C-Structures and F-Structures for the British National Corpus

Joachim Wagner, Djamé Seddah, Jennifer Foster and Josef van Genabith

National Centre for Language Technology
School of Computing
Dublin City University
LFG 07

28th July 2007

Outline

Outline

Research Goal

Obtain LFG representations for all sentences in the British National Corpus

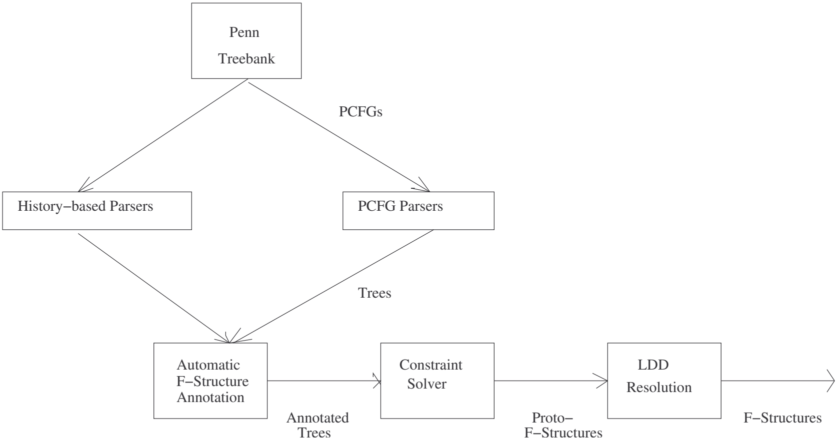
- Using LFG pipeline parsing architecture developed at DCU (Cahill *et al* 2004)
- "Scaling-up" exercise
- Domain adaptation: moving from WSJ to BNC

Previous Work

LFG Pipeline Parsing Architecture

1. C-Structures are produced by WSJ-trained statistical parsers (Bikel 2004, Charniak and Johnson 2005)
2. "Annotation Algorithm"
   2.1 C-structures nodes decorated with functional equations
   2.2 Equations solved by constraint solver to produce f-structures

LFG Pipeline Parsing Architecture



Outline

The BNC

The BNC is a one hundred million word balanced corpus of British English

- part-of-speech tagged with accuracy of 97.5%
- 90% of the BNC is written text
  - 75% factual
  - 25% fiction
- The 10% spoken component consists of
  - informal dialogue
  - business meetings
  - speeches
- encoded in SGML

Outline

C-Structure Parsing

We parse the entire BNC:

1. Charniak and Johnson re-ranking parser (June 2006)
   - lexicalized, history-based, generative statistical parser
   - discriminative re-ranker
2. Trained on WSJ Sections 2-21
3. 91.3% labelled Parseval f-score on WSJ Section 23
4. 85.2% labelled Parseval f-score on Brown Corpus
5. For our experiment, first-stage parser produces *50-best* parse trees which are re-ordered by re-ranker

C-Structure Parsing

To facilitate parsing with WSJ-trained parser:

1. British English → American English (`varcon` package)
2. Punctuation symbols used in BNC and not in Penn Treebank converted to Penn equivalents
3. SGML → UTF-8
4. US$2,000 → US$ 2,000
5. Re-insert appropriate text into anonymised ⟨*gap*⟩ tags

C-Structure Parsing

- *99.8% of sentences received a parse*
- *79.5 hours on 31 2.4GHz CPUs*
- *roughly 1.4 seconds per sentence*

length	$n = 2$	$n = 50$	rise
00–04	0.407	0.414	1.72%
05–09	0.687	0.696	1.31%
10–14	1.153	1.169	1.39%
15–19	1.914	1.946	1.67%
20–24	2.559	2.577	0.70%
25–29	3.594	3.630	1.00%
30–34	4.683	4.664	-0.41%
35–39	6.116	6.139	0.38%

C-Structure Parsing

- *99.8% of sentences received a parse*
- *79.5 hours on 31 2.4GHz CPUs*
- *roughly 1.4 seconds per sentence*

length	$n = 2$	$n = 50$	**rise**
00–04	0.407	0.414	1.72%
05–09	0.687	0.696	1.31%
10–14	1.153	1.169	1.39%
15–19	1.914	1.946	1.67%
20–24	2.559	2.577	0.70%
25–29	3.594	3.630	1.00%
30–34	4.683	4.664	-0.41%
35–39	6.116	6.139	0.38%

BNC Gold Standard

1,000 BNC sentences

- Gold standard trees serve dual purpose:
  1. Training data (future research)
  2. Test data (current research)
- Gold standard sentences are *different* from WSJ training data:
  - Contain a verb in BNC but **not** in WSJ2-21
    - 25,874 verb lemmas in BNC but not in WSJ2-21
    - 14,787 occur only once in BNC (e.g. *jitter, unfade, transpersonalize, kerplonk*)
    - 537 occur greater than 100 times (e.g. *mutter, murmur, frown, damn*)
  - Likely to represent a difficult test for WSJ-trained parsers

BNC Gold Standard

Approx. 6% of 1,000 gold standard sentences is non-standard text:

Text Type	**#**	**Example**
Dramatic	21	Tommy Johnson dribbled past the Oxford keeper
Quote	10	All the same, God damn you
Spoken	10	The seconder of formally seconded
Poem	9	Groggily somersaulting to get airborne
List Item	8	Drink something non-alcoholic to quench thirst
Caption	4	Community Personified
Headline	2	Drunk priest is nicked driving to a funeral

BNC Gold Standard

- One annotator
- Approximately 60 hours
- As references, the annotator used
  1. Penn Treebank bracketing guidelines (Bies *et al* 1995)
  2. Penn Treebank itself

BNC Gold Standard

Inconsistencies between the Penn Treebank bracketing guidelines and the Penn Treebank trees!

- The noun phrase *almost certain death* occurs in BNC gold standard sentence
- According to the guidelines, it should be annotated as *(NP (ADJP almost certain) death)*
- A search for *almost* in the Penn Treebank yields the following example *(NP almost unimaginable speed)*
- In such cases, annotator chose the analysis set out in the guidelines

BNC Gold Standard

References not enough!

- Bracketing guidelines don't cover everything
- Not always possible to find a similar example in Penn Treebank
- Example: *day in, day out*
- In such cases, the annotator chose a reasonable analysis, and documented the decision

C-Structure Results

Labelled Parseval

	LP	LR	F-Score	%100 Match
All Sentences	83.8	83.7	83.7	25.2
Less than 41 words	86.4	86.2	86.3	30.3

Unlabelled Parseval

	LP	LR	F-Score	%100 Match
All Sentences	85.5	85.4	85.4	27.9
Less than 41 words	88.2	88.0	88.1	33.5

C-Structure Results by Constituent Type

Constituent Type	Precision	Recall	F-Score
NP	86.8	88.4	87.6
VP	81.6	81.8	81.7
S	80.0	81.8	80.9
PP	80.2	82.1	81.1
SBAR	75.8	77.6	76.7
ADVP	80.3	77.4	78.8
ADJP	67.2	69.5	68.3
WHNP	91.9	96.8	94.3
PRT	61.4	84.3	71.1
WHADVP	97.3	95.5	96.4

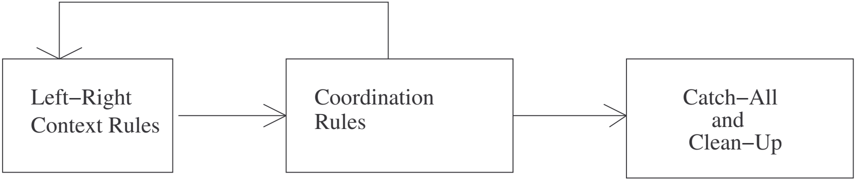
C-Structure Results by Constituent Type

Constituent Type	Precision	Recall	F-Score
NP	86.8	88.4	87.6
VP	81.6	81.8	81.7
S	80.0	81.8	80.9
PP	80.2	82.1	81.1
SBAR	75.8	77.6	76.7
ADVP	80.3	77.4	78.8
ADJP	67.2	69.5	68.3
WHNP	91.9	96.8	94.3
PRT	61.4	84.3	71.1
WHADVP	97.3	95.5	96.4

Outline

Annotation Algorithm

Three components



Annotation Algorithm

1. Left-Right Context Rules
   - Identify head, left context and right context in sub-trees of depth one
   - Certain contexts associated with particular functional equations
2. Co-ordination Rules
3. Catch-All and Clean-Up
   - Exception handling
   - Correct overgeneralizations in prior modules

Applying the Annotation Algorithm

- Faster than c-structure parsing
- C-structures annotated at the rate of 16 per second
- Failed for one sentence:
  And , and , you know , they 've got paid youth officer 's working in Harlow , now they are , there are , they 're over they 're over stretched it 's true and , but we , I mean what were doing here is actually supplementing there service and were not meeting all , we would n't of erm meeting all the demands , but the important thing I think is that were continuing to erm , you know , were trying to do something about it , and one of the things that were trying to do as officer 's in the Local Government Unit is work with Leisure Services and get them to put more resources into doing things for young people .

F-Structure Evaluation

Gold Standard F-Structures

- No hand-built set of reference f-structures
- Automatically create a set of reference f-structures
  - Apply annotation algorithm to 1,000 BNC gold standard parse trees

Procedure

1. Compare *parser-output+annotation-algorithm* f-structures to *gold+annotation-algorithm* reference f-structures
2. Compute precision, recall and f-score on f-structures as sets of term descriptions (Crouch *et al* 2002)

F-Structure Evaluation

Gold Standard F-Structures

- No hand-built set of reference f-structures
- Automatically create a set of reference f-structures
  - Apply annotation algorithm to 1,000 BNC gold standard parse trees

Procedure

1. Compare *parser-output+annotation-algorithm* f-structures to *gold+annotation-algorithm* reference f-structures
2. Compute precision, recall and f-score on f-structures as sets of term descriptions (Crouch *et al* 2002)

F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
OVERALL	91.1	91.4	91.2 (94)
PRED-ONLY	86.5	86.1	86.3 (91)
adjunct	83.3	83.6	83.4 (89)
num	96.6	97.3	96.9 (97)
pers	97.2	97.9	97.5 (98)
obj	90.1	90.4	90.2 (94)
subj	89.6	87.4	88.5 (93)
tense	97.4	96.3	96.8 (95)
det	96.5	96.4	96.4 (98)
pron_form	98.5	99.0	98.7 (99)
coord	84.0	82.1	83.0 (89)
xcomp	87.7	85.6	86.6 (91)

F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
OVERALL	91.1	91.4	91.2 (94)
PRED-ONLY	86.5	86.1	86.3 (91)
adjunct	83.3	83.6	83.4 (89)
num	96.6	97.3	96.9 (97)
pers	97.2	97.9	97.5 (98)
obj	90.1	90.4	90.2 (94)
subj	89.6	87.4	88.5 (93)
tense	97.4	96.3	96.8 (95)
det	96.5	96.4	96.4 (98)
pron_form	98.5	99.0	98.7 (99)
coord	84.0	82.1	83.0 (89)
xcomp	87.7	85.6	86.6 (91)

F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
OVERALL	91.1	91.4	91.2 (94)
PRED-ONLY	86.5	86.1	86.3 (91)
adjunct	83.3	83.6	83.4 (89)
num	96.6	97.3	96.9 (97)
pers	97.2	97.9	97.5 (98)
obj	90.1	90.4	90.2 (94)
subj	89.6	87.4	88.5 (93)
tense	97.4	96.3	96.8 (95)
det	96.5	96.4	96.4 (98)
pron_form	98.5	99.0	98.7 (99)
coord	84.0	82.1	83.0 (89)
xcomp	87.7	85.6	86.6 (91)

F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
OVERALL	91.1	91.4	91.2 (94)
PRED-ONLY	86.5	86.1	86.3 (91)
adjunct	83.3	83.6	83.4 (89)
num	96.6	97.3	96.9 (97)
pers	97.2	97.9	97.5 (98)
obj	90.1	90.4	90.2 (94)
subj	89.6	87.4	88.5 (93)
tense	97.4	96.3	96.8 (95)
det	96.5	96.4	96.4 (98)
pron_form	98.5	99.0	98.7 (99)
coord	84.0	82.1	83.0 (89)
xcomp	87.7	85.6	86.6 (91)

F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
OVERALL	91.1	91.4	91.2 (94)
PRED-ONLY	86.5	86.1	86.3 (91)
adjunct	83.3	83.6	83.4 (89)
num	96.6	97.3	96.9 (97)
pers	97.2	97.9	97.5 (98)
obj	90.1	90.4	90.2 (94)
subj	89.6	87.4	88.5 (93)
tense	97.4	96.3	96.8 (95)
det	96.5	96.4	96.4 (98)
pron_form	98.5	99.0	98.7 (99)
coord	84.0	82.1	83.0 (89)
xcomp	87.7	85.6	86.6 (91)

F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
poss	95.7	94.7	95.2 (96)
comp	72.6	73.7	73.1 (87)
topicrel	77.3	79.8	78.5 (88)
relmod	64.5	69.6	67.0 (80)
quant	87.6	82.7	85.1 (95)
obl	69.0	61.9	65.3 (71)
obl_ag	91.5	91.5	91.5 (96)
app	42.9	43.8	43.3 (86)
obj2	47.1	55.8	51.1 (71)
topic	50.0	75.0	60.0 (87)
focus	100.0	75.0	85.7 (59)
obl2	50.0	33.3	40.0 (22)

F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
poss	95.7	94.7	95.2 (96)
comp	72.6	73.7	73.1 (87)
topicrel	77.3	79.8	78.5 (88)
relmod	64.5	69.6	67.0 (80)
quant	87.6	82.7	85.1 (95)
obl	69.0	61.9	65.3 (71)
obl_ag	91.5	91.5	91.5 (96)
app	42.9	43.8	43.3 (86)
obj2	47.1	55.8	51.1 (71)
topic	50.0	75.0	60.0 (87)
focus	100.0	75.0	85.7 (59)
obl2	50.0	33.3	40.0 (22)

Example 1

The pair whirled round one another like columns of fluid

C-Structure

- Gold:
  (S (NP The pair) (VP whirled (PP round (NP one another)) (PP like (NP (NP columns) (PP of (NP fluid))))))
- Test:
  (S (NP The pair) (VP whirled (NP (NP round one another) (PP like (NP (NP columns) (PP of (NP fluid)))))))

Example 1

The pair whirled round one another like columns of fluid

C-Structure

- Gold:
  (S (NP The pair) (VP whirled (PP round (NP one another)) (PP like (NP (NP columns) (PP of (NP fluid))))))
- Test:
  (S (NP The pair) (VP whirled (NP (NP round one another) (PP like (NP (NP columns) (PP of (NP fluid)))))))

Example 1

The pair whirled round one another like columns of fluid

C-Structure

- Gold:
  (S (NP The pair) (VP whirled (PP round (NP one another)) (PP like (NP (NP columns) (PP of (NP fluid))))))
- Test:
  (S (NP The pair) (VP whirled (NP (NP round one another) (PP like (NP (NP columns) (PP of (NP fluid)))))))

Example 1

The pair whirled round one another like columns of fluid

C-Structure

- Gold:
  (S (NP The pair) (VP whirled (PP round (NP one another)) (PP like (NP (NP columns) (PP of (NP fluid))))))
- Test:
  (S (NP The pair) (VP whirled (NP (NP round one another) (PP like (NP (NP columns) (PP of (NP fluid)))))))

Example 1

The pair whirled round one another like columns of fluid

F-Structure

Gold	Test
det(one,another)	det(round,another)
adjunct(column,of)	adjunct(column,of)
obj(of,fluid)	obj(of,fluid)
adjunct(whirled,round)	obj(whirled,round)
subj(whirled,pair)	subj(whirled,pair)
obj(like,column)	obj(like,column)
adjunct(whirled,like)	adjunct(round,like)
obj(round,one)	quant(round,one)
	num(round,sg)

Example 1

The pair whirled round one another like columns of fluid

F-Structure

Gold	Test
det(one,another)	det(round,another)
adjunct(column,of)	adjunct(column,of)
obj(of,fluid)	obj(of,fluid)
adjunct(whirled,round)	obj(whirled,round)
subj(whirled,pair)	subj(whirled,pair)
obj(like,column)	obj(like,column)
adjunct(whirled,like)	adjunct(round,like)
obj(round,one)	quant(round,one)
	num(round,sg)

Example 2

They've been digging coal in small private mines in the area for centuries

C-Structure

- Gold:
  (S (NP They) (VP 've (VP been (VP (VBG digging) (NP coal) (PP in (NP (NP small private mines) (PP in (NP the area)))) (PP for (NP centuries))))))
- Test:
  (S (NP They) (VP 've (VP been (VP (NN digging) (NP coal) (PP in (NP (NP small private mines) (PP in (NP (NP the area) (PP for (NP centuries))))))))))

Example 2

They've been digging coal in small private mines in the area for centuries

C-Structure

- Gold:
  (S (NP They) (VP 've (VP been (VP (VBG digging) (NP coal) (PP in (NP (NP small private mines) (PP in (NP the area)))) (PP for (NP centuries))))))
- Test:
  (S (NP They) (VP 've (VP been (VP (NN digging) (NP coal) (PP in (NP (NP small private mines) (PP in (NP (NP the area) (PP for (NP centuries))))))))))

Example 2

They've been digging coal in small private mines in the area for centuries

C-Structure

- Gold:
  (S (NP They) (VP 've (VP been (VP (VBG digging) (NP coal) (PP in (NP (NP small private mines) (PP in (NP the area)))) (PP for (NP centuries))))))
- Test:
  (S (NP They) (VP 've (VP been (VP (NN digging) (NP coal) (PP in (NP (NP small private mines) (PP in (NP (NP the area) (PP for (NP centuries))))))))))

Example 2

They've been digging coal in small private mines in the area for centuries

F-Structure

Gold	Test
adjunct(dig,in)	adjunct(coal,in)
subj(dig,pro)	subj(coal,pro)
subj(be,pro)	subj(be,pro)
adjunct(mine,in)	adjunct(mine,in)
passive(be,+)	passive(be,+)
obj(in,area)	adjunct(area,for)
tense(be,past)	tense(be,past)
adjunct(dig,for)	num(digging,sg)
obj(dig,coal)	pers(digging,3)
xcomp(be,dig)	xcomp(be,coal)
participle(dig,pres)	obj(coal,digging)

Example 2

They've been digging coal in small private mines in the area for centuries

F-Structure

Gold	Test
adjunct(dig,in)	adjunct(coal,in)
subj(dig,pro)	subj(coal,pro)
subj(be,pro)	subj(be,pro)
adjunct(mine,in)	adjunct(mine,in)
passive(be,+)	passive(be,+)
obj(in,area)	adjunct(area,for)
tense(be,past)	tense(be,past)
adjunct(dig,for)	num(digging,sg)
obj(dig,coal)	pers(digging,3)
xcomp(be,dig)	xcomp(be,coal)
participle(dig,pres)	obj(coal,digging)

Example 2

They've been digging coal in small private mines in the area for centuries

F-Structure

Gold	Test
adjunct(dig,in)	adjunct(coal,in)
subj(dig,pro)	subj(coal,pro)
subj(be,pro)	subj(be,pro)
adjunct(mine,in)	adjunct(mine,in)
passive(be,+)	passive(be,+)
obj(in,area)	adjunct(area,for)
tense(be,past)	tense(be,past)
adjunct(dig,for)	num(digging,sg)
obj(dig,coal)	pers(digging,3)
xcomp(be,dig)	xcomp(be,coal)
participle(dig,pres)	obj(coal,digging)

Example 2

They've been digging coal in small private mines in the area for centuries

F-Structure

Gold	Test
adjunct(dig,in)	adjunct(coal,in)
subj(dig,pro)	subj(coal,pro)
subj(be,pro)	subj(be,pro)
adjunct(mine,in)	adjunct(mine,in)
passive(be,+)	passive(be,+)
obj(in,area)	adjunct(area,for)
tense(be,past)	tense(be,past)
adjunct(dig,for)	num(digging,sg)
obj(dig,coal)	pers(digging,3)
xcomp(be,dig)	xcomp(be,coal)
participle(dig,pres)	obj(coal,digging)

Example 2

They've been digging coal in small private mines in the area for centuries

F-Structure

Gold	Test
adjunct(dig,in)	adjunct(coal,in)
subj(dig,pro)	subj(coal,pro)
subj(be,pro)	subj(be,pro)
adjunct(mine,in)	adjunct(mine,in)
passive(be,+)	passive(be,+)
obj(in,area)	adjunct(area,for)
tense(be,past)	tense(be,past)
adjunct(dig,for)	num(digging,sg)
obj(dig,coal)	pers(digging,3)
xcomp(be,dig)	xcomp(be,coal)
participle(dig,pres)	obj(coal,digging)

Example 3

Grey-haired, stooping, shabbily dressed

C-Structure

- Gold:
  (FRAG (ADJP (ADJP Grey-haired), (ADJP stooping), (ADJP (ADVP shabbily) dressed)))
- Test:
  (FRAG (ADJP Grey-haired , stooping) , (VP (ADVP shabbily) dressed))

Example 3

Grey-haired, stooping, shabbily dressed

F-Structure

Gold	Test
tense(dress,past)	tense(dress,past)
adjunct(dress,shabbily)	adjunct(dress,shabbily)
adjunct(dress,grey-haired)	adjunct(stooping,grey-haired)
adjunct(dress,stooping)	
participle(stooping,pres)	

Example 3

Grey-haired, stooping, shabbily dressed

F-Structure

Gold	Test
tense(dress,past)	tense(dress,past)
adjunct(dress,shabbily)	adjunct(dress,shabbily)
adjunct(dress,grey-haired)	adjunct(stooping,grey-haired)
adjunct(dress,stooping)	
participle(stooping,pres)	

Example 3

Grey-haired, stooping, shabbily dressed

F-Structure

Gold	Test
tense(dress,past)	tense(dress,past)
adjunct(dress,shabbily)	adjunct(dress,shabbily)
adjunct(dress,grey-haired)	adjunct(stooping,grey-haired)
adjunct(dress,stooping)	
participle(stooping,pres)	

Outline

Domain adaptation and self-training

The problem of domain adaptation in parsing can be solved by training a parser on parse trees from the new domain

But where do we get these parse trees?

Self-training *is a potential solution to the domain adaptation problem:*

- Training a parser on its own output
- Parser provides its own training material

Domain adaptation and self-training

The problem of domain adaptation in parsing can be solved by training a parser on parse trees from the new domain

But where do we get these parse trees?

Self-training *is a potential solution to the domain adaptation problem:*

- Training a parser on its own output
- Parser provides its own training material

Domain adaptation and self-training

The problem of domain adaptation in parsing can be solved by training a parser on parse trees from the new domain

But where do we get these parse trees?

Self-training *is a potential solution to the domain adaptation problem:*

- Training a parser on its own output
- Parser provides its own training material

Successful Self-Training Experiments

McClosky et al (2006) showed that self-training can be used effectively with the re-ranking parser of Charniak and Johnson

- Parse sentences from the North American News Corpus (NANC) with re-ranking parser
- Train first-stage parser on the NANC parse trees
- Improved accuracy on WSJ *and* Brown Corpus

We repeat their experiment with the BNC instead of NANC

- 1.7% c-structure improvement (Foster *et al* 2007)
- Applied annotation algorithm to output of BNC self-trained parser
- F-score increase: 91.2% $\rightarrow$ 91.7%

Successful Self-Training Experiments

McClosky et al (2006) showed that self-training can be used effectively with the re-ranking parser of Charniak and Johnson

- Parse sentences from the North American News Corpus (NANC) with re-ranking parser
- Train first-stage parser on the NANC parse trees
- Improved accuracy on WSJ *and* Brown Corpus

We repeat their experiment with the BNC instead of NANC

- 1.7% c-structure improvement (Foster *et al* 2007)
- Applied annotation algorithm to output of BNC self-trained parser
- F-score increase: 91.2% → 91.7%

Successful Self-Training Experiments

McClosky et al (2006) showed that self-training can be used effectively with the re-ranking parser of Charniak and Johnson

- Parse sentences from the North American News Corpus (NANC) with re-ranking parser
- Train first-stage parser on the NANC parse trees
- Improved accuracy on WSJ *and* Brown Corpus

We repeat their experiment with the BNC instead of NANC

- 1.7% c-structure improvement (Foster *et al* 2007)
- Applied annotation algorithm to output of BNC self-trained parser
- F-score increase: 91.2% → 91.7%

Outline

Summary

1. Application of DCU LFG parsing technology to BNC
2. Evaluation at both c-structure and f-structure level
3. Room for improvement but... our research demonstrates that it is feasible to provide a reasonably accurate LFG analysis of a very large body of sentences in a robust, non-labour-intensive way.

Summary

1. Application of DCU LFG parsing technology to BNC
2. Evaluation at both c-structure and f-structure level
3. Room for improvement but... our research demonstrates that it is feasible to provide a reasonably accurate LFG analysis of a very large body of sentences in a robust, non-labour-intensive way.

Future Work

1. Hand-built reference f-structures
2. More sophisticated self-training
3. Improve annotation algorithm
4. Use automatically acquired f-structures for subcat frame extraction

Thank you for listening!

References

A. Bies, M. Ferguson, K. Katz, and R. MacIntyre.
Bracketing Guidelines for Treebank II Style, Penn Treebank Project.
Technical Report MS-CIS-95-06, University of Pennsylvania, 1995.

A. Cahill.
Parsing with Automatically Acquired, Wide-Coverage, Robust, Probabilistic LFG Approximations.
PhD thesis, Dublin City University, Dublin, 2004.

E. Charniak and M. Johnson.
Course-to-fine N-best-parsing and Maxent Discriminative Reranking.
In *Proceedings of the 43rd Meeting of the ACL*, 173–180, 2005.

R. Crouch, R. Kaplan, T. H. King, and S. Riezler.
A Comparison of Evaluation Metrics for a Broad-coverage Parser.
In *Proceedings of the "Beyond Parseval - Towards Improved Evaluation Measures for Parsing Systems" LREC-02 Workshop*, 67–74, 2002.

J. Foster, J. Wagner,D. Seddah and J. vanGenabith.
Adapting WSJ-Trained Parsers to the British National Corpus using In-domain Self-training.
In *Proceedings of the 10th IWPT*, 33–36,2007.

D. McClosky, E. Charniak, and M. Johnson.
Reranking and Self-training for Parser Adaptation.
In *Proceedings of the 21st COLING and 44th ACL*, 337–344,2006.

WSJ23 F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
OVERALL	94.5	94.0	94.3 (91.2)
PRED-ONLY	91.6	90.7	91.1 (86.3)
num	97	97	97 (97)
pers	98	98	98 (98)
adjunct	89	88	89 (83)
obj	94	94	94 (90)
subj	94	92	93 (89)
tense	95	96	95 (97)
det	97	98	98 (96)
xcomp	93	90	91 (87)
coord	89	89	89 (83)
pron_form	99	99	99 (99)

WSJ23 F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
OVERALL	94.5	94.0	94.3 (91.2)
PRED-ONLY	91.6	90.7	91.1 (86.3)
num	97	97	97 (97)
pers	98	98	98 (98)
adjunct	89	88	89 (83)
obj	94	94	94 (90)
subj	94	92	93 (89)
tense	95	96	95 (97)
det	97	98	98 (96)
xcomp	93	90	91 (87)
coord	89	89	89 (83)
pron_form	99	99	99 (99)

WSJ23 F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
OVERALL	94.5	94.0	94.3 (91.2)
PRED-ONLY	91.6	90.7	91.1 (86.3)
num	97	97	97 (97)
pers	98	98	98 (98)
adjunct	89	88	89 (83)
obj	94	94	94 (90)
subj	94	92	93 (89)
tense	95	96	95 (97)
det	97	98	98 (96)
xcomp	93	90	91 (87)
coord	89	89	89 (83)
pron_form	99	99	99 (99)

WSJ23 F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
comp	89	85	87 (73)
quant	96	95	95 (85)
poss	96	97	96 (95)
topicrel	90	85	88 (79)
relmod	82	78	80 (67)
app	85	88	86 (43)
obl	75	68	71 (65)
topic	89	84	87 (60)
obl_ag	95	97	96 (92)
obj2	66	78	71 (51)
focus	71	50	59 (86)
obl2	33	17	22 (40)

WSJ23 F-Structure Evaluation Results

Attribute	Precision	Recall	F-Score
comp	89	85	87 (73)
quant	96	95	95 (85)
poss	96	97	96 (95)
topicrel	90	85	88 (79)
relmod	82	78	80 (67)
app	85	88	86 (43)
obl	75	68	71 (65)
topic	89	84	87 (60)
obl_ag	95	97	96 (92)
obj2	66	78	71 (51)
focus	71	50	59 (86)
obl2	33	17	22 (40)