Using Supertags as Source Language Context in SMT

Rejwanul Haque[†], **Sudip Kumar Naskar**[†], **Yanjun Ma**[*] and **Andy Way**[†*]

CNGL[†]/NCLT[*]

School of Computing

Dublin City University

Dublin 9, Ireland

`{rhaque, snaskar, yma, away}@computing.dcu.ie`

Abstract

Recent research has shown that Phrase-Based Statistical Machine Translation (PB-SMT) systems can benefit from two enhancements: (i) using words and POS tags as context-informed features on the source side; and (ii) incorporating lexical syntactic descriptions in the form of supertags on the target side. In this work we present a novel PB-SMT model that combines these two aspects by using supertags as source language context-informed features. These features enable us to exploit source similarity in addition to target similarity, as modelled by the language model. In our experiments two kinds of supertags are employed: those from Lexicalized Tree-Adjoining Grammar and Combinatory Categorial Grammar. We use a memory-based classification framework that enables the estimation of these features while avoiding problems of sparseness. Despite the differences between these two approaches, the supertaggers give similar improvements. We evaluate the performance of our approach on an English-to-Chinese translation task using a state-of-the-art phrase-based SMT system, and report an improvement of 7.88% BLEU score in translation quality when adding supertags as context-informed features.

1 Introduction

In log-linear phrase-based SMT, the probability $P(e_1^I|f_1^J)$ of a target phrase e_1^I given a source phrase f_1^J is modelled as a log-linear combination of features which normally consist of a finite set of translational features, and a language model (Och and Ney, 2002). The usual translational features involved in those models express dependencies between the source and target phrases, but not dependencies between the phrases in the source language themselves. Stroppa et al. (2007) were the first to show that incorporating source language context using neighbouring words and part-of-speech tags had the potential to improve translation quality.

In a separate strand of research, Hassan et al. (2006, 2007, 2008) 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 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 that supertaggers exist only for English.

In this paper, we begin to explore whether such a system could indeed generate improvements across all PB-SMT system components. Our novel approach combines the methods of (Stroppa et al., 2007) and (Hassan et al., 2006, 2007, 2008; Hassan, 2009) in one model. We extend a standard PB-SMT system with syntactic descriptions on the source side. Crucially, the kind of lexical descriptions that we employ are those that are commonly devised within lexicon-driven approaches to linguistic syntax, namely Lexicalized Tree-Adjoining Grammar (LTAG: Joshi and Schabes, 1992; Bangalore and Joshi, 1999) and Combinatory Categorial Grammar (CCG: Steedman, 2000). In such approaches, the grammar consists of a very rich lexicon and a small set of combinatory operators that assemble lexical entries together into parse-trees. The lexical entries consist of syntactic constructs ('supertags') that describe information such as the POS tag of the word, its subcategorisation information and the hierarchy of phrase categories

that the word projects upwards. Like (Hassan et al., 2006, 2007, 2008; Hassan, 2009), in this work we employ the lexical entries but exchange the algebraic combinatory operators with the more robust and efficient supertagging approach: like standard taggers, supertaggers employ probabilities based on local context and can be implemented using finite state technology, e.g. Hidden Markov Models (Bangalore and Joshi, 1999).

There are currently two supertagging approaches available: LTAG-based (Bangalore and Joshi, 1999) and CCG-based (Clark and Curran, 2004). Both the LTAG (Chen et al., 2006) and the CCG supertag sets (Hockenmaier, 2003) were acquired from the WSJ section of the Penn-II Treebank using hand-built extraction rules. Here we test both the LTAG and CCG supertaggers. We extract the supertagged components of context words ($\pm 1/\pm 2$) along with the source phrase (Koehn et al., 2003) in a standard PB-SMT system. We use a memory-based classification approach to obtain the probability for the given additional contexts with the source phrase. In this paper we discuss these and other empirical issues.

The remainder of the paper is organized as follows. In section 2 we discuss related work. Section 3 gives a brief overview of PBSMT. In section 4 we describe the context-informed features contained in our baseline log-linear phrase-based SMT system. In section 5 we describe the memory-based classification approach. Section 6 describes the features used in the experiments, and the pre-processing required. Section 7 includes the results obtained, together with some analysis. Section 8 concludes, and provides avenues for further work.

2 Related Work

(Berger et al., 1996) first suggested context-sensitive modelling of word translations in order to integrate local contextual information into their IBM translation models using a Maximum Entropy (MaxEnt) model, but the work is not supported by any significant evaluation results.

García Varea et al. (2001) present a MaxEnt approach to integrate contextual dependencies into the EM algorithm of the statistical alignment model to develop a refined context-dependent lexicon model. Using such a model on the German—English Verbmobil corpus, they obtained better alignment quality in terms of improved alignment error rate (AER). However, since alignment is not an end task in itself and most often used as an intermediate task to generate phrase pairs for the t-tables in PB-SMT systems, improved AER scores do not necessarily result in improved translation quality, as noted by a number of researchers.

(Vickrey et al., 2005) built classifiers inspired by those used in word-sense disambiguation (WSD) to fill in any blanks in a partially completed translation. (Giménez and Màrquez, 2007) extended this work by considering the slightly more general case of very frequent phrases and moved to full translation rather than blank-filling on the target side.

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). However, more recent approaches (Carpuat and Wu, 2007; Chan et al., 2007; Giménez and Màrquez, 2007) of integrating state-of-the-art WSD methods into SMT to improve the overall translation quality have met with more success.

Language models arguably play the most significant role in today's PB-SMT systems. It is obvious that a straightforward addition of a source language model will make no contribution as this will be cancelled out by the denominator in the noisy-channel model of SMT. However, for some time now the feeling was that some incorporation of source language information into SMT systems had to help. (Stroppa et al., 2007) added source-side contextual features to a state-of-the-art log-linear PB-SMT system by incorporating context-dependent phrasal translation probabilities learned using decision trees. They considered up to two words and/or POS tags on either side of the source focus word as contextual features. In order to overcome problems of estimation of such features, they used a decision-tree classifier which implicitly smoothes the probability estimates. Significant improvements over a baseline state-of-the-art PB-SMT system were obtained on Italian—English and Chinese—English IWSLT tasks.

Unlike other recent proposals to exploit the accuracy and the flexibility of discriminative learning (e.g. Cowan et al., 2006; Liang et al., 2006), the strength of the approach of (Stroppa et al., 2007) is that no redefinition of one's training procedures is required.

Like the work of (Max et al., 2008), the present work is directly motivated by and an extension of the approach of (Stroppa et al., 2007).

The work of both (Max et al., 2008) and (Gimpel and Smith, 2008) focus on language pairs where the target is not English. While (Gimpel and Smith, 2008) are unable to show any improvements for English→German, (Max et al., 2008) conduct experiments from English→French. Using the same sorts of local contextual features as (Stroppa et al., 2007), as well as using broader context in addition to grammatical dependency information, (Max et al., 2008) show modest gains over a PB-SMT baseline model in terms of automatic evaluation scores, but more improvements come to light in a manual investigation.

One final paper in this strand of research is that of (He et al., 2008), who despite not mentioning the obvious link between the two pieces of work, show that the source language features used by (Stroppa et al., 2007) are also of benefit when used with the Hiero (Chiang, 2007) decoder.

As regards supertagged models of translation, (Hassan et al., 2006, 2007b, 2008; Hassan, 2009) have demonstrated clearly that adding supertags (essentially, part-of-speech tags of words plus local subcategorisation requirements) in the target language model and on the target side of the translation model improve state-of-the-art PB-SMT systems. The system of (Hassan et al., 2007a) was ranked first according to human evaluators on the IWSLT 2007 Arabic–English task, despite the improvements in system design not being shown to their best advantage by the automatic evaluation metrics. More recently, (Hassan, 2009) has demonstrated that improvements can even be gained over the leading NIST-07 Arabic–English system of (Ittycheriah and Roukos, 2007).

3 Log-Linear PB-SMT

Translation is modelled in PB-SMT as a decision process, in which the translation $e_1^I = e_1 \ldots e_I$ of a source sentence $f_1^J = f_1 \ldots f_J$ is chosen to maximize (1):

$$\underset{I,e_1^I}{\arg\max}\, P(e_1^I \mid f_1^J) = \underset{I,e_1^I}{\arg\max}\, P(f_1^J \mid e_1^I).P(e_1^I) \quad (1)$$

where $P(f_1^J \mid e_1^I)$ and $P(e_1^I)$ denote respectively the translation model and the target language model (Brown et al., 1993). In log-linear phrase-based SMT, the posterior probability $P(e_1^I \mid f_1^J)$ is directly modelled as a (log-linear)

combination of features (Och and Ney, 2002), that usually comprise M translational features, and the language model, as in (2):

$$\log P(e_1^I \mid f_1^J) = \sum_{m=1}^{m} \lambda_m h_m(f_1^J, e_1^I, s_1^K)$$
$$+ \lambda_{LM} \log P(e_1^I) \quad (2)$$

where $s_1^k = s_1 \ldots s_k$ denotes a segmentation of the source and target sentences respectively into the sequences of phrases $(\hat{e}_1, \ldots, \hat{e}_k)$ and $(\hat{f}_1, \ldots, \hat{f}_k)$ such that (we set $i_0 = 0$) (3):

$$\forall 1 \le k \le K, \; s_k = (i_k; b_k, j_k),$$
$$\hat{e}_k = e_{i_{k-1}+1} \ldots e_{i_k},$$
$$\hat{f}_k = f_{b_k} \ldots f_{j_k} \quad (3)$$

The translational features involved depend only on a pair of source/target phrases and do not take into account any context of these phrases. This means that each feature h_m in (2) can be rewritten as in (4):

$$h_m(f_1^J, e_1^I, s_1^K) = \sum_{k=1}^{K} \hat{h}_m(\hat{f}_k, \hat{e}_k, s_k) \quad (4)$$

where $\hat{h}_m$ is a feature that applies to a single phrase-pair. It thus follows:

$$\sum_{m=1}^{m} \lambda_m \sum_{k=1}^{K} \hat{h}_m(\hat{f}_k, \hat{e}_k, s_k) = \sum_{k=1}^{K} \hat{h}(\hat{f}_k, \hat{e}_k, s_k) \quad (5)$$

where, $\hat{h} = \sum_{m=1}^{m} \lambda_m \hat{h}_m$. In this context, the translation process amounts to: (i) choosing a segmentation of the source sentence, (ii) translating each source phrase, and (iii) re-ordering the target segments obtained.

4 Source Context Features in Log-Linear PB-SMT

As well as using local words and POS-tags as features, as in (Stroppa et al., 2007), we introduce supertags as a syntactic source context feature in the log-linear model of PB-SMT. The context of a source phrase $\hat{f}_k$ is defined as the sequence before and after a focus phrase $\hat{f}_k = f_{i_k} \ldots f_{j_k}$. In the following sections we describe both the lexical and syntactic features used.

4.1 Lexical Context Features

These features include the direct left and right context words of length l (resp. $f_{i_k-1} \ldots f_{i_k-l}$

and $f_{j_k+1}...f_{j_k+l}$) of a given focus phrase $\hat{f}_k = f_{i_k}...f_{j_k}$. It forms a window of size $2l+1$ features including the focus phrase. Thus lexical contextual information (CI) can be described as in (6):

$$CI = \{f_{i_k-l}...f_{i_k-1}, \hat{f}_k, f_{j_k+1}...f_{j_k+l}\} \qquad (6)$$

In our experiments we used ±1 and ±2 context words (i.e. $l=1, 2$).

4.2 Syntactic Context Features

We considered the syntactic information (SI) of the focus phrase and of the context words. The syntactic information we use are supertags and/or POS tags. In our model, the supertag or POS tag of a multi-word focus phrase is the concatenation of the supertags or POS tags of the words composing that phrase. We can thus describe our syntactic contextual information as in (7):

$$CI = \{SI(f_{i_k-l})...SI(f_{i_k-1}), \hat{f}_k, SI(\hat{f}_k),$$
$$SI(f_{j_k+1})...SI(f_{j_k+l})\} \qquad (7)$$

Thus a window of size $2l+2$ features is formed including the focus phrase and syntactic information of that phrase. In our experiments we used ±1 and ±2 syntactic information (i.e. $l=1, 2$). We also experimented with both supertag and POS tag features to see whether further improvements could be found. In such cases the contextual information is formed by the union of the two syntactic features, i.e. CI= $CI_{syn1} \cup CI_{syn2}$. We can also combine the syntax and the lexical contextual information in a similar way, if required.

One natural way of expressing a context-informed feature is as the conditional probability of the target phrase given the source phrase and its context information, as in (8):

$$h_m(f_k, CI(\hat{f}_k), \hat{e}_k, s_k) = \log P(\hat{e}_k / \hat{f}_k, CI(\hat{f}_k)) \quad (8)$$

5 Memory-Based Classification

As (Stroppa et al., 2007) point out, directly estimating $P(\hat{e}_k / \hat{f}_k, CI(\hat{f}_k))$ using relative frequencies (say) is problematic. Indeed, Zens and Ney (2004) showed that the estimation of $P(\hat{e}_k / \hat{f}_k)$ using relative frequencies results in the overestimation of the probabilities of long phrases, so 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, since the context is also

taken into account, this estimation problem can only become worse.

To avoid such problems, in this work we use three memory-based classifiers: IG-Tree, IB1 and TRIBL [1] (Daelemans et al., 2007). When predicting a target phrase given a source phrase and its context, the source phrase is intuitively the feature with the highest prediction power; in all our experiments, it is the feature with the highest information gain (IG).

In order to build the set of examples required to train the classifier, we modify the standard phrase-extraction method of (Koehn et al., 2003) 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.

We refer the interested reader to (Stroppa et al., 2007) for more details of how Memory-Based Learning (MBL) is used for classification of source examples for use in the log-linear MT framework.

6 Experimental Set-Up

6.1 Features Used

The distribution of target phrases given a source phrase and its contextual information is normalised to estimate $P(\hat{e}_k / \hat{f}_k, CI(\hat{f}_k))$. Therefore our expected feature is derived as in (9):

$$\hat{h}_{mbl} = \log P(\hat{e}_k / \hat{f}_k, CI(\hat{f}_k)) \qquad (9)$$

In addition to the above feature, we derived two more features $\hat{h}_{mod}$ and $\hat{h}_{best}$ from the posterior probability $P(\hat{e}_k / \hat{f}_k)$ and $P(\hat{e}_k / \hat{f}_k, CI(\hat{f}_k))$. The feature $\hat{h}_{mod}$ is defined as in (10):

$$\hat{h}_{mod} = \log [\alpha P(\hat{e}_k / \hat{f}_k, CI(\hat{f}_k))$$
$$+ (1-\alpha) P(\hat{e}_k / \hat{f}_k)] \quad (10)$$

The interpolation weight α was tuned manually on the devset.

We observed that MBL assigned large weights to more appropriate target phrases rather than less appropriate ones. One interesting observation is that IGTree seems to produce better results on lower α values, while in the case of IB1 and TRIBL, we obtained more mixed results.

---

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

While the best scores for IB1 and TRIBL were produced at both end of the spectrum, they performed best on higher values of α. Combining these weights, we derived $\hat{h}_{mod}$. Our final feature $\hat{h}_{best}$ is defined as in (11):

$$\hat{h}_{best} = \begin{cases} 1 \text{ if } \hat{e}_k \text{ maximizes} \\ \qquad P(\hat{e}_k / \hat{f}_k, CI(\hat{f}_k)) \quad (11) \\ 0 \text{ otherwise,} \end{cases}$$

We performed three different experiments by integrating these three features $\hat{h}_{mbl}$, $\hat{h}_{mod}$ and $\hat{h}_{best}$ directly into the log-linear model. In the first experiment E1, the baseline feature log $P(\hat{e}_k / \hat{f}_k)$ is directly replaced by $\hat{h}_{mod}$. In the second experiment (E2), we integrated the $\hat{h}_{mbl}$ feature together with the baseline features, keeping all the features unaffected. In the third experiment (E3), both the features $\hat{h}_{mbl}$ and $\hat{h}_{best}$ are integrated into the model in the same manner. As for the standard phrase-based approach, their weights are optimized using minimum-error-rate training (Och, 2003) for each of the experiments we carried out.

6.2   Pre-Processing

As (Stroppa et al., 2007) point out, PB-SMT decoders such as Pharaoh (Koehn, 2004) or Moses (Koehn, 2007) rely on a static phrase-table represented as a list of aligned phrases accompanied with several features. Since these features do not express the context 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 take into account the context-informed features for use with such decoders, the devset and test set that need to be translated is pre-processed. Each word appearing in the test set and devset is assigned a unique id. First we prepare the phrase table using the training data. Then we generate all possible phrases from the development set and test set. These devset and test set phrases are then searched for in the phrase table, and if found, then the phrase along with its contextual information is given to MBL for classification. MBL produces class distributions according to the maximum-match of the features contained in the source phrase. We derive new scores from this class distribution and merge them with the initial information of phrase table to take into account our feature functions ($\hat{h}_{mbl}$, $\hat{h}_{mod}$ and $\hat{h}_{best}$) in the log-linear model.

In this way we create a dynamic phrase table containing both the standard and the context-informed features. The new phrase table contains the source phrase (represented by the sequence of ids), target phrase and the new score (which varies depending on which experiments (E1, E2 and E3) are being carried out).

A lexicalized re-ordering model was used for all the experiments undertaken. The source phrase in the reordering table is replaced by the sequence of unique ids when the new phrase table is created. By replacing all the words by their ids in the development set, we perform MERT using our new phrase table to optimize the feature weights. In a similar manner, we translate the test set (represented by ids) using our new phrase table.

7   Results and Analysis

Since we intend to use supertags as source side contextual features, we had to choose English as the source language, given that supertag information is currently available for English only.

The experiments were carried out on the English—Chinese data provided by the IWSLT 2006 evaluation campaign, extracted from the Basic Travel Expression Corpus (BTEC). The training, development and test sets contain 40,274, 489 and 486 sentences respectively. This multilingual speech corpus contains sentences similar to those that are usually found in phrase-books for tourists going abroad. It is observed that sentence length of this speech corpus is very small.

Although our main focus was to see the effect on translation quality of incorporating supertags as a source contextual feature, we also carried out experiments with different contextual features (both individually and in collaboration) and with varying windows of context size. The best results obtained from E1, E2 and E3 are reported in the tables.

The results with uniform context size are shown in Table 1. As demonstrated by (Max et al., 2008), it is clear that translation from English can benefit from the addition of source language features, as the inclusion of any type of contextual feature easily improves upon the baseline across all evaluation metrics. Adding source language POS tags adds almost a whole BLEU point (a relative improvement of 4.67%), and further improvements are to be seen when

	BLEU		NIST		WER		PER	
Baseline	*20.56*		*4.67*		*57.82*		*48.99*	
Context length	±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**	21.34	4.82	4.70	56.63	57.61	**48.43**	49.27
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 + CCG	21.52	21.53	4.75	4.78	57.21	57.38	48.95	49.45
Word + LTAG	21.64	21.37	4.78	4.79	57.15	57.06	48.89	48.95
Word + POS	21.77	**21.89**	4.78	**4.83**	56.77	56.51	48.58	**48.03**

Table 1: Experiments with uniform context size

Experiment	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
Word±2 +POS±1	21.61	4.77	**56.78**	48.66
POS±2 + CCG±1	21.08	4.68	58.22	50.05
Word±2+POS±2+CCG±1	21.23	4.72	57.47	49.82
CCG±1 + LTAG±1	21.79	4.74	58.28	49.59
CCG ±1+ LTAG±1[#]	**22.11**	**4.82**	56.95	48.81
Word±1+CCG±1+LTAG±1[#]	21.48	4.79	56.83	48.53
Supertag-Pair±1[#]	21.99	**4.82**	56.83	48.72

Table 2: Experiments with varying context size
([#] Syntactic features of focus phrase are ignored)

	Experiment	BLEU	NIST	WER	PER
	Baseline	*20.56*	*4.67*	*57.82*	*48.99*
I	CCG±1	22.08	4.83	57.30	48.63
B	LTAG±1	22.06	4.75	58.05	49.04
1	CCG±1+LTAG±1[#]	21.72	4.76	58.48	49.18
	Supertag-Pair±1[#]	22.03	4.79	57.35	49.15
T	CCG±1	**22.18**	**4.85**	**56.31**	**48.55**
R	LTAG±1	21.39	4.78	56.83	48.72
I	CCG±1+LTAG±1[#]	22.00	4.75	58.16	49.59
B	Supertag-Pair±1[#]	22.13	4.80	57.24	48.92
L					

Table 3: Experiments with IB1 and TRIBL

neighbouring words (5.25% relative increase), CCG supertags (5.79%) and LTAG supertags (6.61%) are used.

Interestingly, with respect to BLEU score, for all bar POS tags and the combination of Word+POS, the best scores are observed when a context window of ±1 words is seen. When ±2 words are used, CCG supertags when used as an individual feature produce the best NIST, WER and PER scores (though these scores are slightly worse than when a context window of ±1 words is used).

When combinations of two features were applied, the Word+POS combination improved BLEU, NIST and PER scores on a ±2 word context window, but no combination improved over the LTAG individual feature when used on a ±1 word context window. Interestingly, when used together with the neighbouring words as a feature, the supertags could not improve over the Words feature, and in most cases caused system performance to deteriorate.

Since LTAG±1 and POS±2 produced the best BLEU scores when used as individual features, we were encouraged to try out combinations of features with varying context sizes. The results can be seen in Table 2. This time, adding CCG supertags to the neighbouring words caused system performance to improve to 22.01 BLEU score, 1.45 points (or 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 a BLEU score of 22.11 (7.54% relative improvement compared to the baseline) was obtained for CCG±1+LTAG±1, when ignoring the syntactic feature information of the focus phrase. We also carried out the best performing experiments on IB1 and TRIBL classifiers, with the results shown in Table3. The differences we see between IGTree, TRIBL, and IB1 are generally small and somewhat unpredictable. When considered as a single concatenated feature, the supertag-pair (LTAG, CCG) 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). Among the three classifiers, however, the IGTree score remains the best on CCG±1+LTAG±1.

8 Conclusion and Future Work

In this paper, we have successfully incorporated supertags as a new feature into a state-of-the-art log-linear phrase-based SMT system that takes into account the contextual information of the source phrases. In addition, we have demonstrated that both neighbouring words and the POS tags of those words can improve translation quality significantly over the baseline system for English-to-Chinese.

Our best result of 1.62 BLEU points improvement over the baseline, a 7.88% relative increase in performance, came about on CCG alone. Most encouragingly, supertags produced good results consistently.

Following the work of (Hassan et al., 2006, 2007, 2008; Hassan, 2009), our ultimate aim is to develop a fully supertagged PB-SMT system, with supertags deployed as source language context (as here), 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 clear the advantages that supertags bring to bear in each phase of the translation process.

Other lines of future work include (i) a manual evaluation of the output sentences, to try to identify the exact role that supertags are playing when used as source language contextual information; (ii) an investigation as to why system performance tends to deteriorate when pairs of features are used, and where one of those pairs is a supertag sequence; and (iii) an investigation as to why a context window of ±1 words seems to work better than larger windows.

Acknowledgements

References

Bangalore, Srinivas and Aravind K. Joshi. 1999. Supertagging: An Approach to Almost Parsing. *Computational Linguistics* **25**(2):237–265.

Berger, Adam, Stephen Della Pietra, and Vincent Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, **22**(1):39–68.

Carpuat, Marine, and Dekai Wu. 2005. Word sense disambiguation vs. statistical machine translation. *ACL-2005*, *43rd Annual meeting of the Association for Computational Linguistics*, Ann Arbor, MI, 387—394.

Carpuat, Marine, and Dekai Wu. 2007. Improving statistical machine translation using word sense disambiguation. *EMNLP-CoNLL-2007*, *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Prague, Czech Republic, 61—72.

Chan, Y. S., H. T. Ng., and D. Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic, 33—40.

Chen, J., Srinivas Bangalore and K. Vijay-Shankar. 2006. Automated Extraction of Tree-Adjoining Grammars from Treebanks. *Natural Language Engineering* **12**(3):251—299.

Chiang, David. 2007. Hierarchical phrase-based translation. *Computational Linguistics* 33(2): 202—228.

Clark, Steven and James Curran. 2004. The Importance of Supertagging for Wide-Coverage CCG Parsing. *Coling-2004. 20th International Conference on Computational Linguistics,* Geneva, Switzerland, 282–288.

Cowan, Brooke, Ivona Kučerová, and Michael Collins. 2006. A discriminative model for tree-to-tree translation. *EMNLP-2006: Conference on Empirical Methods in Natural Language Processing*, Sydney, Australia, 232—241.

Daelemans, Walter, Jakub Zavrel, Ko van der Sloot and Antal van den Bosch, A. 2007. *TiMBL: Tilburg Memory Based Learner, version 6.1, Reference Guide*. ILK Research Group Technical Report Series no. 07-07.

Foster, George, Roland Kuhn, and Howard Johnson. 2006. Phrasetable smoothing for statistical machine translation. *EMNLP-2006: Conference on Empirical Methods in Natural Language Processing*, Sydney, Australia, 53—61.

García-Varea,, Ismael, F. J. Och, H. Ney, and F. Casacuberta. 2001. Refined lexicon models for statistical machine translation using a maximum entropy approach. *ACL-2001*, *39th Annual Meeting of the Association for Computational Linguistics and 10th Meeting of the European Chapter*, Toulouse, France, 204—211.

Giménez, Jesús, and Lluís Màrquez. 2007. Context-aware discriminative phrase selection for statistical machine translation. *ACL 2007: proceedings of the Second Workshop on Statistical Machine Translation,* Prague, Czech Republic, 159—166.

Gimpel, Kevin and Noah A. Smith. 2008. Rich source-side context for statistical machine translation. *ACL-08: HLT. Third Workshop on Statistical Machine Translation*, Columbus, OH, pp.9—17.

Hassan, Hany. 2009. *Lexical Syntax for Statistical Machine Translation.* Ph.D Thesis, Dublin City University, Dublin, Ireland.

Hassan, Hany, Mary Hearne, Khalil Sima'an, and Andy Way. 2006. Syntactic Phrase-Based Statistical Machine Translation. *IEEE 2006 Workshop on Spoken Language Translation*, Palm Beach, Aruba.

Hassan, Hany, Yanjun Ma, and Andy Way. 2007. MaTrEx: the DCU Machine Translation System for IWSLT 2007. In *Proceedings of the International Workshop on Spoken Language Technologies*, Trento, Italy, pp.69—75.

Hassan, Hany, Khalil Sima'an, and Andy Way. 2007. Supertagged phrase-based statistical machine translation. *ACL-2007. 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic, 288—295.

Hassan, Hany, Khalil Sima'an, and Andy Way. 2008. Syntactically Lexicalized Phrase-Based SMT. *IEEE Transactions on Audio, Speech and Language Processing* **6**(7):1260—1273.

He, Zhongjun, Qu Liu and Shouxu Lin. 2008. Improving statistical machine translation using lexicalized rule selection. In *Coling 2008: Proceedings of the Conference,* Manchester, UK, pp.321—328.

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

Ittycheriah, Abe and Salim Roukos. 2007. Direct Translation Model 2. *NAACL-HLT-2007, Human Language Technology: the conference of the North American Chapter of the Association for Computational Linguistics*, Rochester, NY, pp.57—64.

Joshi, Aravind and Yves Schabes. 1992. Tree Adjoining Grammars and Lexicalized Grammars. In M. Nivat and A. Podelski (eds.) *Tree Automata and Languages*, Amsterdam, The Netherlands: North-Holland, pp.409–431.

Koehn, Philipp. 2004. Pharaoh: a beam search decoder for phrase-based statistical machine translation models. *Machine translation: from real users to research: 6th conference of the Association for Machine Translation in the Americas, AMTA 2004,* Berlin: Springer Verlag, 2004, 115—124.

Koehn, Philipp, 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, and E. Herbst. 2007. Moses: open source toolkit for statistical machine translation.

ACL 2007: proceedings of demo and poster sessions, Prague, Czech Republic, 177-180.

Koehn, Philipp, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. *HLT-NAACL 2003, conference combining Human Language Technology conference series and the North American Chapter of the Association for Computational Linguistics conference series,* Edmonton, Canada, 48-54.

Liang, Percy, Alexandre Bouchard-Côté, Dan Klein, and Ben Taskar. 2006. An end-to-end discriminative approach to machine translation. *Coling-ACL 2006: 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, 761—768.

Max, A., R. Makhloufi and P. Langlais. 2008. Explorations in Using Grammatical Dependencies for Contextual Phrase Translation Disambiguation. In *Proceedings of the 12th EAMT Conference,* Hamburg, Germany, pp.112—117.

Och, Franz. 2003. Minimum error rate training in statistical machine translation. *41st Annual meeting of the Association for Computational Linguistics*, Sapporo, Japan, 160–167.

Och, Franz, and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. *40th Annual meeting of the Association for Computational Linguistics*, Philadelphia, PA, 295–302.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. *40th Annual meeting of the Association for Computational Linguistics*, Philadelphia, PA, 311–318.

Steedman, Mark. 2000. *The Syntactic Process.* MIT Press: Cambridge, MA.

Stroppa, Nicolas, Antal van den Bosch and Andy Way. 2007. Exploiting Source Similarity for SMT using Context-Informed Features. *TMI-2007, 11th Conference on Theoretical and Methodological Issues in Machine Translation*, Skövde, Sweden, 231—240.

Vickrey, David, Luke Biewald, Marc Teyssier and Daphne Koller. 2005. Word-sense disambiguation for machine translation. *HLT-EMNLP-2005, Human Technology Conference and Conference on Empirical Methods in Natural Language Processing*, Vancouver, 771—778.

Zens, Richard and Hermann Ney. 2004. Improvements in phrase-based statistical machine translation. *HLT/NAACL 2004, Human Language Technology conference/North American Chapter of the Association for Computational Linguistics annual meeting*, Boston, MA, 257–264.