contrast, combining dependency relations and neighbouring words did not contribute much.

Chapter 7

Conclusions and Future Work

7.1 Contributions of this Thesis

In this thesis, we presented a large set of experimental results obtained using a range of features as source-language context to better enable the state-of-the-art SMT systems (Koehn et al., 2003; Chiang, 2007) to select appropriate target-language phrases for consideration in the generation of the most probable translation given the input. Such features include neighbouring position-specific lexical and part-of-speech features of words surrounding the phrase to be translated, information linking the head word of the focus phrase to its syntactic context in terms of supertags or dependency relations, as well as semantic dependencies the source phrase encapsulates. We explored a range of language pairs with featuring typologically different languages, and examined the scalability of our research on larger amounts of training data.

As far as the context-sensitive PB-SMT systems are concerned, the most significant improvements observed in our experiments involve the integration of long-distance contextual features, such as:

- dependency relations in combination with part-of-speech tags in Dutch-to-English subtitle translation,
- the combination of dependency parse and semantic role information in English-

to-Dutch parliamentary debate translation,

- dependency parse information in Dutch-to-English parliamentary debate translation,
- CCG and LTAG supertag features in English-to-Chinese translation.

As far as scalability is concerned, when our PB-SMT systems were trained with larger amounts of parallel data, the effects of the source-language context are lessened somewhat, but remain statistically significant in some cases. For English-to-Dutch, for example, while the POS-based model failed to contribute positively, our dependency- and supertag-based models continued to be effective. Furthermore, use of semantic roles as a source-language discriminative feature showed encouraging improvements over the PB-SMT baseline.

When we varied the amount of English-to-Spanish Europarl training data used from 10,000 to 1.64 million sentences in a learning curve experiment, the resulting curves demonstrated that gains obtained by our source-language contextual models cannot be expected to occur given any amount of training data. We observed that the TRIBL classifier obtains gains at small training set sizes, though not at the smallest sizes (10,000 training sentences). IGTree, on the other hand, disappointed by requiring the maximal amount of training data (1.64 million sentences) just to equal the baseline. Furthermore, learning curve experiments on the Dutch-to-English and English-to-Dutch language pairs show that rich and complex syntactic features were able to surpass basic features (words and POS tags) as source-language context features in both small- and large-scale translation tasks. Moreover, the outcomes of our manual analysis conducted on the MT outputs of several context-informed models against the respective baselines justify our claims established on the basis of the gains obtained on several automatic evaluation measures. We believe that, in general, learning curve experiments give a more complete overview of relative gains when more data is available. In order to obtain better MT performance, using more training data seems to be most effective.

We compared the integration of sentence-similarity features to the integration of supertag features into the PB-SMT system. Furthermore, we performed experiments by integrating both feature types collectively into the PB-SMT model on small- and large-scale data sets. We achieved the highest improvement over the baseline when the global context (sentence-similarity) was combined with the local context (i.e. supertags). Thus, sentence similarity-based source-context modelling proves to be a useful means to improve lexical selection in PB-SMT.

We employed our source context modelling into the state-of-the-art PB-SMT for an English-Hindi transliteration task. We found that our context-sensitive transliteration system achieved substantial improvement over the baseline. This piece of work can be viewed as a successful application of our context-sensitive PB-SMT model to a different NLP application, namely machine transliteration.

We have also shown that source-language contextual features can be integrated successfully into the state-of-the-art Hiero system. As in PB-SMT, we explored basic features as well as rich and complex syntactic features (supertags and grammatical dependency relations) as source-language contexts in Hiero. We carried out experiments on two different language pairs: English-to-Hindi and English-to-Dutch. Considering only individual contextual features, the system produced significant gains over the Hiero baseline for any of the contextual features used. We observed the highest improvement over the baseline when supertags were combined with lexical context in the English-to-Dutch translation task, whereas adding dependency features did not seem to contribute much where translation quality is concerned.

To summarize our findings, we have shown that whatever language pair is deployed, using source-language context system generally produces better translations compared to baseline SMT systems. To be more precise, if one has a parser available for the source language at hand, integrating syntactic dependency information pertaining to the current input string can generate improved translation quality. Alternatively, if no such parser is available, then POS or supertag information can be useful; but even if this is absent, then taking the neighbouring words into account

is also likely to be effective. Such source-language contextual models become less effective when scaling to large amounts of parallel data, yet even here, statistically significant scores are still to be seen. We come to this conclusion having carried out experiments on a wide range of language pairs, and on a variety of domains of training material.

7.2 Future Work

In this section, we provide some potential avenues for future research work as follows:

- In this thesis, we have demonstrated how our source-language contextual models can benefit both PB-SMT (Koehn et al., 2003) and HPB-SMT (Chiang, 2007). We are certain that other SMT systems such as Syntax-Augmented Machine Translation (SAMT) (Zollmann and Venugopal, 2006) could also take advantages of our source-language contextual models, as could any potentially new MT systems that have yet to appear, but which can be expressed in a log-linear framework.

- In order to obtain context-sensitive translation, we used mainly two memory based classifiers: IGTree and TRIBL (Daelemans and van den Bosch, 2005). In future we would like to introduce another classification algorithm: TRIBL2 (Daelemans et al., 1999). Like TRIBL, TRIBL2 is a hybrid combination of IGTREE and IB1. The main difference between TRIBL and TRIBL2 is that the latter does not use threshold parameter. In other words, in TRIBL, we explicitly provide a fixed point in the feature ordering where IGTREE is succeeded by IB1, while TRIBL2 determines this switching point automatically per classification. In this manner, TRIBL2 offers a fairly optimal use of IB1 which is only invoked when mismatching (feature) occurs. Another advantage of TRIBL2 over TRIBL is that the former is faster than the latter in terms of classification time.

- We tried to investigate the actual role of memory-based classifier with a deeper look at the classification results. A classification example has been provided in Section 5.3.7 (on page 136) in order to show how context-sensitive translations are derived. The Section 5.3.7 also compares context-dependent translation probabilities (memory-based classification's scores) with context-independent translations probabilities (baseline scores) considering a particular English phrase in a English-to-Spanish translation task, from which we found that the memory-based translation model appear to be more effective than the maximum-likelihood estimation-based translation model. In future, we would like to dive into the advantages that memory-based classifiers bring. In other words, we want to carry out a detail investigation on how to utilize the memory-based classifiers optimally for this kind of task.

- Following the work of (Hassan et al., 2006, 2007, 2008; Hassan, 2009), we aim to develop fully supertagged SMT systems, with supertags deployed as source language context (as in this thesis), as well as in the target language model and the target side of the t-table. We have been made aware that a German version of the CCGBank may be available, but so far we have been unable to verify this. We will continue to pursue this line of investigation, with a view to benefiting from the clear advantages that supertags bring to bear in each phase of the translation process.

- We introduced semantic roles as a new contextual feature in PB-SMT (Koehn et al., 2003). In future, we would like to introduce semantic roles as a source language context in the state-of-the-art Hiero model (Chiang, 2007). We also intend to model sentence-similarity features as a source-language context in the Hiero system.

  As far as experiments with the Hiero system are concerned, our experiments have focused on a standard but medium sized data set. Despite the challenges in training classifiers with large sets of instances for Hiero (cf. page 144), we

intend to further validate our conclusions by scaling up to larger data sets, and perform learning curve experiments to observe changes in the relative differences between using different types of additional source-side contextual features.

- Finally, apart from experimenting with still more language pairs and different types of training data, we would like to provide a comprehensive guide to how best to combine different source-language contextual features where more than one type is available, and if possible, to predict *a priori* – perhaps based on the combination of language pair and training data type – the optimal features to use in such circumstances. As a first step we would investigate the influence of the degree to which a domain triggers formulaic language.[1] If a domain contains largely formulaic language,[2] selecting only simple lexical features such as neighbouring words could already be effective, as the generalizing power of more abstract linguistic features is not needed. The reverse may be the case in more open, less formulaic domains. Our memory-based classifiers could be used to provide a quantitative estimate of how similar unseen sequences are to the training sentences, much like a fuzziness score in translation memories or example-based machine translation.

[1]Formulaic language might have special kinds of processing, in which proper linguistic procedures are avoided in order to form words with morphemes, phrases with words, and sentences with phrases. By this, one can bypass the need to analyze sentences grammatically. For an example, formulaic languages may contain those kinds of sentences that usually appear in a phrasebook.

[2]More formulaic language means it contains larger number of polysemous words than that of less formulaic language.

Bibliography

Aha, D. W., D. Kibler, and M. K. Albert. Instance-based learning algorithms. *Machine Learning*, 6(1):37–66, 1991.

Bahl, L. R., F. Jelinek, R. L. Mercer. A Maximum Likelihood Approach to Continuous Speech Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-* 5(2):179–190.

Bangalore, S. and A. K. Joshi. Supertagging: An Approach to Almost Parsing. *Computational Linguistics*, 25(2):237–265.

Bangalore, S., P. Haffner, and S. Kanthak. Statistical Machine Translation through Global Lexical Selection and Sentence Reconstruction. In *Proceedings of the 45th Annual meeting of the Association for Computational Linguistics (ACL 2007)*, pages 152–159, Prague, Czech Republic, 2007.

Berger, A L., V. J. Della Pietra, and S. A. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71, 1996.

Blunsom, P. and T. Cohn. Discriminative word alignment with conditional random fields. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 65–72, Sydney, Australia, 2006.

Brown, P. F., J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. D. D. Lafferty, R. L. Mercer and P. S. Roossin. A statistical approach to language

translation. In *Proceedings of the 12th International Conference on Computational Linguistics (COLING-88)*, pages 71–76, Budapest, Hungary, 1988.

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

Brown, P. F., S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer. A statistical approach to sense disambiguation in machine translation. In *Proceedings of the Workshop on Speech and Natural Language, HLT 1991*, pages 146–151, Pacific Grove, CA, 1991.

Brown, P. F., S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer. Word-sense disambiguation using statistical methods. In *29th Annual Meeting of the Association for Computational Linguistics: Proceedings of the conference*, pages 264–270, University of California, Berkeley, CA, 1991.

Brown, P. F., S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311, 1993.

Brunning, J., A. Gispert, and W. Byrne. Context-dependent alignment models for statistical machine translation. In *NAACL HLT 2009: Proceedings of Human Language Technologies: the 2009 annual conference of the North American Chapter of the ACL*, pages 110–118, Boulder, CO, 2009.

Cabezas, C. and P. Resnik. Using WSD techniques for lexical selection in statistical machine translation (CS-TR-4736/LAMP-TR-124/UMIACS-TR-2005-42). *Technical Report*, Institute for Advanced Computer Studies, University of Maryland, College Park, MD, 2005.

Carl, M. and A. Way (eds.). *Recent Advances in Example-Based Machine Translation.* Kluwer Academic Publishers, Dordrecht, the Netherlands, 2003.

Carpuat, M. and D. Wu. Word sense disambiguation vs. statistical machine translation. In *43rd Annual meeting of the Association for Computational Linguistics (ACL 2005)*, pages 387–394, University of Michigan, Ann Arbor, MI, 2005.

Carpuat, M. and D. Wu. Evaluating the word sense disambiguation performance of statistical machine translation. In *Proceedings of the Second International Joint Conference on Natural Language Processing (IJCNLP 2005)*, pages 120–125, Jeju Island, Republic of Korea, 2005b.

Carpuat, M. and D. Wu. Improving Statistical Machine Translation Using Word Sense Disambiguation. In *EMNLP-CoNLL-2007: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 61–72, Prague, Czech Republic, 2007.

Carpuat, M., Y. Shen, X. Yu, and D. Wu. Toward integrating word sense and entity disambiguation into statistical machine translation. In *Third International Workshop on Spoken Language Translation (IWSLT 2006)*, pages 37–44, Kyoto, Japan, 2006.

Carreras, X. and L. Márquez. Introduction to the CoNLL-2004 Shared Task: Semantic Role Labeling. In *Proceedings of the CoNLL 2004 Shared Task*, pages 89–97, Boston, MA, 2004.

Chan, Y. S., H. T. Ng, and D. Chiang. Word sense disambiguation improves statistical machine translation. In *Proceedings of the 45th Annual meeting of the Association for Computational Linguistics (ACL 2007)*, pages 33–40, Prague, Czech Republic, 2007.

Chen, S. F., J. Goodman. An empirical study of smoothing techniques for language modeling. *Technical Report TR1098*, Center for Research in Computing Technology (Harvard University), 1998.

Chen, J. and K. Vijay-Shanker. Automated Extraction Of Tags From The Penn

Treebank. In *Proceedings of the Sixth International Workshop on Parsing Technology*, pages 65–76, Trento, Italy, 2000.

Chen, J., S. Bangalore , and K. Vijay-Shanker. Automated Extraction of Tree-Adjoining Grammars from Treebanks. *Natural Language Engineering*, 12(3):251–299, 2006.

Chiang, D. Hierarchical Phrase-based Translation. *Computational Linguistics*, 33(2):202–228, 2007.

Chiang, D. A hierarchical phrase-based model for statistical machine translation. In *ACL-2005: 43rd Annual meeting of the Association for Computational Linguistics*, pages 263–270, University of Michigan, Ann Arbor, 2005.

Chiang, D., K. Knight, and W. Wang. 11,001 new features for statistical machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT NAACL 2009)*, pages 218–226, Boulder, CO, 2009.

Chiang, D, Y. Marton, and P. Resnik. Online large-margin training of syntactic and structural translation features. In *EMNLP 2008: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 224–233, Honolulu, Hawaii, 2008.

Clark, S. and J. R. Curran. The Importance of Supertagging for Wide-Coverage CCG Parsing. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004)*, pages 282–288, Geneva, Switzerland, 2004.

Clark, S. and J. R. Curran. Wide-Coverage Efficient Statistical Parsing with CCG and Log-Linear Models. *Computational Linguistics*, 33(1):439–552, 2007.

Costa-Jussà, M. R. and R. E. Banchs. A vector-space dynamic feature for phrase-based statistical machine translation. *Journal of Intelligent Information Systems*, 2010.

Cowan, B., I. Kučerová, and M. Collins. A discriminative model for tree-to-tree translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, pages 232–241, Sydney, Australia, 2006.

Crammer, K., O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. Online passive-aggressive algorithms. Journal of Machine Learning Research,7:551–585, 2006.

Daelemans, W., A. van den Bosch, A. Weijters. IGTree: using trees for compression and classification in lazy learning algorithms. *Artificial Intelligence Review*, 11:407–423, 1997a.

Daelemans, W., A. van den Bosch, J. Zavrel. A feature-relevance heuristic for indexing and compressing large case bases. In M. Van Someren and G. Widmer (Eds.), *Poster papers of the Ninth European Conference on Machine Learning*, pages 29–39, Prague, Czech Republic, 1997b.

Daelemans, W., A. van den Bosch, J. Zavrel. Forgetting exceptions is harmful in language learning. *Machine Learning, Special issue on Natural Language Learning*,34:11–41..

Daelemans, W. and A. van den Bosch. *Memory-Based Language Processing*, Cambridge University Press, Cambridge, UK, 2005.

Dempster, A. P., N. M. Laird and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society*, 39(B):1-38, 1977.

Deng, Y. and W. Byrne. HMM word and phrase alignment for statistical machine translation. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 169–176, Vancouver, Canada, 2005.

Dice L. R. Measures of the amount of ecologic association between species. *Journal of Ecology*, 26:297-302, 1945.

Doddington, G. Automatic evaluation of language translation using n-gram cooccurrence statistics. In *HLT 2002:Human Language Technology Conference: proceedings of the second international conference on human language technology research*, pages 138–145, San Diego, CA, 2002.

Ekbal, A., S. K. Naskar and S. Bandyopadhyay. A Modified Joint Source Channel Model for Transliteration. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, pages 191–198, Sydney, Australia, 2006.

Foster, G, R. Kuhn, and H. Johnson. Phrasetable smoothing for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, pages 53–61, Sydney, Australia, 2006.

Fraser, A, D. Marcu. Getting the structure right for word alignment: LEAF. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 51–60, Prague, Czech Republic, 2007.

Fraser, A, D. Marcu. Measuring word alignment quality for statistical machine translation. *Computational Linguistics*, 33(3):293–303, 2007.

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

García-Varea, I., F. J. Och, H. Ney, and F. Casacuberta. Refined lexicon models for statistical machine translation using a maximum entropy approach. In *39th Annual meeting of the Association for Computational Linguistics and 10th Conference of the European Chapter of the Association for Computational Linguistics (ACL/EACL 2001)*, pages 204–211, Toulouse, France, 2001.

García-Varea, I., F. J. Och, H. Ney, and F. Casacuberta. Improving alignment quality in statistical machine translation using context-dependent maximum entropy models. In *Proceedings of The 19th International Conference on Computational Linguistics (Coling 2002)*, pages 1051–1054, Taipei, Taiwan, 2002.

Germann, U. Greedy decoding for statistical machine translation in almost linear time. In *HLT-NAACL: Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 72–79, Edmonton, AL, Canada, 2003.

Giménez, J. and L. Màrquez. Context-aware discriminative phrase selection for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation, ACL 2007*, pages 159–166, Prague, Czech Republic, 2007.

Giménez, J. and L. Màrquez. Discriminative Phrase Selection for Statistical Machine Translation. In C. Goutte, N. Cancedda, M. Dymetman and G. Foster (eds.), *Learning Machine Translation*. NIPS Workshop Series. MIT Press, 2009.

Gimpel, K. and N. A. Smith. Using Grammatical Dependencies for Contextual Phrase Translation Disambiguation. In *Proceedings of the Third Workshop on Statistical Machine Translation, ACL-08:HLT*, pages 9–17, Columbus, OH, 2008.

Gimpel, K. and N. A. Smith. Feature-Rich Translation by Quasi-Syncronous Lattice Parsing. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*, pages 219–228, Singapore.

Habash, N. Four techniques for online handling of out-of-vocabulary words in Arabic-English statistical machine translation. In *ACL-08: HLT. 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Short papers*, pages 57–60, The Ohio State University, Columbus, OH, 2008.

Haque, R., S. K. Naskar, Y. Ma, and A. Way. Using Supertags as Source Language Context in SMT. In *EAMT-2009: Proceedings of the 13th Annual Conference*

of the European Association for Machine Translation, pages 234–241, Barcelona, Spain, 2009a.

Haque, R., S. K. Naskar, A. van den Bosch, and A. Way. Dependency Relations as Source Context in Phrase-Based SMT. In *Proceedings of PACLIC 23: the 23rd Pacific Asia Conference on Language, Information and Computation*, pages 170–179, Hong Kong, China, 2009b.

Haque, R., S. Dandapat, A. Srivastava, S. K. Naskar, and A. Way. English–Hindi Transliteration Using Context-Informed PB-SMT: the DCU System for NEWS 2009. In *Proceedings of Named Entities Workshop 2009, ACL-IJCNLP 2009*, pages 104–107, Singapore, 2009c.

Haque, R., S. K. Naskar, A. van den Bosch, and A. Way. 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., 2010a.

Haque, R., S. K. Naskar, A. Way, M. R. Costa-jussà, and R. E. Banchs. Sentence Similarity-Based Source Context Modelling in PBSMT. In *Proceedings of the International Conference on Asian Language Processing 2010*, pages 257–260, Harbin, China, 2010b.

Haque, R., S. K. Naskar, A. van den Bosch, and A. Way. Integrating Source-Language Context into Phrase-Based Statistical Machine Translation. *Machine Translation (in press)*, 2011.

Hasan, S., J. Ganitkevitch, H. Ney, and J. Andrés-Ferrer. Triplet lexicon models for statistical machine translation. In *EMNLP 2008: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 372–381, Honolulu, HI, 2008.

Hassan, H, M. Hearne, K. Simáan, and A. Way. Syntactic Phrase-Based Statistical

Machine Translation. In *Proceedings of the IEEE 2006 Workshop on Spoken Language Translation*, Palm Beach, Aruba.

Hassan, H, K. Simáan, and A. Way. Supertagged phrase-based statistical machine translation. In *ACL 2007: proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 288–195, Prague, Czech Republic, 2007.

Hassan, H, K. Simáan, and A. Way. Syntactically Lexicalized Phrase-Based SMT. *IEEE Transactions on Audio, Speech and Language Processing*, 16(7):1260–1273, 2008.

Hassan, H. *Lexical Syntax for Statistical Machine Translation.* Ph.D Thesis, Dublin City University, Dublin, Ireland.

He, Z., L. Liu, and S. Lin. Improving statistical machine translation using lexicalized rule selection. In *Proceedings of the 24th International Conference on Computational Linguistics (Coling 2008)*, pages 321–328, Manchester, UK, 2008.

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

Hockenmaier, J. and M. Steedman. Acquiring Compact Lexicalized Grammars from a Cleaner Treebank. In *Proceedings of Third International Conference on Language Resources and Evaluation (LREC 2002)*, pages 1974–1981, Las Palmas, 2002.

Huang, L. and D. Chiang. Forest rescoring: faster decoding with integrated language models. In *ACL 2007: proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 144–151, Prague, Czech Republic, 2007.

Hutchins W. J. and Somers H. L. *An Introduction to Machine Translation.* Academic Press. 1992.

Ittycheriah, A. and S. Roukos. Direct translation model 2. In *NAACL-HLT-2007 Human Language Technology: the conference of the North American Chapter of the Association for Computational Linguistics*, pages 57–64, Rochester, NY, 2007.

Johansson, R. and P. Nugues. Dependency-based Syntactic-Semantic Analysis with PropBank and NomBank. In *Proceedings of the CoNLL-2008 Shared Task*, pages 183–187, Manchester, UK, 2008.

Joshi, A. K. and Y. Schabes. Tree Adjoining Grammars and Lexicalized Grammars. In M. Nivat and A. Podelski (eds.), *Tree Automata and Languages*, pages 409–431, Amsterdam, The Netherlands: North-Holland, 1992.

Kang, B. and K. Choi. Automatic transliteration and back-transliteration by decision tree learning. In *LREC-2000: Second International Conference on Language Resources and Evaluation*, pages 1135–1141, Athens, Greece, 2000.

Knight, K. Decoding Complexity in Word-Replacement Translation Models. *Computational Linguistics*, 25(4):607–615, 1999.

Knight, K and J. Graehl. Machine Transliteration. *Computational Linguistics*, 24(4):559–612, 1998.

Koehn, P. Pharaoh: A Beam Search Decoder for Phrase-Based Statistical Machine Translation Models. In Robert E. Frederking and Kathryn B. Taylor (Eds.), *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, 2004a.

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

Koehn, P. Europarl: A parallel corpus for statistical machine translation. In *MT*

summit X, the tenth machine translation summit, pages 79–86, Phuket, Thailand, 2005.

Koehn, P. *Statistical Machine Translation*. Cambridge University Press, Cambridge, UK., 2009.

Koehn, P, F. J. Och, and D. Marcu. 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, AB, 2003.

Koehn, P., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, E. Herbst. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the demo and poster sessions, ACL 2007*, pages 177–180, Prague, Czech Republic, 2007.

Kumar, S. and W. Byrne. Minimum Bayes-risk decoding for statistical machine translation. In *HLT-NAACL 2004: Human Language Technology conference and North American Chapter of the Association for Computational Linguistics annual meeting*, pages 169–176, Boston, USA, 2003.

Kumaran, A. and T. Kellner. A generic framework for machine transliteration. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 2007.

Langlais, P. and A. Patry. Translating unknown words using analogical learning. In *EMNLP-CoNLL-2007: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Prague, Czech Republic, pages 877–886, 2007.

Langlais, P., A. Patry, and F. Gotti. A greedy decoder for phrase-based statistical machine translation. In *TMI-2007: Proceedings of the 11th International Confer-*

ence on Theoretical and Methodological Issues in Machine Translation, Skövde, Sweden, pages 104–113, 2007.

Lavie, A. and A. Agarwal. 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, ACL 2007*, pages 228–231, Prague, Czech Republic, 2007.

Lepage, Y. and E. Denoual. Objective evaluation of the analogy-based machine translation system ALEPH. In *Proceedings of the 12th Annual Meeting of the Association of Natural Language Processing*, pages 873–876, 2006.

Levenshtein, V. Binary codes capable of correcting deletions, correcting deletions, insertions, and reverals. *Soviet Physics Doklady*, 10:707–710.

Li, H., M. Zhang, S. Jian. A joint source-channel model for machine transliteration. In *ACL 2004: 42nd annual meeting of the Association for Computational Linguistics: Proceedings of the conference*, pages 159–166, Barcelona, Spain, 2004.

Li, H., A. Kumaran, M. Zhang, V. Pervouchine. Whitepaper of NEWS 2009 machine transliteration shared task. In *NEWS'09: Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration, ACL-IJCNLP'09*, Singapore, 2009.

Liang, P., A. Bouchard-Côté, D. Klein, and B. Taskar. An end-to-end discriminative approach to machine translation. In *Coling-ACL 2006: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 761–768, Sydney, Australia, 2006.

Ma, Y., S. Ozdowska, Y. Sun, and A. Way. Improving word alignment using syntactic dependencies. In *Proceedings of the ACL-08: HLT Second Workshop on Syntax and Structure in Statistical Translation (SSST-2)*, pages 69–77, Columbus, OH.

Magerman, D. Statistical decision-tree models for parsing. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics.*, pages 276–283, Morristown, NJ, USA.

Matthews, D. *Machine Transliteration of Proper Names.* Master's Thesis, University of Ed-inburgh, Edinburgh, United Kingdom.

Marcu, D. and W. Wong. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-02)*, pages 133–139, Philadelphia, PA, 2009.

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

Marton, Y. and P. Resnik. Soft syntactic constraints for hierarchical phrased-based translation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*, pages 1003–1011, The Ohio State University, Columbus, OH, 2008.

Mauser, A., S. Hasan, and H. Ney. Extending Statistical Machine Translation with Discriminative and Trigger-Based Lexicon Models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, (EMNLP 2009)*, pages 210–218, Singapore, 2009.

Max, A., R. Makhloufi, and P. Langlais. Explorations in using grammatical dependencies for contextual phrase translation disambiguation. In *EAMT 2008: 12th annual conference of the European Association for Machine Translation*, pages 114–119, Hamburg, Germany, 2008.

Moore, R. A discriminative framework for bilingual word alignment. In *HLT-EMNLP-2005: Proceedings of Human Technology Conference and Conference on*

Empirical Methods in Natural Language Processing, pages 81–88, Vancouver, BC, Canada, 2005.

Moore, R. and C. Quirk. Random restarts in minimum error rate training for statistical machine translation. In *Coling 2008: 22nd International Conference on Computational Linguistics, Proceedings of the conference*, pages 585–592, Manchester UK, 2008.

Nagao, M. A framework of a mechanical translation between Japanese and English by analogy principle. In A. Elithorn and R. Banerji (Eds.),*Artificial and human intelligence*, Amsterdam, North-Holland, pages 173–180, 1984.

Nivre, J. Dependency grammar and dependency parsing. Technical Report MSI report 05133, Växjö University: School of Mathematics and Systems Engineerin, 2005.

Nivre, J., J. Hall, and J. Nilsson. MaltParser: A data-driven parser generator for dependency parsing. In *LREC 2006: Proceedings of the fifth international conference on Language Resources and Evaluation*, pages 2216–2219, Genoa, Italy, 2006.

Och, F. J. Minimum Error Rate Training in Statistical Machine Translation. In *41st Annual Meeting of the Association for Computational Linguistics (ACL 2003)*, pages 160–167, Sapporo, Japan, 2003.

Och, F. J. and H. Ney. A comparison of alignment models for statistical machine translation. In *the 18th International Conference on Computational Linguistics (Colin 2000)*, pages 1086–1090, Universität des Saarlandes, Saarbrücken, Germany, 2000a.

Och, F. J. and H. Ney. Improved statistical alignment models. In *ACL-2000: 38th Annual meeting of the Association for Computational Linguistics*, pages 440–447, Hong Kong, 2000b.

Och, F. J. and H. Ney. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In *40th Annual Meeting of the Association for Computational Linguistics (ACL 2002)*, pages 295–302, Philadelphia, PA, 2002.

Och, F. J. and H. Ney. A systematic comparison of various statistical alignment models. In *Computational Linguistics*, 29(1):19–51, 2003.

Och, F. J. and H. Ney. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449, 2004.

Och, F. J., C. Tillmann, and H. Ney. Improved Alignment Models for Statistical Machine Translation. In *Proceedings of the Joint SIGDAT Conf. on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28, University of Maryland, College Park, MD, 1999.

Och, F. J., Ueffing, N., and Ney, H. An efficient A* search algorithm for statistical machine translation. In *ACL-EACL 2001 Workshop on Data-Driven Machine Translation*, 55–62, 2001.

Och, F. J., D. Gildea, S. Khudanpur, A. Sarkar, K. Yamada, A. Fraser, S. Kumar, L. Shen, D. Smith, K. Eng, V. Jain, Z. Jin, and D. Radev. A smorgasbrod of features for statistical machine translation. In *HLT-NAACL 2004: Human Language Technology conference and North American Chapter of the Association for Computational Linguistics annual meeting*, 161–168, The Park Plaza Hotel, Boston, 2004.

Okita, S., J. Jiang, R. Haque, H. Al-Maghout, J. Du, S. K. Naskar, and A. Way. MaTrEx: the DCU MT System for NTCIR-8. In *Proceedings of NTCIR-8*, pages 377–383, Tokyo, Japan, 2010.

Papineni, K., S. Roukos, and W. Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. In *40th Annual Meeting of the Association for Computational Linguistics (ACL 2002)*, pages 311–318, Philadelphia, PA, 2002.

Patry, A. and P. Langlais. Prediction of words in statistical machine translation using a multilayer perceptron. In *MT Summit XII: proceedings of the twelfth Machine Translation Summit*, pages 101–111, Ottawa, ON, Canada, 2009.

Penkale, S., R. Haque, S. Dandapat, P. Banerjee, A. K. Srivastava, J. Du, P. Pecina, S. K. Naskar, M. L. Forcada, and A. Way. MATREX: The DCU MT System for WMT 2010. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR (WMT-MetricsMATR 2010), ACL 2010*, pages 143–148, Uppsala, Sweden, 2010.

Quirk, C, A. Menezes, and C. Cherry. Dependency Treelet Translation: Syntactically Informed Phrasal SMT. In *ACL-2005: 43rd Annual meeting of the Association for Computational Linguistics*, pages 271–279, Ann Arbor, MI, 2005.

Shen, L., B. Zhang, S. Matsoukas, and R. Weischedel. Effective use of linguistic and contextual information for statistical machine translation. In *EMNLP-2009: proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 72–80, Singapore, 2009.

Shen, L., J. Xu, and R. Weischedel. A new string-to-dependency machine translation algorithm with a target dependency language model. In *Proceedings of ACL-08: HLT: the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 577–585, Columbus, OH, 2008.

Snover, M., B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. A study of translation edit rate with targeted human annotation. In *AMTA 2006: Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, pages 223–231, Cambridge, MA, 2006.

Specia, L., B. Sankaran, and M. G. V. Nunes n-Best Reranking for the Efficient Integration of Word Sense Disambiguation and Statistical Machine Translation. In *Proceedings of International Conference on Intelligent Text Processing and Computational Linguistics (CICLING 2008)*, pages 399–410, Haifa, Israel, 2008.

Somers, H. Review article: Example-based Machine Translation. *Machine Translation*, 14:113–157, 1999.

Somers, H., I. McLean, and D. Jones. Experiments in multilingual example-based generation. In *Proceedings of the 3rd Conference on the Cognitive Science of Natural Language Processing (CSNLP 1994)*, Dublin, Ireland, 1994.

Steedman, M. *The Syntactic Process*. MIT Press: Cambridge, MA, 2000.

Stroppa, N., A. van den Bosch, and A. Way. 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 2007)*, pages 231–240, Skövde, Sweden, 2007.

Surdeanu, M., R. Johansson, A. Meyers, L. Màrquez, and J. Nivre. The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the 12th Conference on Computational Natural Language Learning (CoNLL-2008)*, pages 159–177, Manchester, UK, 2008.

Taskar, B., S. Lacoste-Julien, and D. Klein. A discriminative matching approach to word alignment. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics*, pages 73–80, Vancouver, BC, Canada, 2005.

Tiedemann, J. and L. Nygaard. The OPUS corpus - parallel & free. In *Proceedings of the 4th International Conference on language resources and evaluation (LREC 2004)*, pages 1183–1186, Lisbon, Portugal, 2004.

Tillmann, C., H. Ney, A. Zubiaga. A DP based search using monotone alignments in statistical translation. In *ACL-EACL-1997: 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics: Proceedings of the Conference*, pages 289–296, Madrid, Spain, 1997.

Tillmann, C. A unigram orientation model for statistical machine translation. In *Proceedings Human-Language Technology and North American Association of Computational Linguistics (HLT-NAACL)*, pages 101–104, Boston, MA, 2004.

Tillmann, C, T. Zhang. A discriminative global training algorithm for statistical mt. In *Coling-ACL 2006: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 721–728, Sydney, Australia, 2006.

van den Bosch, A. Wrapped progressive sampling search for optimizing learning algorithm parameters. In R. Verbrugge, N. Taatgen, and L. Schomaker (Eds.), *Proceedings of the 16th Belgian-Dutch Conference on Artificial Intelligence*, Groningen, The Netherlands, 2004.

van den Bosch, A., B. Busser, S. Canisius, and W. Daelemans. An efficient memorybased morpho-syntactic tagger and parser for Dutch. In *Proceedings of Computational Linguistics in the Netherlands: Selected Papers from the Seventeenth CLIN Meeting*, pages 99–114, Leuven, Belgium, 2007.

Vauquois, B and C. Boitet. Automated Translation at Grenoble University. *Computational Linguistics*, 11(1):2836, 1985.

Venkatapathy, S. NLP Tools Contest - 2008: Summary. In *Proceedings of the NLP Tools Contest, ICON 2008*, Pune, India, 2008.

Venkatapathy, S. and S. Bangalore. Three models for discriminative machine translation using global lexical selection and sentence reconstruction. In *SSST, NAACL-HLT-2007 AMTA Workshop on Syntax and Structure in Statistical Translation*, pages 96–102, Rochester, NY, 2007.

Vickrey, D., L. Biewald, M. Teyssier and D. Koller. Word-sense disambiguation for machine translation. In *HLT-EMNLP-2005: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 771–778, Vancouver, BC, Canada, 2005.

Vogel, S., H. Ney, and C. Tillmann. HMM-based word alignment in statistical translation. In *Proceedings of the 16th International Conference on Computational Linguistics*, pages 836–841, Copenhagen, Denmark.

Watanabe, T., J. Suzuki, H. Tsukada, H. Isozaki. Online large-margin training for statistical machine translation. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pages 764–773, Prague, Czech Republic.

Wang, Z. and J. Shawe-Taylor. Kernel regression based machine translation. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers,*, pages 185–188, Rochester, New York, 2008.

Weaver, W. Translation. In Locke, W. and Booth, A. (eds.), *Machine translation of languages: fourteen essays, MIT Press, Cambridge, MA*, pages 15–23, 1955.

Wellington, B., J. Turian, C. Pike, and I.D. Melamed. Scalable purely-discriminative training for word and tree transducers. In *AMTA 2006: Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, Visions for the Future of Machine Translation*, pages 251–260, Cambridge, Massachusetts, 2006.

Wu, D. and P. Fung. Can semantic role labeling improve SMT? In *EAMT-2009: Proceedings of the 13th Annual Conference of the European Association for Machine Translation*, pages 218–225, Barcelona, Spain, 2009.

Yamada, K. and K. Knight. A decoder for syntax-based statistical MT. In *ACL-2002: 40th Annual meeting of the Association for Computational Linguistics*, pages 303–310, Philadelphia, 2002.

Xiong, D., M. Zhang, and H. Li. Learning Translation Boundaries for Phrase-Based Decoding. In *NAACL-HLT-2010: Human Language Technologies: the 2010 an-*

nual conference of the North American Chapter of the Association for Computational Linguistics, pages 136–144, Los Angeles, CA, 2010.

Zens, R. and H. Ney. Improvements in phrase-based statistical machine translation. In *HLT-NAACL 2004: Human Language Technology conference and North American Chapter of the Association for Computational Linguistics annual meeting*, pages 257–264, Boston, MA, 2004.

Zens, R., F.J. Och, and H. Ney. Phrase-Based Statistical Machine Translation. In M. Jarke, J. Koehler, G. Lakemeyer (Eds.),*KI - 2002: Advances in Artificial Intelligence. 25. Annual German Conference on AI, KI 2002*, Vol. LNAI 2479, pages 18–32, Springer Verlag, 2002.

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