

Grammatical Machine Translation

Stefan Riezler and John T. Maxwell III
Palo Alto Research Center
3333 Coyote Hill Road, Palo Alto, CA 94304

Abstract

We present an approach to statistical machine translation that combines ideas from phrase-based SMT and traditional grammar-based MT. Our system incorporates the concept of multi-word translation units into transfer of dependency structure snippets, and models and trains statistical components according to state-of-the-art SMT systems. Compliant with classical transfer-based MT, target dependency structure snippets are input to a grammar-based generator. An experimental evaluation shows that the incorporation of a grammar-based generator into an SMT framework provides improved grammaticality while achieving state-of-the-art quality on in-coverage examples, suggesting a possible hybrid framework.

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 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 translation models has not yet been attempted in dependency-based SMT. Instead, simple target language realization models 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.'s (2005) 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.

¹A 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.

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 SMT². 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 an 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 an example is in-coverage, this opens the possibility for a hybrid system that achieves improved grammaticality at state-of-the-art translation quality.

²A 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.

2 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; Butt et al. (2002) 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, 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 we have created the many-to-many bi-directional word alignment

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*).

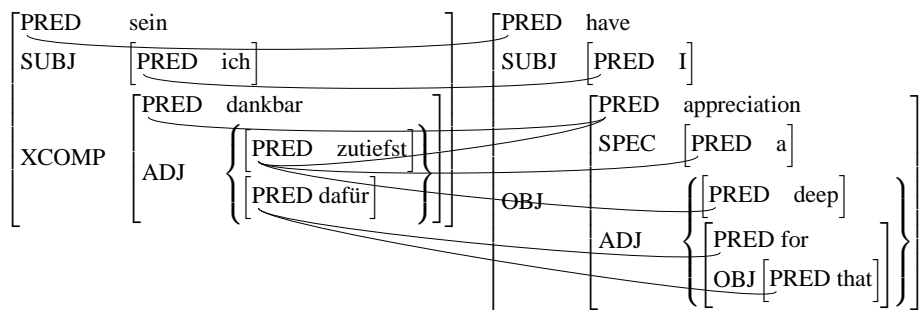


Figure 1: F-structure alignment for induction of German-to-English transfer rules.

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

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

Transfer rules are required to only operate 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, appr.)
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 only allow at most three primitive transfer rules to be combined. This produces $O(n^2)$ trans-

fer 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 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). 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 a target f-structures by non-deterministically applying all of the induced transfer rules 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 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.

4 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 following 8 statistical models: 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 $r(e|f)$ of a rule that transfers source snippet f into target snippet e is estimated by the relative frequency

$$r(e|f) = \frac{\text{count}(f \implies e)}{\sum_{e'} \text{count}(f \implies e')}$$

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 $i = 1, \dots, n$ and target word positions $j = 1, \dots, m$ for stems f_i and e_j in snippets f and e with relative word translation frequen-

cies $t(e_j|f_i)$:

$$l(e|f) = \prod_j \frac{1}{|\{i|(i,j) \in \hat{a}\}|} \sum_{(i,j) \in \hat{a}} t(e_j|f_i)$$

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).

5 Experimental Evaluation

The setup for our experimental comparison is German-to-English translation on the Europarl parallel data set³. 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 1967 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% are parsed as 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 actually 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 (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's (2003) 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 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

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.

j1\j2	adequacy			grammaticality		
	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

Maxwell (2005)). In order to avoid a random assessment of statistical significance in our three-fold pairwise comparison, we reduce the per-comparison significance level to 0.01 so as to achieve a standard experimentwise significance level of 0.05 (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 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 discern the factors of grammaticality and translational adequacy, we conducted a manual

evaluation on randomly selected 500 examples that were in coverage of the grammar-based generator. 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 just look 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 Fig. 2.

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. 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.

6 Discussion

The above presented evaluation of the LFG-based translator 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

(1)	src: ref: LFG: P:	in diesem fall werde ich meine verantwortung wahrnehmen then i will exercise my responsibility in this case i accept my responsibility in this case i shall my responsibilities
(2)	src: ref: LFG: P:	die politische stabilität hängt ab von der besserung der lebensbedingungen political stability depends upon the improvement of living conditions the political stability hinges on the recovery the conditions the political stability is rejects the recovery of the living conditions
(3)	src: ref: LFG: P:	und schließlich muß dieser agentur eine kritische haltung gegenüber der kommission selbst erlaubt sein 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
(4)	src: ref: LFG: P:	nach der ratifizierung werden co2 emissionen ihren preis haben 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
(5)	src: ref: LFG: P:	die lebensmittel müssen die sichere ernährung des menschen gewährleisten 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
(6)	src: ref: LFG: P:	was wir morgen beschließen werden ist letztlich material für das vermittlungsverfahren 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
(7)	src: ref: LFG: P:	die verwaltung muß künftig schneller reagieren können 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
(8)	src: ref: LFG: P:	das ist jetzt über 40 jahre her that was over 40 years ago LFG: on 40 years ago it is now P: that is now over 40 years ago
(9)	src: ref: LFG: P:	das ist schon eine seltsame vorstellung von gleichheit a strange notion of equality LFG: equality that is even a strange idea P: this is already a strange idea of equality
(10)	src: ref: LFG: P:	frau präsidentin ich beglückwünsche herrn nicholson zu seinem ausgezeichneten bericht 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

Figure 2: Examples from manual evaluation: Preference for LFG-based system (LFG) over phrase-based system (P) under both adequacy and grammaticality (ex 1-5), preference of phrased-based system over LFG (6-10) , together with source (src) sentences and human reference (ref) translations. All ratings are agreed on by both judges.

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 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 on the one hand 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.

7 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.

Acknowledgements

We would like to thank Sabine Blum for her invaluable help with the manual evaluation.

References

Miriam Butt, Helge Dyvik, Tracy Holloway King, Hiroshi Masuichi, and Christian Rohrer. 2002. The parallel grammar project. *COLING'02, Workshop on Grammar Engineering and Evaluation*.

Eugene Charniak, Kevin Knight, and Kenji Yamada. 2003. Syntax-based language models for statistical machine translation. *MT Summit IX*.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. *ACL'05*.

Paul R. Cohen. 1995. *Empirical Methods for Artificial Intelligence*. The MIT Press.

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. *ACL'05*.

Yuan Ding and Martha Palmer. 2005. Machine translation using probabilistic synchronous dependency insertion grammars. *ACL'05*.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. *ARPA Workshop on Human Language Technology*.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. *HLT-NAACL'03*.

Philipp Koehn. 2004. Pharaoh: A beam search decoder for phrase-based statistical machine translation models. User manual. Technical report, USC ISI.

Dekang Lin. 2004. A path-based transfer model for statistical machine translation. *COLING'04*.

Arul Menezes and Stephen D. Richardson. 2001. A best-first alignment algorithm for automatic extraction of transfer-mappings from bilingual corpora. *Workshop on Data-Driven Machine Translation*.

Eric W. Noreen. 1989. *Computer Intensive Methods for Testing Hypotheses. An Introduction*. Wiley.

Franz Josef Och, Christoph Tillmann, and Hermann Ney. 1999. Improved alignment models for statistical machine translation. *EMNLP'99*.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. *HLT-NAACL'03*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: a method for automatic evaluation of machine translation. Technical Report IBM RC22176 (W0190-022).

Chris Quirk, Arul Menezes, and Colin Cherry. 2005. Dependency treelet translation: Syntactically informed phrasal SMT. *ACL'05*.

Stefan Riezler and John Maxwell. 2005. On some pitfalls in automatic evaluation and significance testing for mt. *ACL-05 Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*.

Stefan Riezler, Tracy H. King, Ronald M. Kaplan, Richard Crouch, John T. Maxwell, and Mark Johnson. 2002. Parsing the Wall Street Journal using a Lexical-Functional Grammar and discriminative estimation techniques. *ACL'02*.

Stefan Riezler, Tracy H. King, Richard Crouch, and Annie Zaenen. 2003. Statistical sentence condensation using ambiguity packing and stochastic disambiguation methods for lexical-functional grammar. *HLT-NAACL'03*.

Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. *International Conference on Spoken Language Processing*.

Fei Xia and Michael McCord. 2004. Improving a statistical mt system with automatically learned rewrite patterns. *COLING'04*.