TWO APPROACHES TO AUTOMATIC MATCHING OF ATOMIC GRAMMATICAL FEATURES IN LFG

Anton Bryl   and   Josef van Genabith
CNGL, School of Computing,
Dublin City University

Abstract

The alignment of a bilingual corpus is an important step in data preparation for data-driven machine translation. LFG f-structures provide bilexical labelled dependencies in the form of lemmas and core *grammatical functions* linking those lemmas, but also important *grammatical features* (TENSE, NUMBER, CASE, etc.) representing morphological and semantic information. These grammatical features can often be translated independently from the lemmas or words. It is therefore of practical interest to develop methods that align grammatical features which can be considered translations of each other (e.g. the number features of the corresponding words in the source and target parts of the corpus) in data-driven LFG-based MT. In a parallel grammar development scenario, such as ParGram, this is to a large extent captured through manually hardcoding the correspondences in the hand-crafted grammars, using similar or identical feature names for similar phenomena across languages. However, for a completely automatic learning method it is desirable to establish these correspondences without human assistance. In this paper we present and evaluate two approaches to the automatic identification of correspondences between atomic features of LFG (and similar) grammars for different languages. The methods can be used to evaluate the correspondence between feature names in hand-crafted parallel grammars or find correspondences between features in grammars for different languages where feature alignments are not known.

1   Introduction

Recent attention to deep linguistic representations, such as LFG f-structures, in syntax-based statistical machine translation (SMT) (Avramidis and Kuhn, 2009; Graham and van Genabith, 2009; Riezler and Maxwell III, 2006) poses new problems for processing richly annotated data. In our work we focus on one such problem, namely on deepening the automatic cross-language structure alignment by adding the possibility to align not only words, but also atomic f-structure features.

Each lexical node (essentially a node with a PRED feature) in an f-structure is characterized by a number of atomic-valued features, which contain information about case, tense, gender, etc. Though the sets of features differ for different languages, they are far from being disjoint. A number of features, such as number or case are shared between many languages. It is possible to hardcode the knowledge about these similarities in the grammars; that is, to give the same names to the same (or similar) linguistic properties for grammars for different languages (Butt et al., 1999). However, when one wants to make use of correspondences between such features in a language-agnostic syntax-based SMT system, such "feature name alignment" between source and target grammars cannot simply be taken for granted. Moreover, the degree of correspondence may differ from

feature to feature across grammars. This motivates the need for an automatic way to judge correspondences between atomic features in f-structure representations for arbitrary language pairs.

In this paper we show that, provided that we have parsers for two languages and a word-aligned parallel corpus for these languages, it is possible to automatically identify some of the correspondences between atomic-valued grammatical features. Once identified, these feature pairs can further be used to improve the coverage of transfer-based statistical machine translation. For example, if the algorithm identifies that NUM in English and NUM in German generally *co-vary* (we presume that we do not have any prior knowledge about this correspondence), then we can safely induce transfer rules (from aligned parsed bitext corpora) which abstract over the number feature, effectively providing a back-off to more specific transfer rules (which include NUM information), safe in the knowledge that number features can be transferred independently in the majority of cases. At the same time, if an English feature and a German feature, even though they were intended to capture the same (or similar) phenomena by the grammar writers, do not change simultaneously with enough systematicity, then these features are not safe for use in feature-by-feature translation, and one should keep using more specific "building blocks", e.g. complete sub-f-structures, to create transfer rules.

We present two algorithmic solutions to the problem in question, one assessing the frequency of covariation of feature pairs and the other calculating the mutual predictability of source language (SL) and target language (TL) features. We evaluate both methods on German-English Europarl data. We show that the first method identifies a number of correspondences correctly, without false positives, though some theoretically expected correspondences are not identified. The second method proved to be less precise; it is able to detect many correct correspondences, but produces false positives as well.

The paper is organized as follows: Section 2 explains the motivation, Sections 3 and 4 describe the two proposed methods together with the experimental evaluation, and Section 5 presents conclusions.

2   Motivation

Let us first briefly describe the way in which transfer rules for LFG-based SMT may be induced (Graham and van Genabith, 2009). Consider a simplified example where transfer rules are extracted from a word-aligned English-German string pair "big cat" $\Rightarrow$ "große Katze". After parsing, the English and German f-structures are as shown in Figure 1.

The word alignment determines which German word corresponds to which English word and allows us to abstract over the adjuncts in our example. As the German sub-f-structure with PRED='groß' corresponds to the English sub-f-structure with PRED='big', we can replace both with x and obtain the generalized transfer rule in Figure 2.
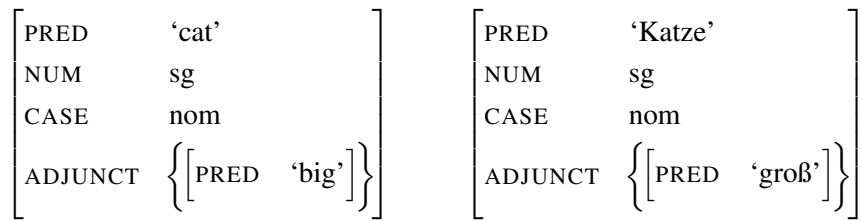
$$
\begin{bmatrix}
\text{PRED} & \text{‘cat’} \\
\text{NUM} & \text{sg} \\
\text{CASE} & \text{nom} \\
\text{ADJUNCT} & \left\{ \begin{bmatrix} \text{PRED} & \text{‘big’} \end{bmatrix} \right\}
\end{bmatrix}
\qquad
\begin{bmatrix}
\text{PRED} & \text{‘Katze’} \\
\text{NUM} & \text{sg} \\
\text{CASE} & \text{nom} \\
\text{ADJUNCT} & \left\{ \begin{bmatrix} \text{PRED} & \text{‘groß’} \end{bmatrix} \right\}
\end{bmatrix}
$$

Figure 1: Parsed structures

$$
\begin{bmatrix}
\text{PRED} & \text{‘cat’} \\
\text{NUM} & \text{sg} \\
\text{CASE} & \text{nom} \\
\text{ADJUNCT} & x
\end{bmatrix}
\Rightarrow
\begin{bmatrix}
\text{PRED} & \text{‘Katze’} \\
\text{NUM} & \text{sg} \\
\text{CASE} & \text{nom} \\
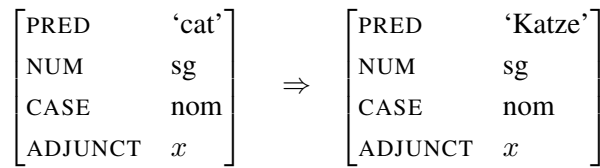\text{ADJUNCT} & x
\end{bmatrix}
$$

Figure 2: Generalized Transfer Rule

To perform this abstraction, the method does not need to know that ADJUNCT in English and ADJUNCT in German are in correpondence. Word alignment alone is sufficient. In fact, this approach will work equally well in the case when e.g. an active construction is translated as a passive construction involving an object and subject argument switch; the word alignment still tells us what corresponds to what. If the method relied on the correspondences of the grammatical function names, its automaticity would be compromised to a certain extent: before being able to learn transfer rules, it would require a team of linguists to manually synchronize the names of grammatical functions for the two languages. Such an alignment cannot be taken for granted if our goal is a fully automatic approach.

Consider the resulting transfer rule (Figure 2) and how it can be improved. It is obvious that the rule could abstract not only over the adjunct *grammatical function*, but also over the number, and maybe also over the case *grammatical feature*. For this, however, we do not have word alignment as our guide. Grammatical functions are bilexical labelled dependencies, and two pairs of aligned words allow us to easily identify a translational correspondence between the grammatical functions which relate them. Grammatical features, such as number and case, are, on the other hand, connected with only one head word. Unless we know which particular German feature corresponds to English NUM, we have no way to align the features and use this alignment for producing more abstract and general transfer rules.

There are two ways to establish correspondences between the features of two languages. One is to turn to human judgement (e.g. use grammars with synchronized terminology and rely on the same-name correspondence), thus making the method less automatic and more dependent on human preprocessing of the data. The other way is to try to extract this correspondence automatically from the data.

An attempt to explore this second way is what the present paper is focused on.

As the methods presented here are intended to compute what is often the judgement of grammar designers, they can also help them in their work. That is, the approaches can be used in grammar design to match a new against an existing grammar, collecting empirical evidence for the correspondence between the features of the two languages.

3 Method 1: Searching for Covariation

3.1 Definition

The idea behind the first method is that if a feature **A** in one language corresponds to a feature **B** in another language, then a change in the value of **A** in a certain phrase frequently corresponds to a change in the value of **B** in the aligned translation of this phrase.

To apply the method to a given pair of languages, we need a parser for each of the languages, and a parallel corpus. The basic procedure (without optimization) is given in Figure 3. It can be seen that for each atomic feature occurring in the first language we will get either a corresponding feature from the second language, or a result "NONE" which means that there is no corresponding feature. Once the features are matched, it is easy to establish correspondences between their values; indeed, if we know that feature **Lang1.A** is translated into feature **Lang2.B**, we just have to calculate which value of **Lang2.B** co-occurs most frequently with each particular value of **Lang1.A**.

It is easy to see that the method is asymmetric with respect to the languages involved. This is meaningful in the context of SMT: even if a change in feature **Lang1.A** can be safely translated with a corresponding change in feature **Lang2.B**, it does not mean that the correspondence will work equally well in the opposite direction.

The method as described above assumes that, apart from the focus feature under consideration, all other local features remain unchanged. However, it is easy to make the method iterative: that is, to remove those features which were already matched from the corpus and to re-run the same algorithm again. This allows the method to make use of the node-pairs which differ in more than one atomic feature.

3.2 Evaluation

In this section we present an experimental evaluation of the algorithm, and discuss the results.

For our experiments we used 219,667 sentences from the German-English part of the Europarl corpus (Koehn, 2005) parsed into f-structures with the XLE parser (Kaplan et al., 2002) using English (Riezler et al., 2002) and German (Dipper, 2003; Rohrer and Forst, 2006) LFGs. For word alignment Giza++ (Och et al., 1999) was used.

1. Word-align the bitext corpus using a standard method based on co-occurrence of lexemes.

2. Parse both language sections of the corpus with the corresponding parsers. *We will say that a node x of an f-structure in the first language corresponds to a node y in the second language if the heads of these two nodes match in the word alignment. The notation is $y = a(x)$.*

3. For each atomic feature **A** occurring in the first language the following actions are performed:

   (a) For the first language all pairs (x_1, x_2) of sub-structures are identified which differ in the value of feature **A** but agree in the values of all other atomic features. For example, for **A**=NUM, if in one local f-structure the word "dog" occurs in plural and in another sentence in singular, while the rest of the features, including e.g. case, carry the same values, these two occurrences form a pair (Figure 4).

   (b) For each such pair:

       i. Compare the aligned structures $a(x_1)$ and $a(x_2)$ in the other language of the bitext.

       ii. If they differ only in the value of a single atomic feature **B**, increase the counter $C_{A,B}$ and the counter C_A

       iii. If they are identical, increase the counter $C_{A,NONE}$ and the counter C_A

4. *Search for correspondences:*
   while at least one feature **A** of the first language with $C_A > 0$ has no correspondence selected, repeat:

   (a) Find the correspondence with the highest normalized score: $(A, B) = \arg\max_{A,B} \frac{C_{A,B}}{C_A}$

   (b) Record **B** as the correspondence to **A**: add the pair (A $\Rightarrow$ B) to the output.

   (c) For all possible values of B_i set $C_{A,B_i} := 0$ *(only one correspondence is chosen for each feature of the first language; once matched, feature **A** is excluded from further consideration).*

   (d) If $B \neq NONE$, for all possible values of A_i set $C_{A_i,B} := 0$ *(each answer except "NONE" may be chosen only once; once chosen, feature **B** is excluded from further consideration).*

Figure 3: The algorithm.

$$
\begin{bmatrix}
\text{PRED} & \text{«dog»} \\
\text{NUM} & \text{sg} \\
\text{CASE} & \text{nom} \\
\text{PERS} & \text{3}
\end{bmatrix}
\begin{matrix}
= \\
\neq \\
= \\
=
\end{matrix}
\begin{bmatrix}
\text{PRED} & \text{«dog»} \\
\text{NUM} & \text{pl} \\
\text{CASE} & \text{nom} \\
\text{PERS} & \text{3}
\end{bmatrix}
$$

$$\Uparrow aligned \Downarrow \qquad\qquad \Uparrow aligned \Downarrow$$

$$
\begin{bmatrix}
\text{PRED} & \text{«Hund»} \\
\text{NUM} & \text{sg} \\
\text{CASE} & \text{nom} \\
\text{PERS} & \text{3} \\
\text{GEND} & \text{masc}
\end{bmatrix}
\begin{matrix}
= \\
\neq \\
= \\
= \\
=
\end{matrix}
\begin{bmatrix}
\text{PRED} & \text{«Hund»} \\
\text{NUM} & \text{pl} \\
\text{CASE} & \text{nom} \\
\text{PERS} & \text{3} \\
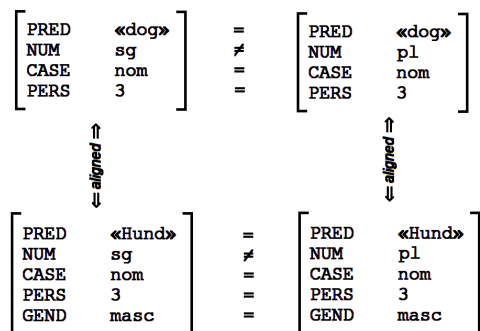\text{GEND} & \text{masc}
\end{bmatrix}
$$

Figure 4: Simultaneous change of the values of NUM in parallel data. Finding such a situation, the algorithm increases the probability counter for the (Eng.NUM ⇒ Ger.NUM) correspondence.

Grammatical features whose values are non-atomic but are sub-structures without a PRED, e.g. TNS-ASP, were also treated as atomic features. Their values were considered equal if and only if the sub-structures were completely identical. Two iterations of the algorithm were run. As the algorithm is asymmetric with respect to the order of languages, the procedure was performed in both directions: correspondences for German atomic features were found in English, and vice versa. Additionally, we ran the first iteration of the method excluding the step 4.c from the algorithm; that is, permitting one feature of the second language to be chosen as a correspondence for several features of the first language.

The results are presented in Table 1. As the XLE LFG grammars for different languages are very consistent (Butt et al., 1999), all the pairs identified consist of same-name features. There are no false positives, but some expected correspondences were not identified, e.g. (ATYPE ⇒ ATYPE) for English-to-German.

The exclusion of the step 4.c has no effect on German-to-English results. For English-to-German, two more pairs emerge: (PFORM ⇒ CASE) and (DEG-DIM ⇒ DEGREE). We observe that these pairs, though not corresponding to exact matches, are far from being random: the second pairs up two closely related attributes, while the first reflects the similarity of semantic functions expressed by prepositions and noun cases.

4 Method 2: Measuring the Predictability

4.1 Definition

The second method makes use of the predictability of target features by the source features. For each possible pair (**Lang1.A**, **Lang2.B**) we calculate the best possible accuracy of the deterministic prediction of the value of **Lang2.B** in the target language structure by the value of **Lang1.A** in the aligned source language f-structure (the absence of a certain feature in a structure is considered here a special *feature*

a. German-to-English

Pair	N_{it}	$\frac{C_{A,B}}{C_A}$
(NUM ⇒ NUM)	1	0.88
(TNS-ASP ⇒ TNS-ASP)	1	0.70
(CLAUSE-TYPE ⇒ CLAUSE-TYPE)	1	0.62
(CASE ⇒ CASE)	1	0.51
(ATYPE ⇒ ATYPE)	1	0.79
(COMP-FORM ⇒ COMP-FORM)	2	0.92
(PASSIVE ⇒ PASSIVE)	2	0.64

b. English-to-German

Pair	N_{it}	$\frac{C_{A,B}}{C_A}$
(NUM ⇒ NUM)	1	0.67
(TNS-ASP ⇒ TNS-ASP)	1	0.81
(CASE ⇒ CASE)	1	0.86
(DEGREE ⇒ DEGREE)	1	0.98
(PASSIVE ⇒ PASSIVE)	2	0.55

Table 1: Experimental results. N_{it} is the number of iteration on which the pair emerged. $\frac{C_{A,B}}{C_A}$ is the normalized score (see algorithm in Figure 3).

absent value of this feature; so in the end each feature is considered to be occurring in each structure, sometimes with this special value). This accuracy is calculated as follows:

$$Acc_{best}(Lang1.A, Lang2.B) = \frac{1}{N} \times \sum_{a \in \mathfrak{V}(Lang1.A)} \max_{b \in \mathfrak{V}(Lang2.B)} \mathfrak{C}(a, b), \quad (1)$$

where $\mathfrak{V}(.)$ is the set of values of the given feature, $\mathfrak{C}(.,.)$ is the counter of co-occurrences of values, and N is the total number of aligned pairs of substructures.

The idea behind the above formula is the following: for each input (a value of Lang1.A) a deterministic predicting algorithm gives only one output (a value of Lang2.B); then, the best possible answer in each case is the value of Lang2.B which collocates with the given value of Lang2.A most frequently.

The accuracy of the pick-most-frequent baseline is then subtracted from the value of Acc_{best}. The resulting value, that is the increase in prediction accuracy over the baseline, is used as the matching scores for the pair of features. Once the matching scores for all feature pairs are calculated, a greedy algorithm is used to establish a one-to-one correspondence.

Pair	Score
(NUM ⇒ NUM)	0.3481754277389169
(PERS ⇒ PERS)	0.34524304388880844
(NTYPE ⇒ NTYPE)	0.3207387801123815
(CASE ⇒ CASE)	0.20730678870766603
(PASSIVE ⇒ PASSIVE)	0.16556148908789317
(CLAUSE-TYPE ⇒ CLAUSE-TYPE)	0.14868700122581083
(PRON-TYPE ⇒ PRON-FORM)	0.1256236959989163
(VTYPE ⇒ VTYPE)	0.11999399711277062
(TNS-ASP ⇒ STMT-TYPE)	0.11956901394608822
(PRON-FORM ⇒ PRON-TYPE)	0.09944473293128155
(ATYPE ⇒ DEG-DIM)	0.08733271268083366
(DEGREE ⇒ DEGREE)	0.0526062756732995
(PTYPE ⇒ PTYPE)	0.047626004188740335
(DET-TYPE ⇒ DET-TYPE)	0.031052988760523315
(ADJUNCT-TYPE ⇒ ADJUNCT-TYPE)	0.021342123401830882
(NUMBER-TYPE ⇒ NUMBER-TYPE)	0.010531614099347783
(ADV-TYPE ⇒ ADV-TYPE)	0.0043149072142222062
(COMP-FORM ⇒ COMP-FORM)	0.0032471370079325762
(DEG-DIM ⇒ ATYPE)	0.0027398133527054827
(PRT-FORM ⇒ TNS-ASP)	3.2670580938708133E-4

Table 2: Experimental results for Method 2: German-to-English.

4.2 Evaluation

For this experiment we used the same data and the same alignment tool as in the previous case (see Section 3.2).

In Tables 2 and 3 we present the results. Though many correspondences are established correctly, there are also clear false positives, e.g. TNS-ASP is matched incorrectly in all cases. One possible reason is the great number of possible values of TNS-ASP, which results in a noisy estimate of prediction accuracy (see the formula in Section 4.1). It may be better to treat sub-features of TNS-ASP as separate features when using this method.

5   Conclusion

In this paper we presented two methods for the automatic identification of functionally similar atomic features in Lexical-Functional Grammars for a given pair of languages. The experimental evaluation shows that the first method, based on the covariation of feature values, is capable of finding strong correspondences for the German-English language pair. In our experiments it produced no false positives and we consider it suitable for inclusion in an MT system. The second method,

Pair	Score
(NUM ⇒ NUM)	0.36233400755141965
(NTYPE ⇒ NTYPE)	0.35840689747679527
(PERS ⇒ PERS)	0.35432041871466496
(CASE ⇒ CASE)	0.24313951002097026
(PASSIVE ⇒ PASSIVE)	0.1679174895181886
(CLAUSE-TYPE ⇒ CLAUSE-TYPE)	0.1540895997322606
(TNS-ASP ⇒ TNS-ASP)	0.1346346672049787
(PRON-TYPE ⇒ PRON-FORM)	0.13356424085389743
(VTYPE ⇒ VTYPE)	0.12779907858337175
(DEGREE ⇒ DEG-DIM)	0.08904991028870966
(HUMAN ⇒ PRON-TYPE)	0.08029658512744847
(ATYPE ⇒ ATYPE)	0.062197614516362514
(PTYPE ⇒ PTYPE)	0.058877433526656406
(DET-TYPE ⇒ DET-TYPE)	0.02897588353336317
(ADJUNCT-TYPE ⇒ ADJUNCT-TYPE)	0.02128900050599558
(NUMBER-TYPE ⇒ NUMBER-TYPE)	0.014094832337500382
(ADV-TYPE ⇒ ADV-TYPE)	0.006284438577315726
(COMP-FORM ⇒ COMP-FORM)	0.003548609441797891
(DEG-DIM ⇒ DEGREE)	6.374747500235733E-4

Table 3: Experimental results for Method 2: English-to-German.

based on mutual predictability of features, proved to be less precise, producing a considerable number of false positives; some improvement is needed before the method becomes accurate enough for practical use.

ParGram XLE LFG grammars are developed manually in parallel with close alignment of the feature space used in the grammars across languages (Butt et al., 1999). It would be interesting to test these methods also on non-parallel grammars developed independently, such as the HPSG and LFG grammars for English and Norwegian, respectively, used in the LOGON project (Lønning et al., 2004). Running the algorithms on two different grammars for one language (e.g. English LFG and English HPSG) might also be of interest.

References

Avramidis, E. and Kuhn, J. 2009. Exploiting XLE's finite state interface in LFG-based statistical machine translation. In *Proceedings of LFG'09*.

Butt, M., King, T. H., Niño, M.-E. and Segond, F. 1999. *A Grammar Writer's Cookbook*. CSLI Publications.

Dipper, S. 2003. *Implementing and Documenting Large-scale Grammars - German LFG*. Ph. D.thesis, IMS, University of Stuttgart.

Graham, Y. and van Genabith, J. 2009. An Open Source Rule Induction Tool for Transfer-Based SMT. *The Prague Bulletin of Mathematical Linguistics* Special Issue: Open Source Tools for Machine Translation, 91, 37–46.

Kaplan, R. M., King, T. H. and Maxwell III, J. T. 2002. Adapting existing grammars: the XLE experience. In *COLING'02, Workshop on Grammar Engineering and Evaluation*, pages 29–35.

Koehn, P. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *MT Summit X*, pages 79–86.

Lønning, J. T., Oepen, S., Beermann, D., Hellan, L., Carroll, J., Dyvik, H., Flickinger, D., Johannessen, J. B., Meurer, P., Nordgård, T., Rosén, V. and Velldal, E. 2004. LOGON. A Norwegian MT Effort. Uppsala, Sweden.

Och, F. J., Tillmann, C. and Ney, H. 1999. Improved Alignment Models for Statistical Machine Translation. In *Proceedings of EMNLP99*, pages 20–28.

Riezler, S., King, T. H., Kaplan, R. M., Crouch, R., Maxwell III, J. T. and Johnson, M. 2002. Parsing the Wall Street Journal using a Lexical-Functional Grammar and discriminative estimation techniques. In *ACL'02*, pages 271–278.

Riezler, S. and Maxwell III, J. T. 2006. Grammatical Machine Translation. In *HLT-ACL*, pages 248–255.

Rohrer, C. and Forst, M. 2006. Improving coverage and parsing quality of a large-scale LFG for German. In *In Proceedings of the Language Resources and Evaluation Conference (LREC-2006)*.