3

Grammatical Machine Translation

Stefan Riezler and John T. Maxwell III

3.1 Introduction

Recent approaches to statistical machine translation (SMT) piggyback on the central concepts of phrase-based SMT (Och et al. 1999, Koehn et al. 2003) and at the same time attempt to improve on some of its shortcomings by incorporating syntactic knowledge in the translation process. Phrase-based translation with multi-word units excels at modeling local ordering and short idiomatic expressions; however, it lacks a mechanism to learn long-distance dependencies and is unable to generalize to unseen phrases that share non-overt linguistic information. Publicly available statistical parsers can provide the syntactic information that is necessary for linguistic generalizations and for the resolution of non-local dependencies. This information source is deployed in recent work either for *pre-ordering* source sentences before they are input to a phrase-based system (Xia and McCord 2004, Collins et al. 2005), or for *re-ordering* the output of translation models by statistical ordering models that access linguistic information on dependencies and part-of-speech (Lin 2004, Ding and Palmer 2005, Quirk et al. 2005).[1]

While these approaches deploy dependency-style grammars for parsing source and/or target text, a utilization of grammar-based generation on the output of dependency-based translation models has not yet been attempted. Instead, simple target language realization models

This is an extended version of a paper for HLT/NAACL 2006.

[1] A notable exception to this kind of approach is Chiang (2005) who introduces syntactic information into phrase-based SMT via hierarchical phrases rather than by external parsing.

Intelligent Linguistic Architectures: Variations on themes by Ronald M. Kaplan.
Miriam Butt, Mary Dalrymple, and Tracy Holloway King (eds.).

that can easily be trained to reflect the ordering of the reference translations in the training corpus are preferred. The advantage of such models over grammar-based generation seems to be supported, for example, by Quirk et al. (2005)'s improvements over phrase-based SMT as well as over an SMT system that deploys a grammar-based generator (Menezes and Richardson 2001) on n-gram based automatic evaluation scores (Papineni et al. 2001, Doddington 2002). Another data point, however, is given by Charniak et al. (2003) who show that parsing-based language modeling can improve grammaticality of translations, even if these improvements are not recorded under n-gram based evaluation measures.

In this paper we would like to step away from n-gram based automatic evaluation scores for a moment, and investigate the possible contributions of incorporating a grammar-based generator into a dependency-based SMT system. We present a dependency-based SMT model that integrates the idea of multi-word translation units from phrase-based SMT into a transfer system for dependency structure snippets. The statistical components of our system are modeled on the phrase-based system of Koehn et al. (2003), and component weights are adjusted by minimum error rate training (Och 2003). In contrast to phrase-based SMT and to the above cited dependency-based SMT approaches, our system feeds dependency-structure snippets into a grammar-based generator, and determines target language ordering by applying n-gram and distortion models after grammar-based generation. The goal of this ordering model is thus not foremost to reflect the ordering of the reference translations, but to improve the grammaticality of translations.

Since our system uses standard SMT techniques to learn about correct lexical choice and idiomatic expressions, it allows us to investigate the contribution of grammar-based generation to dependency-based SMT.[2] In an experimental evaluation on the test-set that was used in Koehn et al. (2003), we show that for examples that are in coverage of the grammar-based system, we can achieve state-of-the-art quality on n-gram based evaluation measures. To discern the factors of grammaticality and translational adequacy, we conducted a manual evaluation on 500 in-coverage and 500 out-of-coverage examples. This showed that incorporation of a grammar-based generator into an SMT framework provides improved grammaticality over phrase-based SMT on in-coverage examples. Since in our system it is determinable whether

[2] A comparison of the approaches of Quirk et al. (2005) and Menezes and Richardson (2001) with respect to ordering models is difficult because they differ from each other in their statistical and dependency-tree alignment models.

an example is in-coverage, this opens the possibility for a hybrid system that achieves improved grammaticality at state-of-the-art translation quality.

3.2 Phrase-based SMT

Phrase-based SMT starts with a sentence-aligned bilingual corpus of translations. The words are aligned using a noisy channel model (IBM model 4: Brown et al. 1999). The word alignment is then improved by intersecting alignment matrices for both translation directions and refining the intersection alignment by adding directly adjacent alignment points and alignment points that align previously unaligned words (Och et al. 1999). This produces a many-to-many alignment between words in the source and the target sentences that is more suitable for extracting phrase translations than just the noisy channel model.

Next, phrase translations are extracted by collecting all aligned phrase pairs that are consistent with the improved word alignment. The words of a legal phrase pair are only aligned to each other, and not to words outside the phrase pair. For instance, suppose our corpus contains the following aligned sentences (this example is taken from our experiments on German-to-English translation):

Dafür bin ich zutiefst dankbar.
I have a deep appreciation for that.

Suppose further that the following many-to-many bi-directional word alignment has been created

Dafür{6 7} bin{2} ich{1} zutiefst{3 4 5} dankbar{5}

indicating for example that *Dafür* is aligned with words 6 and 7 of the English sentence (*for* and *that*). From this, the following primitive phrase translations can be extracted:

Dafür → *for that*
bin → *have*
ich → *I*
zutiefst dankbar → *a deep appreciation*

Note that *zutiefst* → *a deep appreciation* is not allowed because *appreciation* is also aligned with *dankbar*.

In addition, more complex phrase translations can be extracted which are just combinations of the primitive phrase translations:

bin ich → *I have*
bin ich zutiefst dankbar → *I have a deep appreciation*

Dafür bin ich zutiefst dankbar → I have a deep appreciation for that

Note that the "phrases" do not have to correspond to constituents (e.g. *bin ich → I have*). They are just snippets of the original sentences.

Once the phrase translations have been extracted, a sentence in German can be translated into English by non-deterministically applying all of the phrase translations that match on the German side, allowing the English outputs to be rearranged, and then using a statistical model to pick the best English translation. A beam decoder is used to make this process more efficient.

The Pharaoh system is a freely available phrase-based SMT system (Koehn 2004) that is useful as a benchmark system for our work. Its statistical model has eight components. The first two measure the relative frequency of phrase translations in the source-to-target and target-to-source directions. This is simply the number of times that a particular phrase translation appears in a training corpus divided by the number of times either the source phrase appears (for source-to-target) or the target phrase appears (for target-to-source). The counts are not smoothed in any way. This means that long phrase translations usually have a relative frequency of 1.

The second two components measure lexical frequency in the source-to-target and target-to-source directions. Lexical frequency is just the average of the relative alignment frequencies for each word on the source side (for source-to-target) or on the target side (for target-to-source). The lexical frequencies help measure the quality of the phrase translations, especially those that have a relative frequency of 1.

The next component counts the number of phrase translations. In general, fewer phrase translations produce better results because the phrases are longer.

The next two components measure the language model probability and the word count of each translation. The language model probability gives a measure of the likelihood of a particular string of words. By itself, it is biased toward short sentences since they have fewer probabilities multiplied together. The word count component is used to offset this bias.

The last component measures the distortion probability. This is a measure of how far each phrase gets moved from its default position. It is not lexicalized. In general, less movement is better than more movement.

The Pharaoh system works by applying translation rules to snippets of sentences. It is successful in spite of the simplistic linguistic model

because of the sophisticated statistical model. However, the simplistic linguistic model often causes it to produce garbage translations. This led us to wonder whether it was possible to get better translations by applying a similar statistical model to snippets of dependency-based f-structures instead of snippets of strings, thus improving the linguistic model without losing any of the benefits of the statistical model.

3.3 Extracting F-Structure Snippets

Our method for extracting transfer rules for dependency structure snippets operates on the paired sentences of a sentence-aligned bilingual corpus. Similar to phrase-based SMT, our approach starts with an improved word-alignment that is created by intersecting alignment matrices for both translation directions, and refining the intersection alignment by adding directly adjacent alignment points and alignment points that align previously unaligned words (see Och et al. 1999). Next, source and target sentences are parsed using source and target LFG grammars to produce a set of possible f(unctional) dependency structures for each side (see Riezler et al. (2002) for the English grammar and parser, and Butt et al. (2002) and Rohrer and Forst (this volume) for German). The two f-structures that most preserve dependencies are selected for further consideration. Selecting the most similar instead of the most probable f-structures is advantageous for rule induction since it provides for higher coverage with simpler rules.

In the third step, the many-to-many word alignment created in the first step is used to define many-to-many correspondences between the substructures of the f-structures selected in the second step. The parsing process maintains an association between words in the string and particular predicate features in the f-structure, and thus the predicates on the two sides are implicitly linked by virtue of the original word alignment. The word alignment is extended to f-structures by setting into correspondence the f-structure units that immediately contain linked predicates. These f-structure correspondences are the basis for hypothesizing candidate transfer rules.

To illustrate, consider the aligned sentences that we discussed earlier:

Dafür bin ich zutiefst dankbar.
I have a deep appreciation for that.

We use the same many-to-many bi-directional word alignment that the Pharaoh system uses:

Dafür{*6 7*} *bin*{*2*} *ich*{*1*} *zutiefst*{*3 4 5*} *dankbar*{*5*}

This results in the links between the predicates of the source and target f-structures shown in Figure 1.

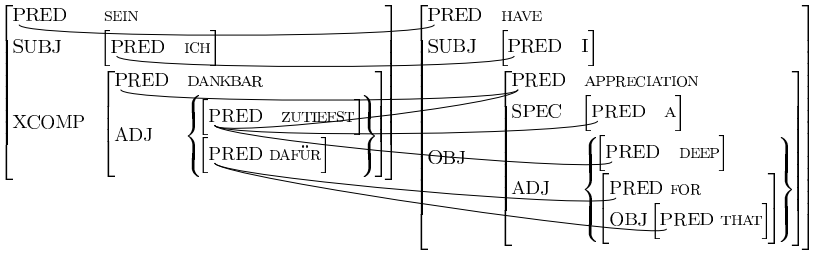

FIGURE 1 F-structure alignment for induction of German-to-English transfer rules.

From these source-target f-structure alignments, transfer rules are extracted in two steps. In the first step, primitive transfer rules are extracted directly from the alignment of f-structure units. These include simple rules for mapping lexical predicates such as:

```
PRED(%X1, ich)   ==>   PRED(%X1, I)
```

and somewhat more complicated rules for mapping local f-structure configurations. For example, the rule shown below is derived from the alignment of the outermost f-structures. It maps any f-structure whose pred is *sein* to an f-structure with pred *have*, and in addition interprets the subj-to-subj link as an indication to map the subject of a source with this predicate into the subject of the target and the xcomp of the source into the object of the target. Features denoting number, person, type, etc. are not shown; variables %X denote f-structure values.

```
PRED(%X1,sein)          PRED(%X1,have)
SUBJ(%X1,%X2)    ==>    SUBJ(%X1,%X2)
XCOMP(%X1,%X3)          OBJ(%X1,%X3)
```

The following rule shows how a single source f-structure can be mapped to a local configuration of several units on the target side, in this case the single f-structure headed by *dafür* into one that corresponds to an English preposition+object f-structure.

```
                          PRED(%X1,for)
PRED(%X1, dafür)   ==>    OBJ(%X1,%X2)
                          PRED(%X2,that)
```

Transfer rules are required to operate only on contiguous units of the f-structure that are consistent with the word alignment. This *transfer contiguity constraint* states that

1. source and target f-structures are each connected;
2. f-structures in the transfer source can only be aligned with f-structures in the transfer target, and vice versa.

This constraint on f-structures is analogous to the constraint on contiguous and alignment-consistent phrases employed in phrase-based SMT. It prevents the extraction of a transfer rule that would translate *dankbar* directly into *appreciation* since *appreciation* is aligned also to *zutiefst* and its f-structure would also have to be included in the transfer. Thus, the primitive transfer rule for these predicates must be:

```
PRED(%X1,dankbar)          PRED(%X1,appreciation)
ADJ(%X1,%X2)         ==>   SPEC(%X1,%X2)
in_set(%X3,%X2)            PRED(%X2,a)
PRED(%X3,zutiefst)         ADJ(%X1,%X3)
                           in_set(%X4,%X3)
                           PRED(%X4,deep)
```

In the second step, rules for more complex mappings are created by combining primitive transfer rules that are adjacent in the source and target f-structures. For instance, we can combine the primitive transfer rule that maps *sein* to *have* with the primitive transfer rule that maps *ich* to *I* to produce the complex transfer rule:

```
PRED(%X1,sein)          PRED(%X1,have)
SUBJ(%X1,%X2)    ==>    SUBJ(%X1,%X2)
PRED(%X2,ich)           PRED(%X2,I)
XCOMP(%X1,%X3)          OBJ(%X1,%X3)
```

In the worst case, there can be an exponential number of combinations of primitive transfer rules, so we allow at most three primitive transfer rules to be combined. This produces $O(n^2)$ transfer rules in the worst case, where n is the number of f-structures in the source.

Other points where linguistic information comes into play is in morphological stemming in f-structures, and in the optional filtering of f-structure phrases based on consistency of linguistic types. For example, the extraction of a phrase-pair that translates *zutiefst dankbar* into *a deep appreciation* is valid in the string-based world, but would be prevented in the f-structure world because of the incompatibility of the types A and N for adjectival *dankbar* and nominal *appreciation*. Similarly, a transfer rule translating *sein* to *have* could be dispreferred because of a mismatch in the verbal types V/A and V/N. However, the transfer of *sein zutiefst dankbar* to *have a deep appreciation* is licensed by compatible head types V.

3.4 Parsing-Transfer-Generation

We use LFG grammars, producing c(onstituent)-structures (trees) and f(unctional)-structures (attribute value matrices) as output, for parsing source and target text (Riezler et al. 2002, Butt et al. 2002, Rohrer and Forst, this volume). To increase robustness, the standard grammar is augmented with a FRAGMENT grammar. This allows sentences that are outside the scope of the standard grammar to be parsed as well-formed chunks specified by the grammar, with unparsable tokens possibly interspersed. The correct parse is determined by a fewest-chunk method.

Transfer converts source into target f-structures by applying all of the induced transfer rules non-deterministically and in parallel. Each fact in the German f-structure must be transferred by exactly one transfer rule. For robustness a default rule is included that transfers any fact as itself. Similar to parsing, transfer works on a chart. The chart has an edge for each combination of facts that have been transferred. When the chart is complete, the outputs of the transfer rules are unified to make sure they are consistent (for instance, that the transfer rules did not produce two determiners for the same noun). Selection of the most probable transfer output is done by beam-decoding on the transfer chart.

LFG grammars can be used bidirectionally for parsing and generation; thus, the existing English grammar used for parsing the training data can also be used for generation of English translations. For in-coverage examples, the grammar specifies c-structures that differ in the linear precedence of subtrees for a given f-structure and realizes the terminal yield according to morphological rules. In order to guarantee non-empty output for the overall translation system, the generation component has to be fault-tolerant in cases where the transfer system operates on a fragmentary parse, or produces non-valid f-structures from valid input f-structures. For generation from unknown predicates, a default morphology is used to inflect the source stem correctly for English. For generation from unknown structures, a default grammar is used that allows any attribute to be generated in any order as any category, with optimality marks set so as to prefer the standard grammar over the default grammar.

3.5 Statistical Models and Training

The statistical components of our system are modeled on the statistical components of the phrase-based system Pharaoh, described in Koehn et al. (2003) and Koehn (2004). Pharaoh integrates the eight statistical components discussed earlier: relative frequency of phrase translations

in source-to-target and target-to-source direction, lexical weighting in source-to-target and target-to-source direction, phrase count, language model probability, word count, and distortion probability.

Correspondingly, our system computes the following statistics for each translation:

1. log-probability of source-to-target transfer rules, where the probability $r(\texttt{e}|\texttt{f})$ of a rule that transfers source snippet `f` into target snippet `e` is estimated by the relative frequency
$$r(\texttt{e}|\texttt{f}) = \frac{\text{count}(\texttt{f ==> e})}{\sum_{\texttt{e'}} \text{count}(\texttt{f ==> e'})}$$
2. log-probability of target-to-source rules
3. log-probability of lexical translations from source to target snippets, estimated from Viterbi alignments $\hat{a}$ between source word positions $i = 1, \ldots, n$ and target word positions $j = 1, \ldots, m$ for stems f_i and e_j in snippets `f` and `e` with relative word translation frequencies $t(e_j|f_i)$:
$$l(\texttt{e}|\texttt{f}) = \prod_j \frac{1}{|\{i|(i,j) \in \hat{a}\}|} \sum_{(i,j)\in\hat{a}} t(e_j|f_i)$$
4. log-probability of lexical translations from target to source snippets
5. number of transfer rules
6. number of transfer rules with frequency 1
7. number of default transfer rules (translating source features into themselves)
8. log-probability of strings of predicates from root to frontier of target f-structure, estimated from predicate trigrams in English f-structures
9. number of predicates in target f-structure
10. number of constituent movements during generation based on the original order of the head predicates of the constituents (for example, `AP[2] BP[3] CP[1]` counts as two movements since the head predicate of CP moved from the first position to the third position)
11. number of generation repairs
12. log-probability of target string as computed by trigram language model
13. number of words in target string

These statistics are combined into a log-linear model whose parameters are adjusted by minimum error rate training (Och 2003).

3.6 Experimental Evaluation

The setup for our experimental comparison is German-to-English translation on the Europarl[3] parallel data set. For quick experimental turnaround we restricted our attention to sentences with 5 to 15 words, resulting in a training set of 163,141 sentences and a development set of 1,967 sentences. Final results are reported on the test set of 1,755 sentences of length 5-15 that was used in Koehn et al. (2003). To extract transfer rules, an improved bidirectional word alignment was created for the training data from the word alignment of IBM model 4 as implemented by GIZA++ (Och et al. 1999). Training sentences were parsed using German and English LFG grammars (Riezler et al. 2002, Butt et al. 2002). The grammars obtain 100% coverage on unseen data. 80% receive full parses; 20% receive FRAGMENT parses. Around 700,000 transfer rules were extracted from f-structures pairs chosen according to a dependency similarity measure. For language modeling, we used the trigram model of Stolcke (2002).

When applied to translating unseen text, the system operates on n-best lists of parses, transferred f-structures, and generated strings. For minimum-error-rate training on the development set, and for translating the test set, we considered 1 German parse for each source sentence, 10 transferred f-structures for each source parse, and 1,000 generated strings for each transferred f-structure. Selection of most probable translations proceeds in two steps: First, the most probable transferred f-structure is computed by a beam search on the transfer chart using the first 10 features described above. These features include tests on source and target f-structure snippets related via transfer rules (features 1-7) as well as language model and distortion features on the target c- and f-structures (features 8-10). In our experiments, the beam size was set to 20 hypotheses. The second step is based on features 11-13, which are computed on the strings that were generated from the selected n-best f-structures.

We compared our system to IBM model 4 as produced by GIZA++ (Och et al. 1999) and a phrase-based SMT model as provided by Pharaoh (Koehn 2004). The same improved word alignment matrix and the same training data were used for phrase-extraction for phrase-based SMT as well as for transfer-rule extraction for LFG-based SMT. Minimum-error-rate training was done using Koehn's implementation of Och (2003)'s minimum-error-rate model. To train the weights for phrase-based SMT, we used the first 500 sentences of the development set; the weights of the LFG-based translator were adjusted on the 750

[3] http://people.csail.mit.edu/koehn/publications/europarl/

TABLE 1 NIST scores on test set for IBM model 4 (M4), phrase-based SMT (P), and the LFG-based SMT (LFG) on the full test set and on in-coverage examples for LFG. Results in the same row that are not statistically significant from each other are marked with a $*$.

	M4	LFG	P
in-coverage	5.13	*5.82	*5.99
full test set	*5.57	*5.62	6.40

TABLE 2 Preference ratings of two human judges for translations of phrase-based SMT (P) or LFG-based SMT (LFG) under criteria of fluency/grammaticality and translational/semantic adequacy on 500 in-coverage examples. Ratings by judge 1 are shown in rows, for judge 2 in columns. Agreed-on examples are shown in boldface in the diagonals.

	adequacy			grammaticality		
j1\j2	P	LFG	equal	P	LFG	equal
P	**48**	8	7	**36**	2	9
LFG	10	**105**	18	6	**113**	17
equal	53	60	**192**	51	44	**223**

sentences that were in coverage of our grammars.

For automatic evaluation, we use the NIST metric (Doddington 2002) combined with the approximate randomization test (Noreen 1989), providing the desired combination of a sensitive evaluation metric and an accurate significance test (see Riezler and Maxwell 2005). In order to avoid a random assessment of statistical significance in our three-fold pairwise comparison, we reduced the per-comparison significance level to .01 so as to achieve a standard experimentwise significance level of .05 (see Cohen 1995). Table 1 shows results for IBM model 4, phrase-based SMT, and LFG-based SMT, where examples that are in coverage of the LFG-based systems are evaluated separately. Out of the 1,755 sentences of the test set, 44% were in coverage of the LFG-grammars; for 51% the system had to resort to the FRAGMENT technique for parsing and/or repair techniques in generation; in 5% of the cases our system timed out. Since our grammars are not set up with punctuation in mind, punctuation is ignored in all evaluations reported below. For in-coverage examples, the difference between NIST scores for the LFG system and the phrase-based system is statistically not significant. On the full set of test examples, the suboptimal quality on out-of-coverage examples overwhelms the quality achieved on

TABLE 3 Preference ratings of two human judges for translations of phrase-based SMT (P) or LFG-based SMT (LFG) under criteria of fluency/grammaticality and translational/semantic adequacy on 500 out-of-coverage examples. Ratings by judge 1 are shown in rows, for judge 2 in columns. Agreed-on examples are shown in boldface in the diagonals.

	adequacy			grammaticality		
j1\j2	P	LFG	equal	P	LFG	equal
P	**156**	1	19	**121**	1	20
LFG	6	**53**	7	0	**23**	11
equal	69	38	**152**	54	21	**250**

in-coverage examples, resulting in a statistically not significant result difference in NIST scores between the LFG system and IBM model 4.

In order to investigate further the quality of in-coverage examples, we randomly selected 500 examples that were in coverage of the grammar-based generator for a manual evaluation. Two independent human judges were presented with the source sentence and the output of the phrase-based and LFG-based systems in a blind test. This was achieved by displaying the system outputs in random order. The judges were asked to indicate a preference for one system translation over the other, or whether they thought them to be of equal quality. These questions had to be answered separately under the criteria of grammaticality/fluency and translational/semantic adequacy. As shown in Table 2, both judges express a preference for the LFG system over the phrase-based system for both adequacy and grammaticality. If we look only at sentences where judges agree, we see a net improvement on translational adequacy of 57 sentences, which is an improvement of 11.4% over the 500 sentences. If this were part of a hybrid system, this would amount to a 5% overall improvement in translational adequacy. Similarly we see a net improvement on grammaticality of 77 sentences, which is an improvement of 15.4% over the 500 sentences or 6.7% overall in a hybrid system. Result differences on agreed-on ratings are statistically significant, where significance was assessed by approximate randomization via stratified shuffling of the preferences between the systems (Noreen 1989). Examples from the manual evaluation are shown in the appendix.

Along the same lines, a further manual evaluation was conducted on 500 randomly selected examples that were out of coverage of the LFG-based grammars. The two judges agreed on a preference for the phrase-based system in 156 cases and for the LFG-based system in 53

cases under the measure of translational adequacy, and on a preference for the phrase-based system in 121 cases and for the LFG-based system in 23 cases under the measure of grammaticality. Across the combined set of 1,000 in-coverage and out-of-coverage sentences, this resulted in an agreed-on preference for the phrase-based system in 204 cases and for the LFG-based system in 158 cases under the measure of translational adequacy. Under the grammaticality measure the phrase-based system was preferred by both judges in 157 cases and the LFG-based system in 136 cases.

3.7 Discussion

The evaluation of the LFG-based translator presented above shows promising results for examples that are in coverage of the employed LFG grammars. However, a back-off to robustness techniques in parsing and/or generation results in a considerable loss in translation quality. The high percentage of examples that fall out of coverage of the LFG-based system can partially be explained by the accumulation of errors in parsing the training data where source and target language parser each produce FRAGMENT parses in 20% of the cases. Together with errors in rule extraction, this results in a large number of ill-formed transfer rules that force the generator to back-off to robustness techniques. In applying the parse-transfer-generation pipeline to translating unseen text, parsing errors can cause erroneous transfer, which can result in generation errors. Similar effects can be observed for errors in translating in-coverage examples. Here disambiguation errors in parsing and transfer propagate through the system, producing suboptimal translations. An error analysis on 100 suboptimal in-coverage examples from the development set showed that 69 suboptimal translations were due to transfer errors, 10 of which were due to errors in parsing.

The discrepancy between NIST scores and manual preference rankings can be explained by the suboptimal integration of transfer and generation in our system, making it infeasible to work with large n-best lists in training and application. Moreover, despite our use of minimum-error-rate training and n-gram language models, our system cannot be adjusted to maximize n-gram scores on reference translation in the same way as phrase-based systems since statistical ordering models are employed in our framework *after* grammar-based generation, thus giving preference to grammaticality over similarity to reference translations.

3.8 Conclusion

We presented an SMT model that marries phrase-based SMT with traditional grammar-based MT by incorporating a grammar-based generator into a dependency-based SMT system. Under the NIST measure, we achieve results in the range of the state-of-the-art phrase-based system of Koehn et al. (2003) for in-coverage examples of the LFG-based system. A manual evaluation of a large set of such examples shows that on in-coverage examples our system achieves significant improvements in grammaticality and also translational adequacy over the phrase-based system. Fortunately, it is determinable when our system is in-coverage, which opens the possibility for a hybrid system that achieves improved grammaticality at state-of-the-art translation quality. Future work thus will concentrate on improvements of in-coverage translations, e.g. by stochastic generation. Furthermore, we intend to apply our system to other language pairs and larger data sets.

Acknowledgments

This paper is dedicated to Ron Kaplan, who gave the first author the great opportunity to work with LFG grammars and the people who invented them by bringing him onto his team five years ago, and who has been a long-time collaborator of the second author. Ron always believed in the power of the combination of deep linguistic knowledge and broad statistical methods. This view was justified in several papers on statistical parsing which we co-authored over the last years. This paper is an attempt to go a step further and show that deep LFG grammars can be deployed to build a hybrid statistical machine translation system that provides improved translational adequacy and grammaticality. Besides the fact that Ron's intuitions always proved to be right, it was always great fun to work with him.

We would also like to thank Sabine Blum for her help with the manual evaluation which was a crucial ingredient of this paper.

Appendix: Examples from manual evaluation

Examples from manual evaluation: Preference for LFG-based system (LFG) over phrase-based system (P) under both adequacy and grammaticality (the first five), preference of phrase-based system over LFG (the second five), together with source (src) sentences and human reference (ref) translations. All ratings are agreed on by both judges.

src: in diesem fall werde ich meine verantwortung wahrnehmen
ref: then i will exercise my responsibility

LFG: in this case i accept my responsibility
P: in this case i shall my responsibilities

src: die politische stabilität hängt ab von der besserung der lebensbedingungen
ref: political stability depends upon the improvement of living conditions
LFG: the political stability hinges on the recovery the conditions
P: the political stability is rejects the recovery of the living conditions

src: und schließlich muß dieser agentur eine kritische haltung gegenüber der kommission selbst erlaubt sein
ref: moreover the agency must be able to criticise the commission itself
LFG: and even to the commission a critical stance must finally be allowed this agency
P: finally this is a critical attitude towards the commission itself to be agency

src: nach der ratifizierung werden co2 emissionen ihren preis haben
ref: after ratification co2 emission will have a price tag
LFG: carbon dioxide emissions have its price following the ratification
P: after the ratification co2 emissions are a price

src: die lebensmittel müssen die sichere ernährung des menschen gewährleisten
ref: man's food must be safe to eat
LFG: food must guarantee the safe nutrition of the people
P: the people of the nutrition safe food must guarantee

src: was wir morgen beschließen werden ist letztlich material für das vermittlungsverfahren
ref: whatever we agree tomorrow will ultimately have to go into the conciliation procedure
LFG: one tomorrow we approved what is ultimately material for the conciliation procedure
P: what we decide tomorrow is ultimately material for the conciliation procedure

src: die verwaltung muß künftig schneller reagieren können
ref: in future the administration must be able to react more quickly
LFG: more in future the administration must be able to react
P: the administration must be able to react more quickly

src: das ist jetzt über 40 jahre her
ref: that was over 40 years ago
LFG: on 40 years ago it is now
P: that is now over 40 years ago

src: das ist schon eine seltsame vorstellung von gleichheit
ref: a strange notion of equality
LFG: equality that is even a strange idea
P: this is already a strange idea of equality

src: frau präsidentin ich beglückwünsche herrn nicholson zu seinem ausgezeichneten bericht
ref: madam president i congratulate mr nicholson on his excellent report
LFG: madam president i congratulate mister nicholson on his report excellented
P: madam president i congratulate mr nicholson for his excellent report

References

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1999. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics* 19(2):263–311.

Butt, Miriam, Helge Dyvik, Tracy H. King, Hiroshi Masuichi, and Christian Rohrer. 2002. The Parallel Grammar project. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING'02), Workshop on Grammar Engineering and Evaluation*, pages 1–7. Taipei, ROC.

Charniak, Eugene, Kevin Knight, and Kenji Yamada. 2003. Syntax-based language models for statistical machine translation. In *Proceedings of the 9th Machine Translation Summit (MTS IX)*, pages 40–46. New Orleans, LA.

Chiang, David. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 236–270. Ann Arbor, MI.

Cohen, Paul R. 1995. *Empirical Methods for Artificial Intelligence*. Cambridge, MA: The MIT Press.

Collins, Michael, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 531–540. Ann Arbor, MI.

Ding, Yuan and Martha Palmer. 2005. Machine translation using probabilistic synchronous dependency insertion grammars. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 541–548. Ann Arbor, MI.

Doddington, George. 2002. Automatic evaluation of machine translation quality using N-gram co-occurrence statistics. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 128–132. San Diego, CA.

Koehn, Philipp. 2004. PHARAOH. A beam search decoder for phrase-based statistical machine translation models. User manual. Tech. rep., USC Information Sciences Institute, Marina del Rey, CA.

Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the Human Language Technology Conference and the 3rd Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL'03)*, pages 127–133. Edmonton, Canada.

Lin, Dekang. 2004. A path-based transfer model for statistical machine translation. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING'04)*, pages 625–630. Geneva, Switzerland.

Menezes, Arul and Stephen D. Richardson. 2001. A best-first alignment algorithm for automatic extraction of transfer-mappings from bilingual corpora. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL'01), Workshop on Data-Driven Machine Translation*, pages 39–46. Toulouse, France.

Noreen, Eric W. 1989. *Computer Intensive Methods for Testing Hypotheses. An Introduction.* New York, NY: Wiley.

Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the Human Language Technology Conference and the 3rd Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL'03)*, pages 160–167. Edmonton, Canada.

Och, Franz Josef, Christoph Tillmann, and Hermann Ney. 1999. Improved alignment models for statistical machine translation. In *Proceedings of the 1999 Conference on Empirical Methods in Natural Language Processing (EMNLP'99)*, pages 20–28. College Park, MD.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: A method for automatic evaluation of machine translation. Tech. Rep. IBM Research Division Technical Report, RC22176 (W0190-022), Yorktown Heights, NY.

Quirk, Chris, Arul Menezes, and Colin Cherry. 2005. Dependency treelet translation: Syntactically informed phrasal SMT. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 271–279. Ann Arbor, MI.

Riezler, Stefan, Tracy H. King, Ronald M. Kaplan, Richard Crouch, John T. Maxwell, III, and Mark Johnson. 2002. Parsing the Wall Street Journal using a Lexical-Functional Grammar and discriminative estimation techniques. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)*, pages 271–278. Philadelphia, PA.

Riezler, Stefan and John T. Maxwell, III. 2005. On some pitfalls in automatic evaluation and significance testing for mt. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*, pages 57–64. Ann Arbor, MI.

Stolcke, Andreas. 2002. SRILM - an extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing*. Denver, CO.

Xia, Fei and Michael McCord. 2004. Improving a statistical MT system with automatically learned rewrite patterns. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING'04)*, pages 508–514. Geneva, Switzerland.