Large-Scale Induction and Evaluation of Lexical Resources from the Penn-II Treebank

Ruth O'Donovan, Michael Burke, Aoife Cahill, Josef van Genabith, Andy Way
National Centre for Language Technology and School of Computing
Dublin City University
Glasnevin
Dublin 9
Ireland
{rodonovan,mburke,acahill,josef,away}@computing.dcu.ie

Abstract

In this paper we present a methodology for extracting subcategorisation frames based on an automatic LFG f-structure annotation algorithm for the Penn-II Treebank. We extract abstract syntactic function-based subcategorisation frames (LFG semantic forms), traditional CFG category-based subcategorisation frames as well as mixed function/category-based frames, with or without preposition information for obliques and particle information for particle verbs. Our approach does not predefine frames, associates probabilities with frames conditional on the lemma, distinguishes between active and passive frames, and fully reflects the effects of long-distance dependencies in the source data structures. We extract 3586 verb lemmas, 14348 semantic form types (an average of 4 per lemma) with 577 frame types. We present a large-scale evaluation of the complete set of forms extracted against the full COMLEX resource.

1 Introduction

Lexical resources are crucial in the construction of wide-coverage computational systems based on modern syntactic theories (e.g. LFG, HPSG, CCG, LTAG etc.). However, as manual construction of such lexical resources is time-consuming, error-prone, expensive and rarely ever complete, it is often the case that limitations of NLP systems based on lexicalised approaches are due to bottlenecks in the lexicon component.

Given this, research on automating lexical acquisition for lexically-based NLP systems is a particularly important issue. In this paper we present an approach to automating subcategorisation frame acquisition for LFG (Kaplan and Bresnan, 1982) i.e. grammatical function-based systems. LFG has two levels of structural representation: c(onstituent)-structure, and f(unctional)-structure. LFG differentiates between governable (argument) and non-governable (adjunct) grammatical functions. Subcategorisation requirements are enforced through *semantic forms* specifying the governable grammatical functions required by a particular predicate (e.g. $\textsc{focus}\langle(\uparrow \textsc{subj})(\uparrow \textsc{obl}_{on})\rangle$). Our approach is based on earlier work on LFG semantic form extraction (van Genabith et al., 1999) and recent progress in automatically annotating the Penn-II treebank with LFG f-structures (Cahill et al., 2004b). Depending on the quality of the f-structures, reliable LFG semantic forms can then be generated quite simply by recursively reading off the subcategorisable grammatical functions for each local `pred` value at each level of embedding in the f-structures. The work reported in (van Genabith et al., 1999) was small scale (100 trees), proof of concept and required considerable manual annotation work. In this paper we show how the extraction process can be scaled to the complete Wall Street Journal (WSJ) section of the Penn-II treebank, with about 1 million words in 50,000 sentences, based on the *automatic* LFG f-structure annotation algorithm described in (Cahill et al., 2004b). In addition to extracting grammatical function-based subcategorisation frames, we also include the syntactic categories of the predicate and its subcategorised arguments, as well as additional details such as the prepositions required by obliques, and particles accompanying particle verbs. Our method does not predefine the frames to be extracted. In contrast to many other approaches, it discriminates between active and passive frames, properly reflects long distance dependencies and assigns conditional probabilities to the semantic forms associated with each predicate.

Section 2 reviews related work in the area of automatic subcategorisation frame extraction. Our methodology and its implementation are presented in Section 3. Section 4 presents the results of our lexical extraction. In Section 5 we evaluate the complete extracted lexicon against the COMLEX resource (MacLeod et al., 1994). To our knowledge, this is the largest evaluation of subcategorisation frames for English. In Section 6, we conclude and give suggestions for future work.

2 Related Work

Creating a (subcategorisation) lexicon by hand is time-consuming, error-prone, requires considerable linguistic expertise and is rarely, if ever, complete. In addition, a system incorporating a manually constructed lexicon cannot easily be adapted to specific domains. Accordingly, many researchers have attempted to construct lexicons automatically, especially for English.

(Brent, 1993) relies on local morphosyntactic cues (such as the *-ing* suffix, except where such a word follows a determiner or a preposition other than *to*) in the untagged Brown Corpus as probabilistic indicators of six different predefined subcategorisation frames. The frames do not include details of specific prepositions. (Manning, 1993) observes that Brent's recognition technique is a "rather simplistic and inadequate approach to verb detection, with a very high error rate". Manning feeds the output from a stochastic tagger into a finite state parser, and applies statistical filtering to the parsing results. He predefines 19 different subcategorisation frames, including details of prepositions. Applying this technique to approx. 4 million words of New York Times newswire, Manning acquires 4900 subcategorisation frames for 3104 verbs, an average of 1.6 per verb. (Ushioda et al., 1993) run a finite state NP parser on a POS-tagged corpus to calculate the relative frequency of just six subcategorisation verb classes. In addition, all prepositional phrases are treated as adjuncts. For 1565 tokens of 33 selected verbs, they report an accuracy rate of 83%.

(Briscoe and Carroll, 1997) observe that in the work of (Brent, 1993), (Manning, 1993) and (Ushioda et al., 1993), "the maximum number of distinct subcategorization classes recognized is sixteen, and only Ushioda *et al.* attempt to derive relative subcategorization frequency for individual predicates". In contrast, the system of (Briscoe and Carroll, 1997) distinguishes 163 verbal subcategorisation classes by means of a statistical shallow parser, a classifier of subcategorisation classes, and *a priori* estimates of the probability that any verb will be a member of those classes. More recent work by Korhonen (2002) on the filtering phase of this approach has improved results. Korhonen experiments with the use of linguistic verb classes for obtaining more accurate back-off estimates for use in hypothesis selection. Using this extended approach, the average results for 45 semantically classified test verbs evaluated against hand judgements are precision 87.1% and recall 71.2%. By comparison, the average results for 30 verbs not classified semantically are precision 78.2% and recall 58.7%.

Carroll and Rooth (1998) use a hand-written head-lexicalised context-free grammar and a text corpus to compute the probability of particular subcategorisation scenarios. The extracted frames do not contain details of prepositions.

More recently, a number of researchers have applied similar techniques to derive resources for other languages, especially German. One of these, (Schulte im Walde, 2002), induces a computational subcategorisation lexicon for over 14,000 German verbs. Using sentences of limited length, she extracts 38 distinct frame types, which contain maximally three arguments each. The frames may optionally contain details of particular prepositional use. Her evaluation on over 3000 frequently occurring verbs against the German dictionary *Duden - Das Stilwörterbuch* is similar in scale to ours and is discussed further in Section 5.

There has also been some work on extracting subcategorisation details from the Penn Treebank. (Kinyon and Prolo, 2002) introduce a tool which uses fine-grained rules to identify the arguments, including optional arguments, of each verb occurrence in the Penn Treebank, along with their syntactic functions. They manually examined the 150+ possible sequences of tags, both functional and categorial, in Penn-II and determined whether the sequence in question denoted a modifier, argument or optional argument. Arguments were then mapped to traditional syntactic functions. As they do not include an evaluation, currently it is impossible to say how effective this technique is.

(Xia et al., 2000) and (Chen and Vijay-Shanker, 2000) extract lexicalised TAGs from the Penn Treebank. Both techniques implement variations on the approaches of (Magerman, 1994) and (Collins, 1997) for the purpose of differentiating between complement and adjunct. In the case of (Xia et al., 2000), invalid elementary trees produced as a result of annotation errors in the treebank are filtered out using linguistic heuristics.

(Hockenmaier et al., 2002) outline a method for the automatic extraction of a large syntactic CCG lexicon from Penn-II. For each tree, the algorithm annotates the nodes with CCG categories in a top-down recursive manner. In order to examine the coverage of the extracted lexicon in a manner similar to (Xia et al., 2000), (Hockenmaier et al., 2002) compared the reference lexicon acquired from Sections 02-21 with a test lexicon extracted from Section 23 of the WSJ. It was found that the reference CCG lexicon contained 95.09% of the entries in the test lexicon, while 94.03% of the entries in the test TAG lexicon also occurred in the reference lexicon.

Both approaches involve extensive correction and clean-up of the treebank prior to lexical extraction.

3 Our Methodology

The first step in the application of our methodology is the production of a treebank annotated with LFG f-structure information. F-structures are feature structures which represent abstract syntactic information, approximating to basic predicate-argument-modifier structures. We utilise the automatic annotation algorithm of (Cahill et al., 2004b) to derive a version of Penn-II where each node in each tree is annotated with an LFG functional annotation (i.e. an attribute value structure equation). Trees are traversed top-down, and annotation is driven by categorial, basic configurational, trace and Penn-II functional tag information in local subtrees of mostly depth one (i.e. CFG rules). The annotation procedure is dependent on locating the head daughter, for which the scheme of (Magerman, 1994) with some changes and amendments is used. The head is annotated with the LFG equation $\uparrow=\downarrow$. Linguistic generalisations are provided over the left (the prefix) and the right (suffix) context of the head for each syntactic category occurring as the mother node of such heads. To give a simple example, the rightmost NP to the left of a VP head under an S is likely to be its subject ($\uparrow$ SUBJ $=\downarrow$), while the leftmost NP to the right of the V head of a VP is most probably its object ($\uparrow$ OBJ $=\downarrow$). (Cahill et al., 2004b) provide four sets of annotation principles, one for non-coordinate configurations, one for coordinate configurations, one for traces (long distance dependencies) and a final 'catch all and clean up' phase. Distinguishing between argument and adjunct is an inherent step in the automatic assignment of functional annotations.

The satisfactory treatment of long distance dependencies by the annotation algorithm is imperative for the extraction of accurate semantic forms. The Penn Treebank employs a rich arsenal of traces and empty productions (nodes which do not realise any lexical material) to co-index displaced material with the position where it should be interpreted semantically. The algorithm of (Cahill et al., 2004b) translates the traces into corresponding re-entrancies in the f-structure representation (Figure 1). Passive movement is also captured and expressed at f-structure level using a `passive:+` annotation. Once a treebank tree is annotated with feature structure equations by the annotation algorithm, the equations are collected and passed to a constraint solver which produces the f-structures.

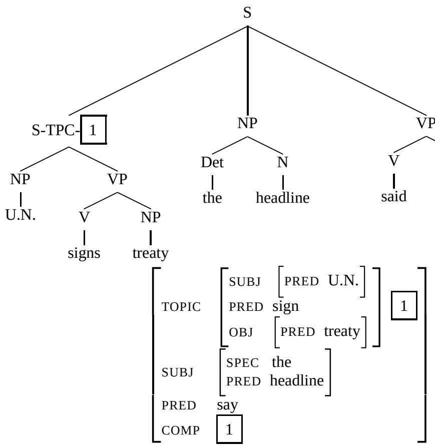

Figure 1: Penn-II style tree with long distance dependency trace and corresponding reentrancy in f-structure

In order to ensure the quality of the semantic forms extracted by our method, we must first ensure the quality of the f-structure annotations. (Cahill et al., 2004b) measure annotation quality in terms of precision and recall against manually constructed, gold-standard f-structures for 105 randomly selected trees from section 23 of the WSJ section of Penn-II. The algorithm currently achieves an F-score of 96.3% for complete f-structures and 93.6% for preds-only f-structures.[1]

Our semantic form extraction methodology is based on the procedure of (van Genabith et al., 1999): For each f-structure generated, for each level of embedding we determine the local PRED value and collect the subcategorisable grammatical functions present at that level of embedding. Consider the f-structure in Figure 1. From this we recursively extract the following non-empty semantic forms: `say([subj,comp])`, `sign([subj,obj])`. In effect, in both (van Genabith et al., 1999) and our approach semantic forms are reverse engineered from automatically generated f-structures for treebank trees. We extract the following subcategorisable syntactic functions: SUBJ, OBJ, OBJ2, OBL$_{prep}$, OBL2$_{prep}$, COMP, XCOMP and PART. Adjuncts (e.g. ADJ, APP etc) are not included in the semantic forms. PART is not a syntactic function in the strict sense but we capture the relevant co-occurrence patterns of verbs and particles in the semantic forms. Just as OBL includes the prepositional head of the PP, PART includes the actual particle which occurs e.g. `add([subj,obj,part:up])`.

In the work presented here we substantially extend the approach of (van Genabith et al., 1999) as

[1] Preds-only measures only paths ending in PRED:VALUE so features such as `number`, `person` etc are not included.

regards coverage, granularity and evaluation: First, we scale the approach of (van Genabith et al., 1999) which was proof of concept on 100 trees to the full WSJ section of the Penn-II Treebank. Second, our approach fully reflects long distance dependencies, indicated in terms of traces in the Penn-II Treebank and corresponding re-entrancies at f-structure. Third, in addition to abstract syntactic function-based subcategorisation frames we compute frames for syntactic function-CFG category pairs, both for the verbal heads and their arguments and also generate pure CFG-based subcat frames. Fourth, our method differentiates between frames captured for active or passive constructions. Fifth, our method associates conditional probabilities with frames.

In contrast to much of the work reviewed in the previous section, our system is able to produce surface syntactic as well as abstract functional subcategorisation details. To incorporate CFG details into the extracted semantic forms, we add an extra feature to the generated f-structures, the value of which is the syntactic category of the `pred` at each level of embedding. Exploiting this information, the extracted semantic form for the verb *sign* looks as follows: `sign(v,[subj(np),obj(np)])`.

We have also extended the algorithm to deal with passive voice and its effect on subcategorisation behaviour. Consider Figure 2: not taking voice into account, the algorithm extracts an intransitive frame `outlaw([subj])` for the transitive `outlaw`. To correct this, the extraction algorithm uses the feature value pair `passive:+`, which appears in the f-structure at the level of embedding of the verb in question, to mark that predicate as occurring in the passive: `outlaw([subj],p)`.

In order to estimate the likelihood of the cooccurrence of a predicate with a particular argument list, we compute conditional probabilities for subcategorisation frames based on the number of token occurrences in the corpus. Given a lemma l and an argument list s, the probability of s given l is estimated as:

$$\mathcal{P}(s|l) := \frac{count(l,s)}{\sum_{i=1}^{n} count(l,s_i)}$$

We use thresholding to filter possible error judgements by our system. Table 1 shows the attested semantic forms for the verb `accept` with their associated conditional probabilities. Note that were the distinction between active and passive not taken into account, the intransitive occurrence of `accept` would have been assigned an unmerited probability.

```
subj : spec : quant : pred : all
              adjunct : 2 : pred : almost
       adjunct : 3 : pred : remain
                     participle : pres
                 4 : obj : adjunct : 5 : pred : cancer-causing
                           pers : 3
                           pred : asbestos
                           num : sg
                     pform : of
       pers : 3
       pred : use
       num : pl
passive : +
adjunct : 1 : obj : pred : 1997
              pform : by
xcomp : subj : spec: quant : pred : all
                     adjunct : 2 : pred : almost
               ...
               ...
        passive : +
        xcomp : subj : spec: quant : pred : all
                                     adjunct : 2 : pred : almost
                       ...
                       ...
                passive : +
                pred : outlaw
                tense : past
        pred : be
pred : will
modal : +
```

Figure 2: Automatically generated f-structure for the string wsj_0003_23“`By 1997, almost all remaining uses of cancer-causing asbestos will be outlawed.`”

Semantic Form	Frequency	Probability
`accept([subj,obj])`	122	0.813
- **`accept([subj],p)`**	9	0.060
`accept([subj,comp])`	5	0.033
- **`accept([subj,obl:as],p)`**	3	0.020
`accept([subj,obj,obl:as])`	3	0.020
`accept([subj,obj,obl:from])`	3	0.020
- **`accept([subj])`**	2	0.013
`accept([subj,obj,obl:at])`	1	0.007
`accept([subj,obj,obl:for])`	1	0.007
`accept([subj,obj,xcomp])`	1	0.007

Table 1: Semantic Forms for the verb `accept` marked with `p` for passive use.

4 Results

We extract non-empty semantic forms[2] for 3586 verb lemmas and 10969 unique verbal semantic form types (lemma followed by non-empty argument list). Including prepositions associated with the OBLs and particles, this number rises to 14348, an average of 4.0 per lemma (Table 2). The number of unique frame types (without lemma) is 38 without specific prepositions and particles, 577 with (Table 3). F-structure annotations allow us to distinguish passive and active frames.

5 COMLEX Evaluation

We evaluated our induced (verbal) semantic forms against COMLEX (MacLeod et al., 1994). COM-

[2] Frames with at least one subcategorised grammatical function.

	Without Prep/Part	With Prep/Part
Sem. Form Types	10969	14348
Active	8516	11367
Passive	2453	2981

Table 2: Number of Semantic Form Types

	Without Prep/Part	With Prep/Part
# Frame Types	38	577
# Singletons	1	243
# Twice Occurring	1	84
# Occurring max. 5	7	415
# Occurring $>$ 5	31	162

Table 3: Number of Distinct Frames for Verbs (not including syntactic category for grammatical function)

LEX defines 138 distinct verb frame types without the inclusion of specific prepositions or particles. The following is a sample entry for the verb `reimburse`:

```
(VERB   :ORTH "reimburse"   :SUBC   ((NP-NP)
                                     (NP-PP :PVAL ("for"))
                                     (NP)))
```

Each verb has a :SUBC feature, specifying its subcategorisation behaviour. For example, `reimburse` can occur with two noun phrases (NP-NP), a noun phrase and a prepositional phrase headed by "for" (NP-PP :PVAL ("for")) or a single noun phrase (NP). Note that the details of the subject noun phrase are not included in COMLEX frames. Each of the complement types which make up the value of the :SUBC feature is associated with a formal frame definition which looks as follows:

```
(vp-frame   np-np   :cs   ((np 2)(np 3))
                    :gs   (:subject 1 :obj 2 :obj2 3)
                    :ex   "she asked him his name")
```

The value of the :cs feature is the constituent structure of the subcategorisation frame, which lists the syntactic CF-PSG constituents in sequence. The value of the :gs feature is the grammatical structure which indicates the functional role played by each of the CF-PSG constituents. The elements of the constituent structure are indexed, and referenced in the :gs field. This mapping between constituent structure and functional structure makes the information contained in COMLEX suitable as an evaluation standard for the LFG semantic forms which we induce.

5.1 COMLEX-LFG Mapping

We devised a common format for our induced semantic forms and those contained in COMLEX. This is summarised in Table 4. COMLEX does not distinguish between obliques and objects so we converted Obj$_i$ to OBL$_i$ as required. In addition, COMLEX does not explicitly differentiate between COMPs and XCOMPs, but does encode control information for any Comps which occur, thus allowing us to deduce the distinction automatically. The manually constructed COMLEX entries provided us with a gold standard against which we evaluated the automatically induced frames for the 2992 (active) verbs that both resources have in common.

LFG	COMLEX	Merged
SUBJ	Subject	SUBJ
OBJ	Object	OBJ
OBJ2	Obj2	OBJ2
OBL	Obj3	OBL
OBL2	Obj4	OBL2
COMP	Comp	COMP
XCOMP	Comp	XCOMP
PART	Part	PART

Table 4: COMLEX and LFG Syntactic Functions

We use the computed conditional probabilities to set a threshold to filter the selection of semantic forms. As some verbs occur less frequently than others we felt it was important to use a relative rather than absolute threshold. For a threshold of 1%, we disregard any frames with a conditional probability of less than or equal to 0.01. We carried out the evaluation in a similar way to (Schulte im Walde, 2002). The scale of our evaluation is comparable to hers. This allows us to make tentative comparisons between our respective results. The figures shown in Table 5 are the results of three different kinds of evaluation with the threshold set to 1% and 5%. The effect of the threshold increase is obvious in that Precision goes up for each of the experiments while Recall goes down.

For **Exp 1**, we excluded prepositional phrases entirely from the comparison, i.e. assumed that PPs were adjunct material (e.g. [subj,obl:for] becomes [subj]). Our results are better for Precision than for Recall compared to Schulte im Walde (*op cit.*), who reports Precision of 74.53%, Recall of 69.74% and an F-score of 72.05%.

Exp 2 includes prepositional phrases but not parameterised for particular prepositions (e.g. [subj,obl:for] becomes [subj,obl]). While our figures for Recall are again lower, our results for Precision are considerably higher than those of Schulte im Walde (*op cit.*) who recorded Precision of 60.76%, Recall of 63.91% and an F-score of 62.30%.

For **Exp. 3**, we used semantic forms which contained details of specific prepositions for any subcategorised prepositional phrase. Our Precision figures are again high (in comparison to 65.52% as recorded by (Schulte im Walde, 2002)). However,

	Threshold 1%			Threshold 5%		
	P	**R**	**F-Score**	**P**	**R**	**F-Score**
Exp. 1	79.0%	59.6%	68.0%	83.5%	54.7%	66.1%
Exp. 2	77.1%	50.4%	61.0%	81.4%	44.8%	57.8%
Exp. 2a	76.4%	44.5%	56.3%	80.9%	39.0%	52.6%
Exp. 3	73.7%	22.1%	34.0%	78.0%	18.3%	29.6%
Exp. 3a	73.3%	19.9%	31.3%	77.6%	16.2%	26.8%

Table 5: COMLEX Comparison

	Precision	**Recall**	**F-Score**
Experiment 3	81.7%	40.8%	54.4%
Experiment 3a	83.1%	35.4%	49.7%

Table 6: COMLEX Comparison using p-dir(Threshold of 1%)

Passive	**Precision**	**Recall**	**F-Score**
Experiment 2	80.2%	54.7%	65.1%
Experiment 2a	79.7%	46.2%	58.5%
Experiment 3	72.6%	33.4%	45.8%
Experiment 3a	72.3%	29.3%	41.7%

Table 7: Passive evaluation (Threshold of 1%)

our Recall is very low (compared to the 50.83% that Schulte im Walde (*op cit.*) reports). Consequently our F-score is also low (Schulte im Walde (*op cit.*) records an F-score of 57.24%). Experiments 2a and 3a are similar to Experiments 2 and 3 respectively except they include the specific particle associated with each PART.

5.1.1 Directional Prepositions

There are a number of possible reasons for our low recall scores for Experiment 3 in Table 5. It is a well-documented fact (Briscoe and Carroll, 1997) that subcategorisation frames (and their frequencies) vary across domains. We have extracted frames from one domain (the WSJ) whereas COMLEX was built using examples from the San Jose Mercury News, the Brown Corpus, several literary works from the Library of America, scientific abstracts from the U.S. Department of Energy, and the WSJ. For this reason, it is likely to contain a greater variety of subcategorisation frames than our induced lexicon. It is also possible that due to human error COMLEX contains subcategorisation frames, the validity of which may be in doubt. This is due to the fact that the aim of the COMLEX project was to construct as complete a set of subcategorisation frames as possible, even for infrequent verbs. Lexicographers were allowed to extrapolate from the citations found, a procedure which is bound to be less certain than the assignment of frames based entirely on existing examples. Our recall figure was particularly low in the case of evaluation using details of prepositions (Experiment 3). This can be accounted for by the fact that COMLEX errs on the side of overgeneration when it comes to preposition assignment. This is particularly true of directional prepositions, a list of 31 of which has been prepared and is assigned in its entirety by default to any verb which can potentially appear with any directional preposition. In a subsequent experiment, we incorporate this list of directional prepositions by default into our semantic form induction process in the same way as the creators of COMLEX have done. Table 6 shows the results of this experiment. As expected there is a significant improvement in the recall figure, being almost double the figures reported in Table 5 for Experiments 3 and 3a.

5.1.2 Passive Evaluation

Table 7 presents the results of our evaluation of the passive semantic forms we extract. It was carried out for 1422 verbs which occur with passive frames and are shared by the induced lexicon and COMLEX. As COMLEX does not provide explicit passive entries, we applied Lexical Redundancy Rules (Kaplan and Bresnan, 1982) to automatically convert the active COMLEX frames to their passive counterparts. For example, the COMLEX entry `see([subj,obj])` is converted to `see([subj])`. The resulting precision is very high, a slight increase on that for the active frames. The recall score drops for passive frames (from 54.7% to 29.3%) in a similar way to that for active frames when prepositional details are included.

5.2 Lexical Accession Rates

As well as evaluating the quality of our extracted semantic forms, we also examine the rate at which they are induced. (Charniak, 1996) and (Krotov et al., 1998) observed that treebank grammars (CFGs extracted from treebanks) are very large and grow with the size of the treebank. We were interested in discovering whether the acquisition of lexical material on the same data displays a similar propensity. Figure 3 displays the accession rates for the semantic forms induced by our method for sections 0–24 of the WSJ section of the Penn-II treebank. When we do not distinguish semantic forms by category, all semantic forms together with those for verbs display smaller accession rates than for the PCFG.

We also examined the coverage of our system in a similar way to (Hockenmaier et al., 2002). We extracted a verb-only reference lexicon from Sections 02-21 of the WSJ and subsequently compared this to a test lexicon constructed in the same way from

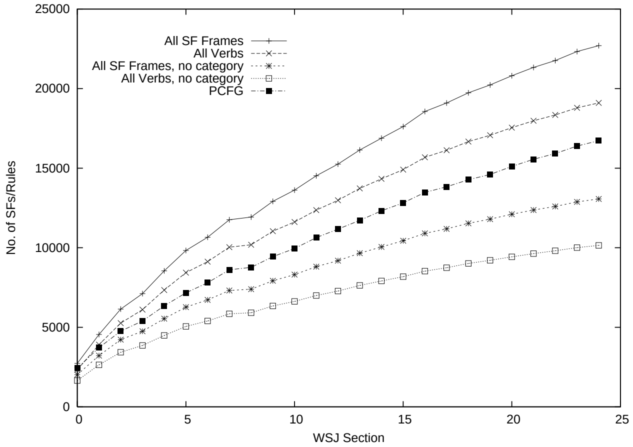

Figure 3: Accession Rates for Semantic Forms and CFG Rules

Entries also in reference lexicon:	89.89%
Entries not in reference lexicon:	10.11%
Known words:	7.85%
- Known words, known frames:	7.85%
- Known words, unknown frames:	-
Unknown words:	2.32%
- Unknown words, known frames:	2.32%
- Unknown words, unknown frames:	-

Table 8: Coverage of induced lexicon on unseen data (Verbs Only)

Section 23. Table 8 shows the results of this experiment. 89.89% of the entries in the test lexicon appeared in the reference lexicon.

6 Conclusions

We have presented an algorithm and its implementation for the extraction of semantic forms or subcategorisation frames from the Penn-II Treebank, automatically annotated with LFG f-structures. We have substantially extended an earlier approach by (van Genabith et al., 1999). The original approach was small-scale and ‘proof of concept’. We have scaled our approach to the entire WSJ Sections of Penn-II (50,000 trees). Our approach does not predefine the subcategorisation frames we extract as many other approaches do. We extract abstract syntactic function-based subcategorisation frames (LFG semantic forms), traditional CFG category-based frames as well as mixed function-category based frames. Unlike many other approaches to subcategorisation frame extraction, our system properly reflects the effects of long distance dependencies and distinguishes between active and passive frames. Finally our system associates conditional probabilities with the frames we extract. We carried out an extensive evaluation of the complete induced lexicon (not just a sample) against the full COMLEX resource. To our knowledge, this is the most extensive qualitative evaluation of subcategorisation extraction in English. The only evaluation of a similar scale is that carried out by (Schulte im Walde, 2002) for German. Our results compare well with hers. We believe our semantic forms are fine-grained and by choosing to evaluate against COMLEX we set our sights high: COMLEX is considerably more detailed than the OALD or LDOCE used for other evaluations.

Currently work is under way to extend the coverage of our acquired lexicons by applying our methodology to the Penn-III treebank, a more balanced corpus resource with a number of text genres (in addition to the WSJ sections). It is important to realise that the induction of lexical resources is part of a larger project on the acquisition of wide-coverage, robust, probabilistic, deep unification grammar resources from treebanks. We are already using the extracted semantic forms in parsing new text with robust, wide-coverage PCFG-based LFG grammar approximations automatically acquired from the f-structure annotated Penn-II treebank (Cahill et al., 2004a). We hope to be able to apply our lexical acquisition methodology beyond existing parse-annotated corpora (Penn-II and Penn-III): new text is parsed by our PCFG-based LFG approximations into f-structures from which we can then extract further semantic forms. The work reported here is part of the core component for bootstrapping this approach.

As the extraction algorithm we presented derives semantic forms at f-structure level, it is easily applied to other, even typologically different, languages. We have successfully ported our automatic annotation algorithm to the TIGER Treebank, despite German being a less configurational language than English, and extracted wide-coverage, probabilistic LFG grammar approximations and lexical resources for German (Cahill et al., 2003). Currently, we are migrating the technique to Spanish, which has freer word order than English and less morphological marking than German. Preliminary results have been very encouraging.

7 Acknowledgements

The research reported here is supported by Enterprise Ireland Basic Research Grant SC/2001/186 and an IRCSET PhD fellowship award.

References

M. Brent. 1993. From Grammar to Lexicon: Unsupervised Learning of Lexical Syntax. *Computational Linguistics*, 19(2):203–222.

E. Briscoe and J. Carroll. 1997. Automatic Extraction of Subcategorization from Corpora. In *Proceedings of the 5th ACL Conference on Applied Natural Language Processing*, pages 356–363, Washington, DC.

A. Cahill, M. Forst, M. McCarthy, R. O'Donovan, C. Rohrer, J. van Genabith, and A. Way. 2003. Treebank-Based Multilingual Unification-Grammar Development. In *Proceedings of the Workshop on Ideas and Strategies for Multilingual Grammar Development at the 15th ESSLLI*, pages 17–24, Vienna, Austria.

A. Cahill, M. Burke, R. O'Donovan, J. van Genabith, and A. Way. 2004a. Long-Distance Dependency Resolution in Automatically Acquired Wide-Coverage PCFG-Based LFG Approximations. In *Proceedings of the 42nd Annual Conference of the Association for Computational Linguistics (ACL-04)*, Barcelona, Spain.

A. Cahill, M. McCarthy, M. Burke, R. O'Donovan, J. van Genabith, and A. Way. 2004b. Evaluating Automatic F-Structure Annotation for the Penn-II Treebank. *Journal of Research on Language and Computation*.

G. Carroll and M. Rooth. 1998. Valence Induction with a Head-Lexicalised PCFG. In *Proceedings of the 3rd Conference on Empirical Methods in Natural Language Processing*, pages 36–45, Granada, Spain.

E. Charniak. 1996. Tree-bank Grammars. In *AAAI-96: Proceedings of the Thirteenth National Conference on Artificial Intelligence, MIT Press*, pages 1031–1036, Cambridge, MA.

J. Chen and K. Vijay-Shanker. 2000. Automated Extraction of TAGs from the Penn Treebank. In *Proceedings of the 38th Annual Meeting of the Association of Computational Linguistics*, pages 65–76, Hong Kong.

M. Collins. 1997. Three generative lexicalised models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 16–23.

J. Hockenmaier, G. Bierner, and J. Baldridge. 2002. Extending the Coverage of a CCG System. *Journal of Language and Computation*, (2).

R. Kaplan and J. Bresnan. 1982. Lexical Functional Grammar: A Formal System for Grammatical Representation. In Joan Bresnan, editor, *The Mental Representation of Grammatical Relations*, pages 206–250. MIT Press, Cambridge, MA, Mannheim, 8th Edition.

A. Kinyon and C. Prolo. 2002. Identifying Verb Arguments and their Syntactic Function in the Penn Treebank. In *Proceedings of the 3rd LREC Conference*, pages 1982–1987, Las Palmas, Spain.

A. Korhonen. 2002. *Subcategorization Acquisition*. PhD thesis published as Techical Report UCAM-CL-TR-530, Computer Laboratory, University of Cambridge, UK.

A. Krotov, M. Hepple, R. Gaizauskas, and Y. Wilks. 1998. Compacting the Penn Treebank Grammar. In *Proceedings of COLING-ACL'98*, pages 669–703, Montreal, Canada.

C. MacLeod, R. Grishman, and A. Meyers. 1994. The Comlex Syntax Project: The First Year. In *Proceedings of the ARPA Workshop on Human Language Technology*, pages 669–703, Princeton, NJ.

D. Magerman. 1994. *Natural Language Parsing as Statistical Pattern Recognition*. PhD Thesis, Stanford University, CA.

C. Manning. 1993. Automatic Acquisition of a Large Subcategorisation Dictionary from Corpora. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 235–242, Columbus, OH.

S. Schulte im Walde. 2002. Evaluating Verb Subcategorisation Frames learned by a German Statistical Grammar against Manual Definitions in the Duden Dictionary. In *Proceedings of the 10th EURALEX International Congress*, pages 187–197, Copenhagen, Denmark.

A. Ushioda, D. Evans, T. Gibson, and A. Waibel. 1993. The Automatic Acquisition of Frequencies of Verb Subcategorization Frames from Tagged Corpora. In *SIGLEX ACL Workshop on the Acquisition of Lexical Knowledge from Text*, pages 95–106, Columbus, OH.

J. van Genabith, A. Way, and L. Sadler. 1999. Data-driven Compilation of LFG Semantic Forms. In *EACL-99 Workshop on Linguistically Interpreted Corpora*, pages 69–76, Bergen, Norway.

F. Xia, M. Palmer, and A. Joshi. 2000. A Uniform Method of Grammar Extraction and its Applications. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2000)*, pages 53–62, Hong Kong.