

# Trebank-based Multilingual Unification-Grammar Development

Aoife Cahill, Martin Forst, Ruth O' Donovan, Christian Rohrer, Josef van Genabith, Andy Way

ESLLI 2003 Workshop on "Ideas and Strategies for Multilingual Grammar Engineering"

# Outline

- Background and Motivation
  - Why LFG?
- Automatic F-Structure Annotation of Penn II
  - Algorithm
  - Grammar Extraction and Parsing
  - Evaluation
- Automatic F-Structure Annotation of TIGER
  - Algorithm
  - Grammar Extraction and Parsing
  - Evaluation
- Conclusion

# Background and Motivation

- Large grammars are expensive and time-consuming to construct manually
- We aim to semi-automatically create wide-coverage, probabilistic, deep LFG grammar resources for various languages
- We have already created a resource for English (Penn II). Is our methodology portable to other languages/trebanks?

# Why use the LFG framework?

- Abstract representation of predicate-argument structure, independent of language-specific surface realisations.
- It provides a precise, flexible, computationally tractable and non-transformational interface between c-structure and f-structure representations.
- Much work has already been done on automatic f-structure annotation architectures.

# Example LFG

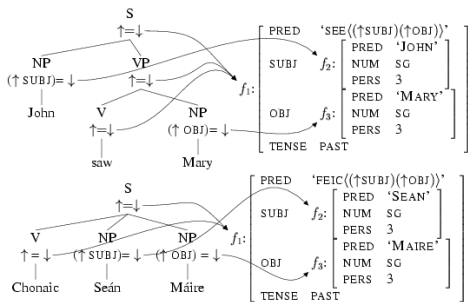
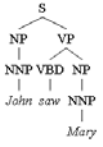


Figure 1: C- and f-structures for an English and corresponding Irish sentence

# Automatic Annotation

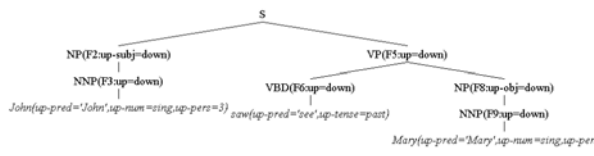
- Penn-II trebank (~50,000 sentences).
- Automatically annotate each node in the tree with f-structure information.
- Get f-structure from annotated tree.
- Extract annotated PCFG from annotated trebank.

# Automatic Annotation

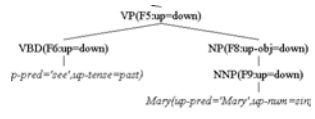
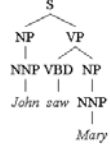


f<sub>i</sub>:

PRED	'SEE'({TRUB} {TOB})'
PREL	'JOHN'
SUBJ	f <sub>i</sub>
NUM	SG
PERS	3
PRED	'MARY'
NUM	SG
PERS	3
TENSE	PAST



# Extracting a Grammar



Rule	Probability
S → NP VP	1
NP → NNP	1
VP → VBD NP	1

Rule	Probability
S → NP(^subj=!) VP(^=!)	1
NP(^subj=!) → NNP(^=!)	1
NP(^obj=!) → NNP(^=!)	1
VP(^=!) → VBD(^=!) NP(^obj=!)	1

# Parsing with an A-PCFG



- Use Viterbi methods to always get the most probable parse
- Parser output is an f-annotated tree
- Collect f-equations from tree and use a constraint solver to produce an f-structure

# Evaluation of Automatic Annotation



- Fragmentation – i.e. how many sentences produce just 1 f-structure:
 

0 f-structure fragments	226	0.47%
1 f-structure	48141	99.41%
2 f-structure fragments	58	0.12%
- Quality (evaluated against gold standard of 105 manually annotated sentences)

All grammatical functions:	Precision	93.53
	Recall	94.69
	F-Score	94.11
Preds Only:	Precision	90.46
	Recall	91.26
	F-Score	90.86

# Evaluation of Parsing



- Train grammar on sections 02-21 of annotated Penn-II treebank
- Evaluate on Section 23 (parsing raw text)

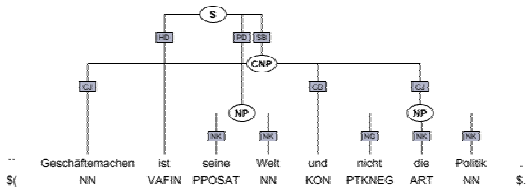
	Full Grammar	Compacted Grammar (2)
General evaluation		
# Rules	26517	7653
# Parses	2416	2407
Time (hrs)	4.91	0.87
Parse Tree evaluation		
Labelled F-score	77.12	74.39
Unlabelled F-score	79.93	76.89
F-Structure evaluation		
Fragmentation (%)	99.42	99.13
Preds Only		
F-score	70.95	70.27
Complete Match	13	13
All grammatical functions		
F-score	80.86	79.89
Complete Match	9	9

# Automatically annotating TIGER



- TIGER Treebank of German newspaper text (~36,000 sentences)
- Annotated *graphs* rather than context-free trees.
- Functional information already encoded by means of labelled edges.
- Need to convert the graphs to Penn-II style trees in order to automatically add f-structure information to the trees.

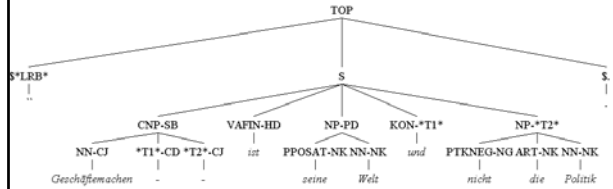
## Conversion of Graphs to Trees DCU



Business is his world, and not the politics.  
 "Business is his world, not politics."

13

## Penn-II style tree DCU



Business is his world, and not the politics.  
 "Business is his world, not politics."

14

## Automatic Annotation DCU

- Compile a lookup table for traces
- Assign a default annotation based on the edge label information (e.g. SB, OA)
- Overwrite some of the default annotations (e.g. for prepositional phrases, complementisers etc.)
- Explicitly link back trace nodes

15

## Automatic Annotation DCU

- Some default annotations  
 SB - ^subj=!  
 OA - ^obj=!  
 CC - ^obl-compar=!
- Overwriting the default annotations



16

## TIGER → Annotated Tree → F-Structure DCU

```

subj : conj : 1 : pred : 'Geschäftsmachen'
      2 : spec : det : pred : die
      adjunct : 3 : pred : nicht
      pred : 'Politik'
coord_form : und
xcomp_pred : spec : poss : pred : pro
              pred : 'Welt'
pred : sein
    
```

17

## Evaluation of Annotation DCU

- Fragmentation – i.e. how many sentences produce just 1 f-structure:

0 F-structure fragments	136	0.373%
1 F-structure	35209	96.511%
2 F-structure fragments	1053	2.886%
3 F-structure fragments	77	0.211%
4 F-structure fragments	2	0.005%
5 F-structure fragments	1	0.003%
6 F-structure fragments	1	0.003%
7 F-structure fragments	3	0.008%

- Quality (evaluated against gold standard of 100 manually corrected f-structures)

Preds Only:	Precision	93.62
	Recall	88.59
	Complete Match	25.00
	F-Score	91.03

18

- Extract an *annotated* grammar from all but sentences 8001-10000 of TIGER.
- Evaluate on remaining 2000 sentences

Full Grammar	
General evaluation	
# Rules	54066
# Parses	1992
Time (cpu)	1646
Parse Tree evaluation	
Labelled F-score	70.59
Unlabelled F-score	74.67
F-Structure evaluation	
# f-structures	1972
Fragmentation (%)	95.25
Preds Only (F-score)	71.31
Complete Match	6

- Successful development of a system that produces f-structures for English on the basis of the Penn II treebank
- Methodology does seem to be portable even to much less configurational languages such as German
  - Successful development of a parallel system for German based on the TIGER treebank
- Architecture should be portable to other languages and annotation schemes for which a treebank is available

- Still more work possible on automatic annotation
- More work to do on grammar extraction (compaction techniques, long distance dependencies,...)
- Integration of morphological information (in particular, case in German)

<http://www.computing.dcu.ie/research/nclt/demos.html>

<http://www.computing.dcu.ie/research/nclt/>