Using Feature Structures to Improve Verb Translation in English-to-German Statistical MT

Using Feature Structures to Improve Verb Translation in English-to-German Statistical MT

Philip Williams[⋆]
p.j.williams-2@sms.ed.ac.uk
School of Informatics[⋆]
University of Edinburgh

Philipp Koehn[⋆†]
pkoehn@inf.ed.ac.uk
Center for Speech and Language Processing[†]
The Johns Hopkins University

Abstract

SCFG-based statistical MT models have proven effective for modelling syntactic aspects of translation, but still suffer problems of overgeneration. The production of German verbal complexes is particularly challenging since highly discontiguous constructions must be formed consistently, often from multiple independent rules. We extend a strong SCFG-based string-to-tree model to incorporate a rich feature-structure based representation of German verbal complex types and compare verbal complex production against that of the reference translations, finding a high baseline rate of error. By developing model features that use source-side information to influence the production of verbal complexes we are able to substantially improve the type accuracy as compared to the reference.

1 Introduction

Syntax-based models of statistical machine translation (SMT) are becoming increasingly competitive against state-of-the-art phrase-based models, even surpassing them for some language pairs. The incorporation of syntactic structure has proven effective for modelling reordering phenomena and improving the fluency of target output, but these models still suffer from problems of overgeneration.

One example is the production of German verbal constructions. This is particularly challenging for SMT models since highly discontiguous constructions must be formed consistently, often from multiple independent rules. Whilst the model's grammar may contain rules in which a complete multi-word verb translation is captured in a single discontiguous rule, in practice many verb translations are incompletely or inconsistently produced.

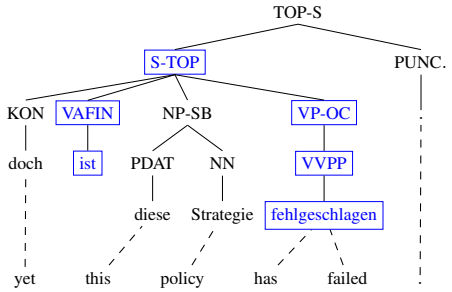

Figure 1: Alignment graph for a sentence pair from the training data. The boxes indicate the components of the target-side verbal complex: a main verb, *fehlgeschlagen*, and an auxiliary, *ist*.

There are many routes by which ill-formed constructions come to be licensed by the model, none of which is easy to address. For instance, Figure 1 shows an example from our training data in which a missing alignment link (between *has* and *ist*) allows the extraction of rules that translate *has failed* to the incomplete *fehlgeschlagen*.

Even with perfect word alignments, the extracted rules may not include sufficient context to ensure the overall grammaticality of a derivation. The extent of this problem will depend partly on the original treebank annotation style, which typically will not have been designed with translation in mind. The problem may be further exacerbated by errors during automatic parsing.

In this paper, we address the problem by focusing on the derivation process. We extend a strong SCFG-based string-to-tree model to incorporate a rich feature-structure based representation

Proceedings of the 3rd Workshop on Hybrid Approaches to Translation (HyTra) @ EACL 2014, pages 21–29,
Gothenburg, Sweden, April 27, 2014.

of German verbal complex types. During decoding, our model composes type values for every clause. When we compare these values against those of the reference translations, we find a high baseline rate of error (either incomplete or mismatching values). By developing model features that use source-side information to influence the production of verbal complexes we are able to substantially improve the type accuracy as compared to the reference.

2 Verbal Complex Structures

Adopting the terminology of Gojun and Fraser (2012), we use the term 'verbal complex' to mean a main verb and any associated auxiliaries within a single clause.

2.1 Feature Structures

We use feature structures to represent the underlying grammatical properties of German verbal complexes. The feature structures serve two main functions: the first is to specify a type for the verbal complex. The types describe clause-level properties and are defined along four dimensions: 1. tense (present, past, perfect, pluperfect, future, future perfect), 2. voice (active, werden-passive, sein-passive), 3. mood (indicative, subjunctive I, subjunctive II), and 4. auxiliary modality (modal, non-modal).

The second function is to restrict the choice of individual word forms that are allowed to combine within a given type. For example, a feature structure value for the verbal complex *hat ... gespielt* belongs to the perfect, active, indicative, non-modal type. Additionally, it specifies that for this type, the verbal complex comprises exactly two verbs: one is a finite, indicative form of the auxiliary *haben* or *sein*, the other is a past-participle.

2.2 The Lexicon

Our model uses a lexicon that maps each German verb in the target-side terminal vocabulary to a set of features structures. Each feature structure contains two top-level features: POS, a part-of-speech feature, and VC, a verbal complex feature of the form described above.

Since a verbal complex can comprise multiple individual verbs, the lexicon entries include partial VC structures. The full feature structure values are composed through unification during decoding.

$$\begin{aligned}
&\text{VP-OC} \rightarrow \langle\ \textit{rebuilt}\ ,\ \textit{wieder aufgebaut}\ \rangle \\
&\quad \langle\ \text{VP-OC}_{\text{VC}}\ \rangle = \langle\ \textit{aufgebaut}_{\text{VC}}\ \rangle \\
&\quad \langle\ \textit{aufgebaut}_{\text{POS}}\ \rangle = \text{VVPP}
\end{aligned}$$

$$\begin{aligned}
&\text{S-TOP} \rightarrow \langle\ \text{X}_1\ \textit{have}\ \text{X}_2\ \textit{been}\ \text{X}_3\ , \\
&\qquad \text{PP-MO}_1\ \textit{wurde}\ \text{NP-SB}_2\ \text{VP-OC}_3\ \rangle \\
&\quad \langle\ \text{S-TOP}_{\text{VC}}\ \rangle = \langle\ \textit{wurde}_{\text{VC}}\ \rangle \\
&\quad \langle\ \text{S-TOP}_{\text{VC}}\ \rangle = \langle\ \text{VP-OC}_{\text{VC}}\ \rangle \\
&\quad \langle\ \textit{wurde}_{\text{POS}}\ \rangle = \text{VAFIN}
\end{aligned}$$

Figure 2: SCFG rules with constraints

The lexicon's POS values are derived from the parse trees on the target-side of the training data. The VC values are assigned according to POS value from a small set of hand-written feature structures. Every main verb is assigned VC values from one of three possible groups, selected according to whether the verb is finite, a past-participle, or an infinitive. For the closed class of modal and non-modal auxiliary verbs, VC values were manually assigned.

3 The Grammar

Our baseline translation model is learned from a parallel corpus with automatically-derived word alignments. In the literature, string-to-tree translation models are typically based on either synchronous context-free grammars (SCFGs) (as in Chiang et al. (2007)) or tree transducers (as in Galley et al. (2004)). In this work, we use an SCFG-based model but our extensions are applicable in both cases.

Following Williams and Koehn (2011), each rule of our grammar is supplemented with a (possibly-empty) set of PATR-II-style identities (Shieber, 1984). Figure 2 shows two example rules with identities. The identities should be interpreted as constraints that the feature structures of the corresponding rule elements are compatible under unification. During decoding, this imposes a hard constraint on rule application.

3.1 Identity Extraction

The identities are learned using the following procedure:

1. The syntax of the German parse trees is used to identify verbal complexes and label the participating verb and clause nodes.

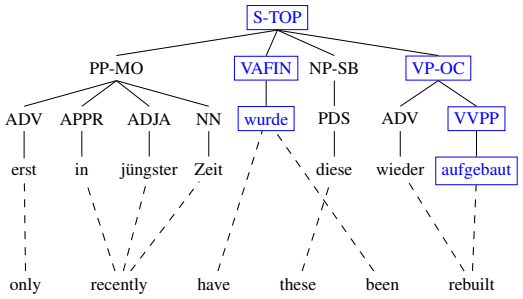

Figure 3: Alignment graph for a sentence pair from the training data. The target sentence has a single verbal complex. Participating nodes are indicated by the boxes.

2. Grammar rule extraction is extended to generate identities between VC values when an SCFG rule contains two or more nodes from a common verbal complex.

3. POS identities are added for terminals that appear in VC identities.

Figure 3 shows a sentence-pair from the training data with the verbal complex highlighted. The rules in Figure 2 were extracted from this sentence-pair.

Crucially, in step 2 of the extraction procedure the identities can be added to SCFG rules that cover only part of a verbal complex. For example, the first rule of Figure 2 includes the main verb but not the auxiliary. On application of this rule, the partial VC value is propagated from the main verb to the root. The second rule in Figure 2 identifies the VC value of an auxiliary with the VC value of a VP-OC subderivation (such as the subderivation produced by applying the first rule).

4 Source-side Features

Since Och and Ney (2002), most SMT models have been defined as a log-linear sum of weighted feature functions. In this section, we define two verbal-complex-specific feature functions. In order to do so, we first describe 'clause projection,' a simple source-syntactic restriction on decoding. We then describe our heuristic method of obtaining probability estimates for a target verbal complex value given the source clause.

4.1 Clause Projection

Our feature functions assume that we have an alignment from source-side clauses to target clauses. In order to satisfy this requirement, we adopt a simple restriction that declarative clauses (both main and embedded) on the source-side must be translated as clauses on the target-side. This is clearly an over-simplification from a linguistic perspective but it appears not to harm translation quality in practice. Table 1 shows small gains in BLEU score over our baseline system with this restriction.

Test Set	Baseline	Clause Proj.
newstest2008	15.7	15.8 (+0.1)
newstest2009	14.9	15.0 (+0.1)
newstest2010	16.5	16.8 (+0.3)
newstest2011	15.4	15.5 (+0.1)

Table 1: Results with and without clause projection (baseline tuning weights are used for clause projection)

Clause projection is implemented as follows:

1. The input sentence is parsed and a set of clause spans is extracted according to the 1-best parse. We use the Berkeley parser (Petrov and Klein, 2007), which is trained on the Penn Treebank and so we base our definition of a declarative clause on the treebank annotation guidelines.

2. We modify the decoder to produce derivations in chart cells only if the cell span is consistent with the set of clause spans (i.e. if source span [i,j] is a clause span then no derivation is built over span [m,n] where $i < m \leq j$ and $n > j$, etc.)

3. We modify the decoder so that grammar rules can only be applied over clause spans if they have a clause label ('S' or 'CS', since the parser we use is trained on the Tiger treebank).

4.2 Verbal Complex Probabilities

When translating a clause, the source-side verbal complex will often provide sufficient information to select a reasonable type for the target verbal complex, or to give preferences to a few candidates. By matching up source-side and target-side verbal complexes we estimate co-occurrence frequencies in the training data. To do this for all pairs in the training data, we would need to align clauses between the source and target training sentences. However, it is not crucial that we identify

every last verbal complex and so we simplify the task by restricting training data to sentence pairs in which both source and target sentences are declarative sentences, making the assumption that the main clause of the source sentence aligns with the main clause of the target.

We represent source-side verbal complexes with a label that is the string of verbs and particles and their POS tags in the order that they occur in the clause, e.g. `plays_VBZ`, `is_addressing_VBZ_VBG`. The target-side feature structures are generated by identifying verbal complex nodes in the training data parse trees (as in Section 3.1) and then unifying the corresponding feature structures from the lexicon.

Many source verbal complex labels exhibit a strong co-occurrence preference for a particular target type. For example, Table 2 shows the three most frequent feature structure values for the target-side clause when the source label is `is_closed_VBZ_VBN`. The most frequent value corresponds to a non-modal, sein-passive construction in the present tense and indicative mood.

RF	F-Structure
0.841	[FIN [AUX [LEMMA sein, MOOD indicative, TENSE present]]] [NON-FIN [PP/SP [PP [LEMMA *]]]]
0.045	[FIN [FULL [LEMMA sein]]] [NON-FIN none]
0.034	[FIN [AUX [LEMMA werden, MOOD indicative, TENSE present]]] [NON-FIN [WPP [PP [LEMMA *], WERDEN none, WORDEN none, SEIN none]]]
...	...

Table 2: Observed values and relative frequencies (RF) for *is closed*, which was observed 44 times in the training data.

4.3 Feature Functions

As with the baseline features, our verbal complex-specific feature functions are evaluated for every rule application r_i of the synchronous derivation. Like the language model feature, they are non-local features and so cannot be pre-computed. Unlike the baseline features, their value depends on whether the source span that the rule is applied to is a declarative clause or not.

Both features are defined in terms of X, the verbal complex feature structure value of the sub-derivation at rule application r_i.

The first feature function, $f(r_i)$, uses the source verb label, l, and the probability estimate, $P(X|l)$, learned from the training data:

$$f(r_i) = \begin{cases} P(X|l) & \text{if } r_i \text{ covers a clause span} \\ & \quad \text{with verb label } l \\ & \quad \text{and } c_l \geq c_{min} \\ 1 & \text{otherwise} \end{cases}$$

The probability estimates are not used for scoring if the number of training observations falls below a threshold, c_{min}. We use a threshold of 10 in experiments.

The second feature function, $g(r_i)$, is simpler: it penalizes the absence of a target-side finite verb when translating a source declarative clause:

$$g(r_i) = \begin{cases} exp(1) & \text{if } r_i \text{ covers a clause span} \\ & \quad \text{and } X \text{ has no finite verb} \\ 1 & \text{otherwise} \end{cases}$$

Unlike f, which requires the verb label to have been observed a number of times during training, g is applied to all source spans that cover a declarative clause.

Dropped finite verbs are a frequent problem in our baseline model and this feature was motivated by an early version of the analysis presented in Section 5.3.

5 Experiments and Analysis

In preliminary experiments, we found that changes in translation quality resulting from our verb translation features were difficult to measure using BLEU. In the following experiments, we measure accuracy by comparing verbal complex values against feature structures derived from the reference sentences.

5.1 Setup

Our experiments use the GHKM-based string-to-tree pipeline implemented in Moses (Koehn et al., 2007; Williams and Koehn, 2012). We extend a conventional baseline model using the constraints and feature functions described earlier.

Data Set	Reference			Baseline			Hard Constraint		
(MC count)	F	E	Total	F	E	Total	F	E	Total
Dev	95.6%	4.4%	100.0%	86.1%	13.9%	100.0%	87.6%	12.4%	100.0%
(633)	637	29	666	545	88	633	559	79	638
Test	92.2%	7.8%	100.0%	83.5%	16.5%	100.0%	85.4%	14.6%	100.0%
(2445)	2439	206	2645	2034	403	2437	2096	359	2455

Table 3: Counts of main clause VC structures that are present and contain at least a finite verb (F) versus those that are empty or absent (E). Declarative main clause counts (MC count) are given for each input set. Counts for the three test sets are aggregated.

We extracted a translation grammar using all English-German parallel data from the WMT 2012 translation task (Callison-Burch et al., 2012), a total of 2.0M sentence pairs. We used all of the WMT 2012 monolingual German data to train a 5-gram language model.

The baseline system uses the feature functions described in Williams and Koehn (2012). The feature weights were tuned on the WMT newstest2008 development set using MERT (Och, 2003). We use the newstest2009, newstest2010, and newstest2011 test sets for evaluation. The development and test sets all use a single reference.

5.2 Main Clause Verb Errors

When translating a declarative main clause, the translation should usually also be a declarative main clause – that is, it should usually contain at least a finite verb. From manually inspecting the output it is clear that verb dropping is a common source of translation error in our baseline system. By making the assumption that a declarative main clause should *always* be translated to a declarative main clause, we can use the absence of a finite verb as a test for translation error.

By evaluating identities, our decoder now generates a trace of verbal complex feature structures. We obtain a reference trace by applying the same process of verbal complex identification and feature structure unification to a parse of our reference data. Given these two traces, we compare the presence or absence of main clause finite-verbs in the baseline and reference.

Since we do not have alignments between the clause nodes of the test and reference trees, we restrict our analysis to a simpler version of this task: the translation of declarative input sentences that contain only a single clause. To select test sentences, we first parse the source-side of the tuning and test sets. Filtering out sentences that are not declarative or that contain multiple clauses leaves 633, 699, 793, and 953 input sentences for newstest2008, 2009, 2010, and 2011, respectively.

Our baseline system evaluates constraints in order to generate a trace of feature structures but constraint failures are allowed and hypotheses are retained. Our hard constraint system discards all hypotheses for which the constraints fail. The f and g feature functions are not used in these experiments.

For all main clause nodes in the output tree, we count the number of feature structure values that contain finite verbs and are complete versus the number that are either incomplete or absent. Since constraint failure results in the production of empty feature structures, incompatible verbal combinations do not contribute to the finite verb total even if a finite verb is produced. We compare the counts of clause nodes with empty feature structures for these two systems against those of the reference set.

Table 3 shows total clause counts for the reference, baseline, and hard constraint system (the 'total' columns). For each system, we record how frequently a complete feature structure containing at least a finite verb is present (the F columns) or not (E).

As expected, the finite verb counts for the reference translations closely match the counts for the source sentences. The reference sets also contain verb-less clauses (accounting for 4.4% and 7.8% of the total clause counts for the dev and test sets). Verb-less clauses are common in the training data and so it is not surprising to find them in the reference sets.

Our baseline and hard constraint systems both fail to produce complete feature structures for a high proportion of test sentences. Table 4 shows the proportion of single-clause declarative source sentences for which the translation trace does not

include a complete feature structure. As well as suggesting a high level of baseline failure, these results suggest that using constraints alone is insufficient.

Test set	Ref.	Baseline	HC
newstest2008	0.0%	13.9%	11.7%
newstest2009	0.6%	18.6%	16.0%
newstest2010	0.0%	14.5%	12.5%
newstest2011	1.4%	17.4%	14.4%

Table 4: Proportion of declarative single-clause sentences for which there is not a complete feature structure for the translation. Ref. is the reference and HC is our hard constraint system.

5.3 Error Classification

In order to verify that the incomplete feature structures indicate genuine translation errors and to understand the types of errors that occur, we manually check 100 sentences from our baseline system and classify the errors. We check the verb constructions of the sentences containing the first 50 failures in newstest2009 and the first 50 failures in newstest2011.

Invalid Combination (27) An ungrammatical combination of auxiliary and main verbs.
Example: *im Jahr 2007* ***hatte*** *es bereits um zwei Drittel* ***reduziert worden*** .

Perfect missing aux (25) There is a past-participle in sentence-final position, but no auxiliary verb.
Example: *der Dow Jones etwas später wieder* ***bereitgestellt*** .

False positive (14) Output is OK. In the sample this happens either because the output string is well-formed in terms of verb structure, but the tree is wrong, or because the parse of the source is wrong and the input does not actually contain a verb.

No verb (13) The input contains at least one verb that should be translated but the output contains none.
Example: *der universelle Charakter der Handy auch Nachteile* .

Invalid sentence structure (13) Verbs are present and make sense, but sentence structure is wrong
Example: *die rund hunderttausend Menschen in Besitz von ihren eigenen Chipcard Opencard in dieser Zeit , diese Kupon* ***bekommen kann*** .

Inf missing aux (5) There is an infinitive in sentence-final position, but no auxiliary verb *or* the main verb is erroneously in final position (the output is likely to be ambiguous for this error type).
Example: *die Preislisten dieser Unternehmen in der Regel nur ausgewählte Personen* ***erreichen*** .

Unknown verb (2) The input verb is untranslated.
Example: *dann* ***scurried*** *ich auf meinem Platz* .

Werden-passive missing aux (1) There is a werden-passive non-finite part, but no finite auxiliary verb.
Example: *die meisten geräumigen und luxuriösesten Wohnung im ersten Stock für die Öffentlichkeit* ***geöffnet worden*** .

In our classification, the most common individual error type in the baseline is the ungrammatical combination of verbs, at 27 out of 100. However, there are multiple categories that can be characterized as the absence of a required verb and combined these total 44 out of 100 errors. There are also some false positives and potentially misleading results in which wider syntactic errors result in the failure to produce a feature structure, but the majority are genuine errors. However, this method fails to identify instances where the verbal complex is grammatical but has the wrong features. For that, we compare accuracy against reference values.

5.4 Feature Structure Accuracy

If we had gold-standard feature structures for our reference sets and alignments between test and reference clauses then we could evaluate accuracy by counting the number of matches and reporting precision, recall, and F-measure values for this task. In the absence of gold reference values, we rely on values generated automatically from our reference sets. This requires accepting some level of error from parsing and verb labelling (we perform a manual analysis to estimate the degree of this problem). We also require alignments between

Data Set	Experiment	F	E	g	m	Prec.	Recall	F1
Dev	Baseline	545	88	637	253	46.4	39.7	42.8
	f	610	48	637	312	51.1	49.0	50.0
	g	600	58	637	289	48.2	45.4	46.7
	$f + g$	627	29	637	317	50.6	49.8	50.2
Test	Baseline	2034	403	2439	993	48.8	40.7	44.4
	f	2370	224	2439	1214	51.2	49.8	50.5
	g	2307	278	2439	1072	46.5	44.0	45.2
	$f + g$	2437	145	2439	1225	50.3	50.2	50.2

Table 5: Feature structure accuracy for the development and test sets. As in Table 3, counts are given for main clause VC structures that are present and contain at least a finite verb (F) versus those that are absent or empty (E). The VC values of the output are compared against the reference values giving the number of matches (m). The counts F, m, and g, (the number of gold reference values) are used to compute precision, recall, and F1 values.

Input	Bangladesh ex-PM **is denied** bail
Reference	Ehemaliger Premierministerin von Bangladesch **wird** Kaution **verwehrt**
Baseline	Bangladesch ex-PM **ist** keine Kaution
$f + g$	Bangladesch ex-PM **wird** die Kaution **verweigert**

Input	the stock exchange in Taiwan **dropped** by 3.6 percent according to the local index .
Reference	Die Börse in Taiwan **sank** nach dem dortigen Index um 3,6 Prozent .
Baseline	die Börse in Taiwan die lokalen Index entsprechend um 3,6 Prozent **gesunken** .
$f + g$	die Börse in Taiwan **fiel** nach Angaben der örtlichen Index um 3,6 Prozent .

Input	the commission **had been assembled** at the request of Minister of Sport Miroslav Drzeviecki .
Reference	Die Kommission **war** auf Anfrage von Sportminister Miroslaw Drzewiecki **zusammengekommen**.
Baseline	die Kommission **hatte** auf Antrag der Minister für Sport Miroslav Drzeviecki **montiert worden** .
$f + g$	die Kommission **war** auf Antrag der Minister für Sport Miroslav Drzeviecki **versammelt** .

Figure 4: Example translations where the baseline verbal complex type does not match the reference but the $f + g$ system does.

test and reference clauses. Here we make the same simplification as in Section 5.2 and restrict evaluation to single-clause declarative sentences.

We test the effect of the f and g features on feature structure accuracy. Their log-linear model weights were tuned by running a line search to optimize the F1 score on a subset of the newstest2008 dev set containing sentences up to 30 tokens in length (all baseline weights were fixed). For the experiments in which both features are used, we first tune the weight for f and then tune g with the f weight fixed.

Table 5 reports feature structure accuracy for the development and test sets. On the test set, the individual f and g features both improve the F1 score. f is effective in terms of both precision and recall, but the g feature degrades precision compared to the baseline. Using both features appears to offer little benefit beyond using f alone.

Compared with the baseline or using hard constraints alone (Table 3), the proportion of sentences with incomplete or inconsistent verbal complex values (column E) is substantially reduced by the f and g feature functions.

To estimate the false match rate, we manually checked the first 50 sentences from the 2009 test set in which one system was reported to agree with reference and the other not:

37/50 Verb constructions are grammatical. We agree with comparisons against the reference value.

9/50 Verb constructions are grammatical. We agree with the comparison for the test system but not the baseline.

4/50 Verb constructions are ungrammatical or difficult to interpret in both baseline and test.

Figure 4 shows some example translations from our system.

5.5 BLEU

Finally, we report BLEU scores for two versions of our dev and test sets: in addition to the full data sets (Table 6), we use sub-sets that contain all source sentences up to 30 tokens in length (Table 7). There are two reasons for this: first, we expect shorter sentences to use simpler sentence structure with less coordination and fewer relative and subordinate clauses. All else being equal, we expect to see a greater degree of high-level structural divergence between complex source and target sentence structures than between simple ones. We therefore anticipate that our naive clause projection strategy is more likely to break down on long sentences. Second, we expect the effects on BLEU score to become diluted as sentence length increases, for the simple reason that verbs are likely to account for a smaller proportion of the total number of words (though this effect seems to be small: in a parse of the newstest2009-30 subset, verbs account for 14.2% of tokens; in the full set they account for 13.1%). We find that the change in BLEU is larger for the constrained test sets, but only slightly.

Experiment	2008	2009	2010	2011
baseline	15.7	14.9	16.5	15.4
f	15.8	15.0	16.9	15.5
g	15.9	15.1	16.9	15.6
$f + g$	15.8	15.0	16.9	15.6

Table 6: BLEU scores for full dev/test sets

Experiment	2008	2009	2010	2011
baseline	16.1	15.7	16.3	15.1
f	16.2	15.8	16.9	15.3
g	16.4	15.9	16.9	15.4
$f + g$	16.3	15.9	16.9	15.4

Table 7: BLEU scores for constrained dev/test sets (max. 30 tokens)

6 Related Work

The problem of verbal complex translation in English-to-German is tackled by Gojun and Fraser (2012) in the context of phrase-based SMT. They overcome the reordering limitation of phrase-based SMT by preprocessing the source-side of the training and test data to move English verbs within clauses into more 'German-like' positions. In contrast, our SCFG-based baseline model does not place any restriction on reordering distance.

Arora and Mahesh (2012) address a similar problem in English-Hindi translation. They improve a phrase-based model by merging verbs and associated particles into single tokens, thus simplifying the task of word alignment and phrase-pair extraction. Their approach relies upon the mostly-contiguous nature of English and Hindi verbal complexes. The discontiguity of verbal complexes rules out this approach for translation into German.

Our model adopts a similar constraint-based extension of SCFG to that described in Williams and Koehn (2011). In that work, constraints are used to enforce target-side agreement between nouns and modifiers and between subjects and verbs. Whilst that constraint model operates purely on the target-side, our verbal complex feature functions also take source-side information into account.

7 Conclusion

We have presented a model in which a conventional SCFG-based string-to-tree system is extended with a rich feature-structure based representation of German verbal complexes, a grammatical construction that is difficult for an SMT model to produce correctly. Our feature structure representation enabled us to easily identify where our baseline model made errors and provided a means to measure accuracy against the reference translations. By developing feature functions that use source-side information to influence verbal complex formation we were able to improve translation quality, measured both in terms of BLEU score where there were small, consistent gains across the test sets, and in terms of task-specific accuracy.

In future work we intend to explore the use of richer models for predicting target-side verbal complex types. For example, discriminative models that include non-verbal source features.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful feedback and suggestions. The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement 287658 (EU-BRIDGE).

References

Karunesh Kumar Arora and R. Mahesh K. Sinha. 2012. Improving statistical machine translation through co-joining parts of verbal constructs in english-hindi translation. In *Proceedings of the Sixth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 95–101, Jeju, Republic of Korea, July. Association for Computational Linguistics.

Chris Callison-Burch, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2012. Findings of the 2012 workshop on statistical machine translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 10–51, Montréal, Canada, June. Association for Computational Linguistics.

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

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What's in a translation rule? In *HLT-NAACL '04*.

Anita Gojun and Alexander Fraser. 2012. Determining the placement of german verbs in english–to–german smt. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 726–735, Avignon, France, April. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Morristown, NJ, USA. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 295–302, Morristown, NJ, USA. Association for Computational Linguistics.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 160–167, Morristown, NJ, USA. Association for Computational Linguistics.

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

Stuart M. Shieber. 1984. The design of a computer language for linguistic information. In *Proceedings of the 10th international conference on Computational linguistics*, COLING '84, pages 362–366, Stroudsburg, PA, USA. Association for Computational Linguistics.

Philip Williams and Philipp Koehn. 2011. Agreement constraints for statistical machine translation into german. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 217–226, Edinburgh, Scotland, July. Association for Computational Linguistics.

Philip Williams and Philipp Koehn. 2012. Ghkm rule extraction and scope-3 parsing in moses. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 388–394, Montréal, Canada, June. Association for Computational Linguistics.