Guessing the Grammatical Function of a Non-Root F-Structure in LFG

Anton Bryl
CNGL,
Dublin City University,
Dublin 9, Ireland

Josef van Genabith
CNGL,
Dublin City University,
Dublin 9, Ireland

Yvette Graham
NCLT,
Dublin City University,
Dublin 9, Ireland

{abryl,josef,ygraham}@computing.dcu.ie

Abstract

Lexical-Functional Grammar (Kaplan and Bresnan, 1982) f-structures are bilexical labelled dependency representations. We show that the Naive Bayes classifier is able to guess missing grammatical function labels (i.e. bilexical dependency labels) with reasonably high accuracy (82–91%). In the experiments we use f-structure parser output for English and German Europarl data, automatically "broken" by replacing grammatical function labels with a generic UNKNOWN label and asking the classifier to restore the label.

1 Introduction

The task of labeling unlabelled dependencies, a sub-task of dependency parsing task, can occur in transfer-based machine translation (when only an inexact match can be found in the training data for the given SL fragment) or in parsing where the system produces fragmented output. In such cases it is often reasonably straightforward to guess which fragments are dependent on which other fragments (e.g. in transfer-based MT). What is harder to guess are the labels of the dependencies connecting the fragments.

In this paper we systematically investigate the labelling task by automatically deleting function labels from Lexical-Functional Grammar-based parser output for German and English Europarl data, and then restoring them using a Naive Bayes classifier trained on attribute names and attribute values of the f-structure fragments. We achieve 82% (German) to 91% (English) accuracy for both single and multiple missing function labels.

The paper is organized as follows: in Section 2 we define the problem and the proposed solution more formally. Section 3 details the experimental evaluations, and in Section 4 we present our conclusions.

$$\begin{bmatrix} \text{PRED} & \text{'adopt'} & \\ \text{UNKNOWN} & f_1 & \begin{bmatrix} \text{PERS} & 3 & \\ \text{NUM} & \text{sg} & \\ \text{PRED} & \text{'resolution'} & \\ \text{SPEC} & f_2 & \begin{bmatrix} \text{DET} & f_3 & [\text{PRED 'the'}] \end{bmatrix} \end{bmatrix} \\ \text{SUBJ} & f_4 & \begin{bmatrix} \text{PERS} & 3 & \\ \text{NUM} & \text{sg} & \\ \text{PRED} & \text{'Parliament'} & \end{bmatrix} \end{bmatrix}$$

Figure 1: Example of a "broken" f-structure (simplified). The sentence is *'Parliament adopted the resolution.'* The missing function of f_1 is OBJ.

2 Guessing Unknown Grammatical Functions

Let us introduce some useful definitions. By dependent f-structure of the parent f-structure f_P we mean an f-structure f_d which bears a grammatical function within f_P, or belongs to a set which bears a grammatical function within f_P. E.g., in Figure 1 f_2 is a dependent f-structure of f_1. In this paper we will not distinguish between these two situations, but simply refer to multiple f-structures bearing the same function within the same parent for set-valued grammatical functions. $C(\phi, f_P)$ denotes the number of dependent f-structures of f_P which bear the grammatical function ϕ in f_P (either directly or as members of a set).

Let us formalize the simple case when the grammatical function of only one dependent f-structure is missing. Let F_P be the set of f-structures which have a dependent f-structure with an UNKNOWN label instead of the grammatical function. Let Φ be the set of all grammatical functions of the given grammar. We need a guessing function $G : F_P \rightarrow \Phi$, such that $G(f_P)$ is a meaningful replacement for the UNKNOWN label in f_P. As the set Φ is finite, the problem is evidently a classification task.

F-structures are characterized by attributes some of which potentially carry information about the f-structure's grammatical function, even if

Proceedings of the 11th International Conference on Parsing Technologies (IWPT), pages 146–149,
Paris, October 2009.

Language	N-GF	N-DEP	AVG-DEP	MIN-DEP	MAX-DEP
English	24	9724	1.57	1	5
German	39	10910	1.55	1	5

Table 1: Data used in the evaluation. **N-GF** is the number of different grammatical functions occurring in the dataset. **N-DEP** is the number of dependent f-structures in the test set. **AVG-DEP**, **MIN-DEP**, **MAX-DEP** is the average, min. and max. number of dependant structures per parent in the test set.

we observe these attributes completely separately from each other. For example, it seems likely that an f-structure with an ATYPE attribute is an ADJUNCT, while an f-structure which has CASE is probably a SUBJ or an OBJ. Given this, Naive Bayes appears to be a promising solution here. Below we describe a way to adapt this classifier to the problem of grammatical function guessing.

Let $\Phi_P \subseteq \Phi$ be the set of grammatical functions which are already present in f_P. Let $\Xi = \{\xi_1..\xi_n\}$ be the set of features, and let $X = \{x_1..x_n\}$ be the values of these features for the f-structure f_d for which the function should be guessed. Then the answer ϕ_d is chosen as follows:

$$\phi_d = \arg\max_{\phi \in \Phi} \left(p(\phi) M_P(\phi) \prod_{i=1}^{n} p(\xi_i = x_i | \phi) \right) \quad (1)$$

$$M_P(\phi) = \begin{cases} p(C(\phi, f_P) > 1), \textit{ if } \phi \in \Phi_P \\ 1, \textit{ otherwise} \end{cases} \quad (2)$$

where the probabilities are estimated from the training data. Equation (2) states that if ϕ is already present in the parent f-structure, the probability of ϕ being set-valued is considered.

We propose two ways of building the feature set Ξ. First, it is possible to consider the presence/absence of each particular attribute in f_d as a binary feature. Second, it is possible to consider atomic attribute values as features as well. To give a motivating example, in many languages the value of CASE is extremely informative when distinguishing objects from subjects. We use only those atomic attribute values which do not represent words. E.g., NUM, PRED or NUM=sg are features, while PRED='resolution' is not a feature. This distinction prevents the feature set from growing too large and thus the probability estimates from being too inaccurate.

If grammatical functions are missing for several dependent f-structures, it is possible to use the same approach, guessing the missing functions one by one. In general, however, these decisions will not be independent. To illustrate this, let us consider a situation when the functions are to be guessed for two dependent f-structures of the same parent f-structure, OBJ being the correct answer for the first and SUBJ for the second. If the guesser returns SUBJ for the first of the two, this answer will not only be incorrect, but also decrease the probability of the correct answer for the second by decreasing $M_P(\texttt{SUBJ})$ in Equation (1). This suggests that in such cases maximization of the joint probability of the values of all the missing functions may be a better choice.

3 Experimental Evaluation

We present two experiments which assess the accuracy of the proposed approach and compare different variants of it in order to select the best, and an additional one which assesses the usefulness of the approach for practical machine translation.

3.1 Data Used in the Evaluation

For our experiments we used 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 (Butt et al., 2002) LFGs. We parsed only sentences of length 5–15 words. For the first two experiments, we picked 2000 sentences for training and 1000 for testing for both languages. We ignored robustness features (FIRST, REST), functions related to c-structure constraints (MOTHER, LEFT_SISTER, etc.), and TOPIC. Of the remaining functions, we considered only those occurring in the PREDs-only part of f-structure. If a dependent f-structure has multiple functions within the same parent f-structure, only the first function occurring in the description is considered. This does not unduely influence the results, as the grammatical function of an f-structure, after exclusion of TOPIC, carries multiple labels in only about 2% of the cases in the English data and about 1% in the German data. In Table 1 we provide some useful statistics to help the reader interpret the results of the experiments.

Language	MF	NB-CASE	NB-N	NB-N&V
English	36.3%	56.7%	85.6%	91.6%
German	23.4%	51.0%	74.8%	82.5%

Table 2: Experiment 1: Guessing a Single Missing Grammatical Function. **MF** is the pick-most-frequent classifier. **NB-CASE** is Naive Bayes (NB) with only CASE values used as features. **NB-N** is NB with only attribute names used as features. **NB-N&V** is NB with both attribute names and atomic attribute values used as features.

3.2 Experiment 1: Guessing a Single Missing Grammatical Function

The goal of this experiment is to evaluate the accuracy of the Bayesian guesser in the case when the grammatical function is unknown only for one dependent f-structure, and to assess whether the inclusion of attribute values into the feature set improves the results, and whether attributes other than CASE are useful.

Procedure. As a baseline, we used a pick-most-frequent algorithm **MF** which considers only the function's prior probability and the presence of this function in the parent (returning to Equations (1) and (2), **MF** is in fact Naive Bayes with an empty feature set Ξ). The guesser was evaluated in three variants: **NB-CASE** with the feature set formed only from the values of CASE attributes (if the f-structure has no CASE feature, the classifier degenerates to **MF**), **NB-N** with the feature set formed only from attribute names, and **NB-N&V** with the feature set formed from both attribute names and values. All grammatical functions in the test set were used as test cases. At each step in the evaluation, one function was removed and then guessed by each algorithm. For both languages the test set was split into 10 non-intersecting subsets with approximately equal numbers of grammatical functions in each, and the values obtained for the 10 subsets were further used to assess the statistical significance of the differences in the results with the paired Student's t-test.

Results. Table 2 presents the results. For both English and German all the three versions of the classifier clearly outperform the baseline, and even the advantage of **NB-CASE** over the baseline is statistically significant at the 0.5% level for both languages. However, **NB-CASE** performs much worse than **NB-N** and **NB-N&V** (their advantage over **NB-CASE** is statistically significant at the 0.5% level for both languages), confirming that CASE is not the only feature which is useful in our task. The increase in accuracy brought about by including the atomic attribute values into the feature space is visible and significant at the same level. The increase is somewhat more pronounced for German than for English. For English the inclusion of attribute values into the feature space affects primarily the accuracy of SUBJ vs. OBJ decisions. For German, the accuracy notably increases for telling SUBJ, OBJ and ADJ-GEN from one another.

Language	MF	NB-S	NB-J
English	22.0%	90.4%	91.2%
German	17.1%	81.4%	82.1%

Table 3: Experiment 2: Guessing Multiple Missing Functions. **MF** is the pick-most-frequent classifier. **NB-S** and **NB-J** are one-by-one and joint-probability-based Naive Bayesian guessers.

3.3 Experiment 2: Guessing Multiple Missing Grammatical Functions

The goal of this experiment is to assess the accuracy of the Bayesian guesser for multiple missing grammatical functions within one parent f-structure, and to compare the accuracy of one-by-one vs. joint-probability-based guessing. Our evaluation procedure models the extreme case when the functions are unknown for *all* the dependent f-structures of a particular parent.

Procedure. As a baseline, we use the same algorithm **MF** as in Experiment 1, applied to the missing grammatical functions one by one. Two Bayesian guessers are evaluated, **NB-S** guessing the missing grammatical functions one by one, and **NB-J** guessing them all at once by maximizing the joint probability of the values. Both Bayesian guessers use attribute names and values as features. All grammatical functions in the test set were used as test cases. At each step of the experiment, the grammatical functions of all the dependent f-structures of a particular parent were removed simultaneously, and then guessed with each of the algorithms considered in this experiment. Statistical significance was assessed in the same way as in Experiment 1.

Results. Table 3 presents the accuracy scores. The one-by-one guesser and the joint-probability-based guesser perform nearly equally well, resulting in accuracy levels very close to those obtained in Experiment 1 for f-structures with a single

missing function. Joint-probability-based guessing achieves an advantage which is statistically significant at the 0.5% level for both languages but is not exceeding 1% absolute improvement. For both languages errors typically occur in distinguishing OBJ vs. SUBJ and ADJUNCT vs. MOD, and additionally in XCOMP vs. OBJ for English.

3.3.1 Experiment 3: Postprocessing the Output of an MT Decoder

The goal of this experiment is to see how the method influences the results of an SMT system.

Procedure. For this experiment we use the Sulis SMT system (Graham et al., 2009), and a decoder, which selects the transfer rules by maximizing the source-to-target probability of the complete translation. Such a decoder, though simple, allows us to create a realistic environment for evaluation. From the f-structures produced by the decoder, candidate sentences are generated with XLE, and then the one best translation is selected for each sentence using a language model. The function guesser is used to postprocess the output of the decoder before sentence generation. In the experiment, the function guesser uses both attribute names and values to make a guess. Guessing of multiple missing functions is performed one-by-one, as joint guessing complicates the algorithm and leads to a very small improvement in accuracy. The function guesser is trained on 3000 sentences, which are a subset of the set used for inducing the transfer rules. The overall MT system is evaluated both with and without function guessing on 500 held-out sentences, and the quality of the translation is measured using the BLEU metric (Papineni et al., 2002). We also calculate the number of sentences for which the generator output is unempty.

Results. The system without function guesser produced results for 364 sentences out of 500, with BLEU score equal to 5.69%; with function guesser the number of successfully generated sentences increases to 433, with BLEU improving to 6.95%. Thus, the absolute increase of BLEU score brought about by the guesser is 1.24%. This suggests that the algorithm succeeds on real data and is useful in grammar-based machine translation.

4 Conclusion

In this paper we addressed the problem of restoring unknown grammatical functions in automatically generated f-structures. We proposed to view this problem as a classification task and to solve it with the Naive Bayes classifier, using the names and the values of the attributes of the dependent f-structure to construct the feature set.

The approach was evaluated on English and German data, and showed reasonable accuracy, restoring the missing functions correctly in about 91% of the cases for English and about 82% for German. It is tempting to interpret the differences in accuracy for English and German as reflecting the complexity of grammatical function assignment for the two languages. It is not clear, however, whether the differences are due to differences in the grammars or in the underlying data.

The experiments reported here use LFG-type representations. However, nothing much in the method is specific to LFG, and therefore we are confident that our method also applies to other dependency-based representations.

Acknowledgments

The research presented here was supported by Science Foundation Ireland grant 07/CE2/I1142 under the CNGL CSET programme.

References

M. Butt, H. Dyvik, T. H. King, H. Masuichi, and C. Rohrer. 2002. The parallel grammar project. In *COLING'02, Workshop on Grammar Engineering and Evaluation*.

Y. Graham, A. Bryl, and J. van Genabith. 2009. F-structure transfer-based statistical machine translation. In *LFG'09 (To Appear)*.

R. Kaplan and J. Bresnan. 1982. Lexical functional grammar, a formal system for grammatical represenation. *The Mental Representation of Grammatical Relations*, pages 173–281.

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

P. Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit X*, pages 79–86.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL'02*, pages 311–318.

S. Riezler, T. H. King, R. M. Kaplan, R. Crouch, J. T. Maxwell III, and M. Johnson. 2002. Parsing the wall street journal using a lexical-functional grammar and discriminative estimation techniques. In *ACL'02*, pages 271–278.