STRONG DOMAIN VARIATION AND TREEBANK-INDUCED LFG RESOURCES

John Judge, Michael Burke, Aoife Cahill, Ruth O'Donovan, Josef van Genabith, and Andy Way
National Centre for Language Technology and School of Computing,
Dublin City University, Dublin, Ireland

Proceedings of the LFG05 Conference
University of Bergen
Miriam Butt and Tracy Holloway King (Editors)
2005
CSLI Publications
http://csli-publications.stanford.edu/

Abstract

In this paper we present a number of experiments to test the portability of existing treebank-induced LFG resources. We test the LFG parsing resources of Cahill et al. (2004) on the ATIS corpus which represents a considerabley different domain to the Penn-II Treebank Wall Street Journal sections, from which the resources were induced. This testing shows an underperformance at both c- and f-structure level as a result of the domain variation. We show that in order to adapt the LFG resources of Cahill et al. (2004) to this new domain, all that is necessary is to retrain the c-structure parser on data from the new domain.

1 Introduction

Probabilistic, treebank-based parsing resources (Collins, 1999; Charniak, 2000; Bikel, 2002) are of high quality and can be rapidly induced from appropriate treebank material. However, treebank- and machine learning-based grammatical resources reflect the characteristics of the training data. They generally underperform on test data substantially different from the training data. In this paper we investigate the effects of strong domain variation on the treebank-induced, “deep”, probabilistic Lexical-Functional Grammar resources of Cahill et al. (2004) and show how these resources can be adapted to handle strong domain variation. In our experiments, we use the Penn-II treebank (Marcus et al., 1994) Wall Street Journal (WSJ) newspaper sections and the ATIS (Hemphill et al., 1990) transcribed spoken language airline reservation resource. The Penn-II WSJ vs. ATIS domain change results in a markedly stronger drop in performance, both on the trees and the f-structures, for the Penn-II trained LFG resources of Cahill et al. (2004), compared to the drop observed by Gildea (2001) for the Penn-II WSJ vs. Brown domain variation experiments with Collins’s (1997) parser.

This poses a research question: is the observed performance drop of the LFG resources of Cahill et al. (2004) due to the decrease in quality of c-structure parsing, or is it a lack of coverage of the f-structure annotation algorithm (ibid.), or both? We report on experiments which answer this question. The main, and surprising, result is that, while the Penn-II trained c-structure component of Cahill et al. (2004) requires retraining, the f-structure annotation algorithm (originally designed for Penn-II WSJ data) requires no changes or extensions. The linguistic information encoded in the f-structure annotation algorithm is already complete with respect to strong domain variation as exemplified between the Penn-II WSJ and ATIS corpora. This is a surprising result as Penn-II WSJ data represents a markedly different text domain to that of ATIS, as discussed in Section 3. A possible explanation is that, compared to c-structure, f-structure is a more abstract and “normalised” level of representation in the LFG architecture, less affected by domain variation than c-structure.

Section 2 gives a brief outline of related work on treebank induced resources. In Section 3, we compare and contrast the ATIS corpus with the WSJ sections from the Penn-II Treebank. We outline our baseline experiments and present the results in Section 4. We analyse the results, investigate the underperformance and present experiments to improve performance in Sections 5 and 6. We investigate retraining the c-structure parser with appropriate data. In a CCG-style experiment with the retrained parser we achieve a c-structure labelled f-score of 86.07 and an f-structure all grammatical functions f-score of 88.11. This constitutes an improvement of over 14% on c-structure parsing, and over 7% on f-structure annotation compared to unadapted parsing and annotation with the same system. In some additional experiments we parameterise the amount of WSJ material in the parser's training set. We then measure the effect of adding punctuation to the ATIS test set and assess the question/non-question performance of the parser and annotation algorithm and perform a back-testing experiment with the retrained resources.

2 Background Work and Motivation

Wide coverage parsers are now being used for question analysis in open-domain question answering (QA) systems as described in Pasca and Harabagiu (2001) for example. In ongoing work we are investigating the use of the LFG annotation algorithm of Cahill et al. (2004) with Bikel's (2002) parser to analyse TREC[1] question material into f-structures to develop a question tree- and f-structure bank resource for developing QA systems.

2.1 Previous Work

Domain variation and its effects on "shallow" [2] probabilistic parser performance has been investigated by Gildea (2001). For example, training on the Penn-II Treebank WSJ sections and parsing Brown corpus text resulted in a drop in labelled bracketing f-score for trees of 5.7% compared to parsing the WSJ. This shows the negative effect of domain variation on parser performance even when the test data is not substantially different from the training data (both the Penn II and Brown corpora consist primarily of written texts of American English, the main difference is the considerably more varied nature of the text in the Brown corpus). Gildea also shows how to resolve this problem by adding appropriate data to the training corpus, but notes that a large amount of additional data makes little impact if it is not matched to the test material.

Clark et al. (2004) have worked specifically with question parsing to generate dependencies for QA with Penn-II treebank based Combinatory Categorial Grammars (CCG's). In their work they focus on "what" questions taken from the TRECQA dataset. Their solution is to retrain the lexical annotation component (the supertagger) of the parser rather than the whole parser. They evaluate accuracy at the lexical category level. In their work the supertagger's accuracy improves over 13% with retraining on appropriate data. This gives a good indication of what can be achieved by retraining resources for questions.

Burke et al. (2004), Cahill et al. (2004), and O'Donovan et al. (2004) present a substantial body of work on automatically producing LFG resources from treebanks. However, to date no previous

[1] http://www.trec.nist.gov

[2] A "shallow" grammar defines a language as a set of strings and may associate syntactic representations with strings. A "deep" grammar (in addition) associates strings with information/meaning representations, usually in the form of predicate-argument structures, dependency relations or logical forms. In order to construct accurate and complete "meaning" representations, deep grammars usually resolve long-distance dependencies.

research has been carried out to test the effect of domain variance on the treebank-induced LFG parsing resources of Cahill et al. (2004). Given that the resources are induced from the Penn-II Treebank, the expectation is that performance will suffer in a similar way as the experiments of Gildea with Collins' (1997) parser showed. In Section 4, we present experiments to test this hypothesis on the ATIS corpus, which contains transcribed spoken language with a significant proportion of question material and constitutes an instance of strong domain variation.

3 Corpus Description

3.1 ATIS

The Air Travel Information System (ATIS) corpus (Hemphill et al., 1990) is a transcription of spoken dialog with an automated air travel information system. ATIS represents a different style of language from the Wall Street Journal texts of the Penn-II Treebank: a significant proportion of the sentences in ATIS are questions, imperatives and non-sentential utterances, which are generally shorter than those in the WSJ sections of Penn-II and the transcription does not contain punctuation marks.

1. Are there any flights arriving after eleven a.m

2. Show me the T W A flight

3. I need a flight from Los Angeles to Charlotte today

4. Flights from Los Angeles to Pittsburgh

5. On Tuesday arriving before five p.m

6. What flights from Philadelphia to Atlanta

Figure 1: Example ATIS utterances

Figure 1 illustrates typical ATIS corpus data including both question (1) and non-question sentences (2,3), as well as sub-sentential (4,5) and incomplete utterances (6). Note also, that punctuation has not been added.

3.2 Penn-II WSJ vs. ATIS

	ATIS	Penn-II WSJ
Words	4000 words	1 Million words
Sentences	578 sentences	50,000 sentences
Average sentence length	7 words	21 words
Source	Transcription of spoken dialog	WSJ Newspaper text
#Questions	213 Direct questions	233 Direct questions
Sentence type	Interrogatives, imperatives, and fragments	Declarative sentences
Inter-Word Punctuation	None	Punctuated

Table 1: Corpus statistics compared

Both Penn-II WSJ and ATIS are POS- and parse-annotated corpora (ie. treebanks) following the same general annotation guidelines (Bies et al., 1995). Despite these similarities, the two treebanks exhibit strong differences as regards size, domain, phrase type distribution and punctuation.

Table 1 shows a comparison of the Penn-II WSJ sections and the ATIS corpus. The most striking difference between the Penn-II Treebank WSJ sections and the ATIS is the difference in size between the two corpora: the WSJ sections of the Penn-II Treebank with 50,000 sentences are over eighty times the size of ATIS with only 578 sentences. Another important difference between the two is in the average sentence length, those in ATIS tend to be much shorter than the WSJ, with an average length of 7 words, compared to 21 words in the WSJ. Figure 2 plots the number of sentences against the sentence length for the ATIS corpus and Section 23 of the WSJ section of the Penn-II treebank illustrating the difference in sentence length distribution between the corpora.

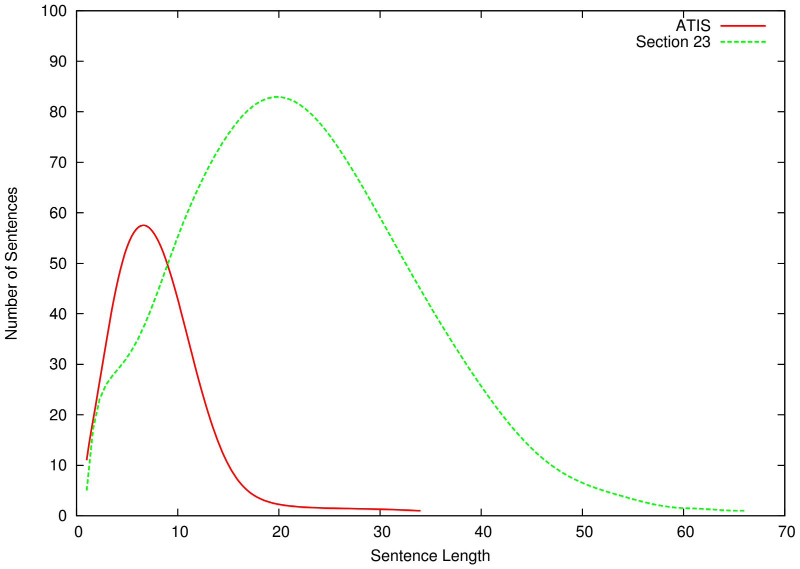

Figure 2: Sentence length distributions ATIS vs WSJ Section 23

The graph shows how significantly larger a single section of the Penn-II Treebank WSJ sections is than ATIS. It also shows the broader distribution of data over the sentence lengths in the section of the Penn-II Treebank, which has a much wider spread over the sentence lengths. Section 23 has a mean sentence length of 21 words with a standard deviation of 8.6, while ATIS has a mean sentence length of 7 words with a standard deviation of 2.9.

The source of text for the two corpora also highlights some important differences. The source for the ATIS corpus is spoken dialogue which tends to be more casual and brief (Figure 1) than the longer, more complex structures found in the Penn-II Treebank (Figure 3). Also the nature of the air travel information system results in the ATIS corpus containing sentences of a predominantly interrogative nature. Of the 578 sentences in the ATIS corpus, 213 are questions, accounting for over 36% of the entire corpus. Comparatively, the WSJ has very few interrogative sentences or questions, only 233 over the entire WSJ sections (accounting for less than a half of a percent of the corpus). In addition, many of these are embedded or rhetorical questions (Figure 4 (3)), which unlike those in the ATIS do not seek information. None of the 233 questions in the WSJ sections are to be found in section 23 of the treebank, which is the standard testing section for parser evaluation. Therefore, none of the evaluations carried out on this section reflect the quality of parsing/annotation of question data.

1. Shares of UAL, the parent of United Airlines, were extremely active all day Friday, reacting to news and rumors about the proposed $6.79 billion buy-out of the airline by an employee-management group.
2. Ports of Call Inc. reached agreements to sell its remaining seven aircraft to buyers that weren't disclosed.
3. As a group, stock funds held 10.2% of assets in cash as of August, the latest figures available from the Investment Company Institute.

Figure 3: Example Penn-II Treebank WSJ sentences

1. For example, what exactly did the CIA tell Major Giroldi and his fellow coup plotters about U.S. laws and executive orders on assassinations?
2. Who'd have thought that the next group of tough guys carrying around reputations like this would be school superintendents?
3. What is the way forward?
4. But if rational science and economics have nothing to do with the new environment initiative, what is going on?

Figure 4: Example Penn-II Treebank WSJ questions

4 Preliminary Experiments and Results

4.1 Baseline Resources

This section describes our baseline experiments to determine the portability of the resources of Cahill et al. (2004) to a new domain, the ATIS corpus.

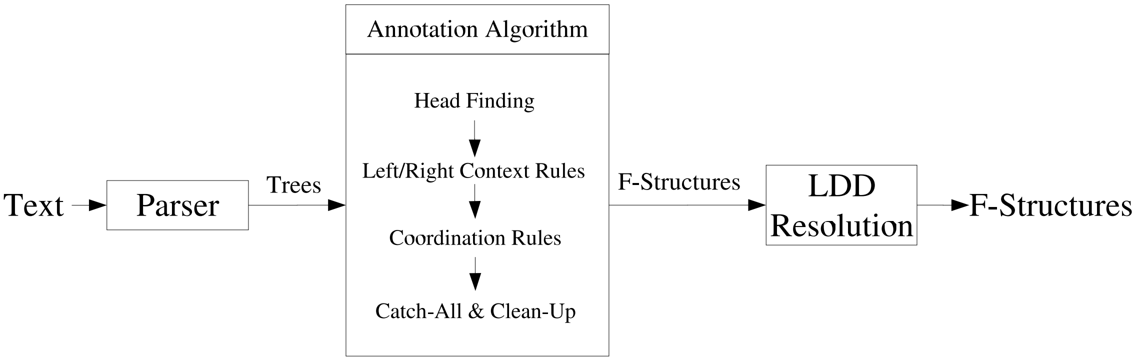

Figure 5: Pipeline Architecture

We use the pipeline model of Cahill et al. (2004) (Figure. 5) to generate f-structures from raw text. The c-structure parser used is that of Bikel (2002) which emulates Collins' (1999) model 2 parser. The grammar used by the parser is trained on sections 2-21 of the Penn-II Treebank. The f-structure annotation algorithm (also developed on Penn-II WSJ material) is modular, taking c-structure trees and automatically adding LFG f-structure equations to each node in the tree. A modified version of Magerman's (1994) scheme is used for determining the head of each subtree. The first module of the algorithm (Left-Right Context Rules) assigns annotations to the tree nodes based on whether they occur to the left or right of the head. Since the analysis of co-ordination in the Penn-II Treebank is very flat, co-ordination is treated separately in order to keep the left-right context rules concise. In the "Catch-All and Clean-Up" module of the algorithm, overgeneralisations made by the previous modules are corrected. The three modules generate "proto" f-structures which are then passed to a post-annotation long distance dependency (LDD) resolution module, which resolves long distance dependencies and outputs the final "proper" f-structures which we evaluate.

4.2 Evaluation

We use the pipeline architecture shown in Figure 5 to generate c- and f-structures from raw strings taken from the ATIS corpus. We evaluate both the c-structure trees outputted by the parser using PARSEVAL metrics (Black et al., 1991), and the LDD-resolved f-structures output by the annotation algorithm using the triple encoding and evaluation software of Crouch et al. (2002). The parser output is evaluated against the parse trees in the ATIS corpus, and the f-structures are evaluated against a hand crafted gold standard of f-structures for 100 sentences randomly selected from the ATIS corpus. We also perform a CCG-style (Hockenmaier, 2003) evaluation whereby we generate f-structures for the entire ATIS corpus from the original ATIS treebank trees and evaluate f-structures generated from the parser output against these 578 pseudo gold standard f-structures.

4.3 Results

(a)

100 Gold Standard		Precision	Recall	F-Score
Trees (labelled bracketing)		73.77	67.05	70.25
F-Structures	All GFs	82.17	67.41	74.06
	Preds-only	70.33	56.97	62.95

(b)

578 ATIS		Precision	Recall	F-Score
Trees (labelled bracketing)		75.49	67.77	71.42
F-Structures	All GFs	81.23	80.29	80.76
	Preds-only	69.27	67.02	68.13

(c)

DCU 105		Precision	Recall	F-Score
Trees (labelled bracketing)		86.56	85.59	86.07
F-Structures	All GFs	83.45	78.95	81.14
	Preds-Only	76.32	72.0	74.10

(c)

Table 2: Results for baseline experiments

Table 2 gives the results for the two evaluations described above. Table 2 (a) shows the evaluation against the 100 sentence ATIS hand-crafted f-structure gold standard. Compared to the most recent results for the Penn-II WSJ section 23 based DCU 105[3] evaluation in Table 2(c), the treebank-based LFG parsing resources of Cahill et al. (2004) show a significant drop in both the tree- and f-structure-based analysis scores for the ATIS material. The c-structures output by the parser have an f-score around 16% less than in the in-domain (section 23) evaluation for the same parser/grammar combination (Bikel trained on sections 02-21 of the Penn-II Treebank). Likewise the f-structure evaluation has suffered, with the preds-only f-score over 11% lower than on in-domain data.

[3] http://nclt.dcu.ie/gold105.txt

Dependency	Precision	Recall	F-Score
adjunct	159/258=62	159/353=49	55
comp	0/5=0	0/3=0	0
coord	15/23=65	15/24=62	64
det	56/64=88	56/70=80	84
focus	**9/9=100**	**9/33=27**	**43**
obj	172/206=83	172/216=80	82
obj2	17/18=94	17/18=94	94
obl	1/2=50	1/12=8	14
obl2	0/0=0	0/5=0	0
poss	1/1=100	1/1=100	100
quant	2/16=12	2/6=33	18
relmod	9/13=69	9/16=56	62
subj	10/27=37	10/17=59	54
topicrel	**10/27=37**	**10/17=59**	**45**
xcomp	23/33=70	23/46=50	58

Table 3: Annotation results for selected features

Table 3 shows a more detailed analysis of the f-structure evaluation in Table 2(a) for selected features. The table shows that in particular for features such as **focus** and **topicrel**, which are important to analyse correctly in questions, the performance is quite low. This indicates that, as it stands, the Penn-II treebank-based LFG parsing system is not well suited to analysing questions and performance has suffered substantially as a result of the change in domain.

We have seen that by changing the domain from WSJ text to ATIS, the overall performance for c-structure analysis and f-structure analysis has dropped significantly. The strong domain variance between ATIS and WSJ data has affected both shallow (c-structure trees) and deep (f-structure dependencies) analyses and is more pronounced than was observed in earlier work by Gildea (2001).[4]

5 Why the Performance Drop?

The drop in performance can be attributed to the domain variance, but the question remains which module in the pipeline parsing architecture in Figure 5 (c-structure parser, f-structure annotation algorithm or LDD resolution) is underperforming due to the change in domain, or is it a combination? We can narrow the possibilities down to two of the three modules shown in Figure 5.[5] Either the c-structure parser is underperforming and consequently the annotation algorithm is unable to generate sufficiently good f-structures from the bad c-structures, or the annotation algorithm is incomplete with respect to the domain variance.

[4] Gildea's work focused on c-structure parsing as opposed to full LFG f-structures.

[5] Testing on the long distance dependency resolution module showed that problems with LDD resolution were directly related to bad c-structure parsing.

The results in Table 2 have shown that the c-structure parser performance has dropped by almost 16% as a result of the domain variance. Previous work has shown that parser performance can be boosted through retraining with appropriate data (Gildea, 2001; Clark et al., 2004). We carry out an experiment to try and boost the question domain performance of Bikel's parser by retraining a grammar with appropriate material from the ATIS corpus.

6 Retraining Experiments and Results

6.1 Retraining (WSJ + ATIS)

In order to improve the performance of the c-structure parser on ATIS sentences we create a new training set from which to extract a grammar for the parser. This new, larger, training set consists of sections 02-21 of the Penn-II Treebank WSJ (the original training data) and 90% of the ATIS corpus. We then train the parser on this new training set, and repeat the parsing and annotation experiments outlined in Section 4. C-structures for each of the 578 ATIS sentences are generated by retraining a grammar and parsing using a 10-fold cross-validation experiment with a 90%:10% training:test split over the ATIS corpus, and adding the 90% ATIS split to sections 02-21 of the Penn-II Treebank WSJ for training. The parser output c-structures are then passed to the f-structure annotation algorithm and LDD-resolution and the f-structures evaluated as before.

(a)

100 Gold Standard		Precision	Recall	F-Score	Diff
Trees (labelled bracketing)		88.03	78.78	83.14	+12.89
F-Structures	All GFs	88.04	79.10	83.33	+9.27
	Preds-only	80.17	73.66	76.77	+13.82

(b)

578 ATIS		Precision	Recall	F-Score	Diff
Trees (labelled bracketing)		80.66	92.26	86.07	+14.65
F-Structures	All GFs	87.27	88.97	88.11	+7.35
	Preds-only	80.21	80.81	80.51	+12.38

Table 4: Results for experiments with retrained grammar for 10-fold cross validation

Tables 4 (a) and (b) give the results of evaluating c-structures and f-structures generated with Bikel's parser retrained as described above. Evaluating against the 100-sentence ATIS gold standard, the c-structure f-score has increased by almost 13% to 83.14. The quality of the f-structures has also increased with an improvement of almost 14% in the preds-only f-score, to 76.77. The performance over the whole corpus, in a CCG-style experiment against automatically generated f-structures for the original 578 treebank trees, has increased correspondingly, with the c-structure f-score increasing over 14% to 86.07, and a preds only evaluation of the f-structures gaining over 12% to achieve an f-score of 80.51.

Dependency	Precision	Recall	F-Score	Diff
adjunct	229/292=78	229/324=71	74	+19
comp	0/4=0	0/3=0	0	-
coord	16/24=67	16/24=67	67	+3
det	67/66=92	61/70=87	90	+6
focus	**23/23=100**	**23/33=70**	**82**	**+39**
obj	193/223=87	193/216=89	88	+6
obj2	17/17=100	17/18=94	97	+3
obl	1/1=100	1/12=8	15	+1
obl2	0/0=0	0/5=0	0	-
poss	1/1=100	1/1=100	100	-
quant	2/16=12	2/6=33	18	-
relmod	14/19=74	14/16=88	80	+18
subj	75/89=84	75/133=56	68	+14
topicrel	**14/19=74**	**14/17=82**	**78**	**+33**
xcomp	25/30=83	25/46=54	66	+12

Table 5: Annotation results for selected features

Table 5 shows a more detailed analysis of the evaluations in Table 4(a) for a number of features. Compared to Table 3 the table shows that the retraining has had no negative effect on any of the features. The majority of features have improved in terms of both precision and recall. Of those features which benefited from the retraining, two features have gained significantly more than the others, **focus** and **topicrel**. These are two features which are important for analysing questions correctly.

Our experiments so far indicate that the annotation algorithm of Cahill et al. (2004), Burke et al. (2004), and O'Donovan et al. (2004) is complete with respect to the strong domain variance encountered in our experiments. We have seen that in order to cope with a new domain only the c-structure parser needs to be retrained.

In order to estimate an upper bound for our experiments, we took the original ATIS treebank trees for the 100 sentences in the gold standard and automatically annotated them to produce f-structures, thereby removing the c-structure parser margin of error. We then evaluated these f-structures against the hand-crafted f-structures in the gold standard. In this evaluation the all grammatical functions f-score is 92.80 and the preds-only f-score is 89.88 (Table 6). This is a satisfactory upper bound and the results are comparable to a similar experiment on the DCU 105.

	All GFs	Preds-only
F-Score	92.80	89.88

Table 6: Upper bound for gold standard trees

These results demonstrate that improving the c-structure parsing is sufficient to improve the overall performance of the annotation algorithm on sentences outside of the domain on which it was developed. This is quite a surprising result, as we did not modify the annotation algorithm of Burke et al. (2004) in any way.

6.2 Parameterisation of Penn-II WSJ Training Data

We have seen above that adding a (relatively) small amount of domain appropriate material to the training set for the c-structure parser has resulted in quite significant gains for both c-structure and f-structure analysis of ATIS sentences. Previous work by Gildea (2001) has shown that a large amount of additional data makes little impact if it is not matched to the test material. With this in mind one can wonder if, due to its relative size, the Penn-II Treebank WSJ material in the training set for the parser might constitute such a large amount of redundant additional data.

In order to test, this we conducted a number of ablation experiments using the automatically f-structure annotated 578 ATIS trees as gold standard in a CCG-style experiment, where we evaluate c-structures and f-structure parser output algorithm, while reducing the amount of Penn-II Treebank material in the parser's training set. The graphs in Figures 6 and 7 show the effect for evaluations against the entire ATIS corpus in a series of 10-fold cross validation experiments, in which the training set for the parser consists of 90% of the ATIS corpus and a varying (randomly selected) percentage of the Penn-II Treebank.

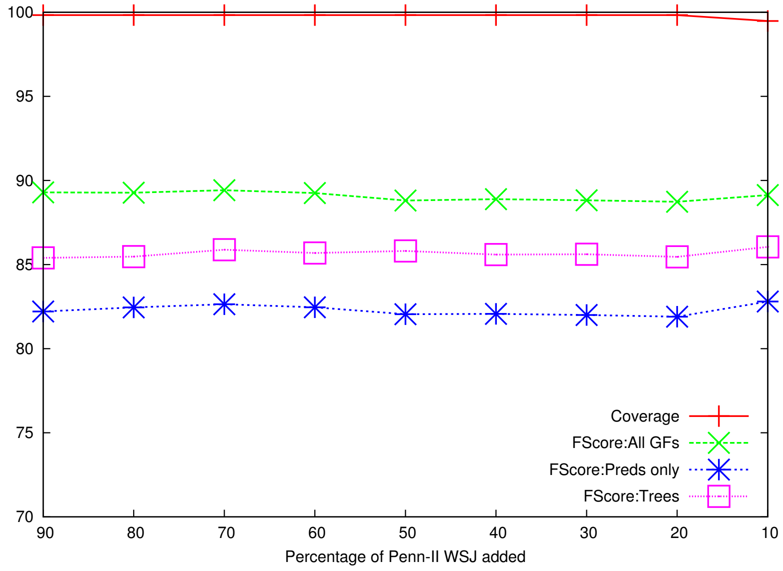

Figure 6: Reducing Penn-II Treebank content (90%-10% of sections 02-21 WSJ, CCG-style experiment)

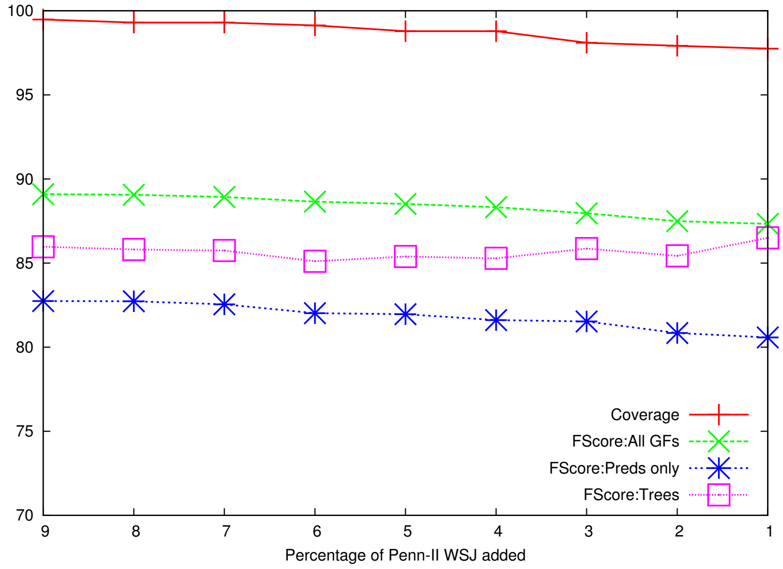

Figure 7: Reducing Penn-II Treebank content (9%-1% of sections 02-21 WSJ, CCG-style experiment)

The graphs show that reducing the amount of Penn-II Treebank WSJ material in the training set adversely affects the overall performance. Grammar coverage, c-structure parsing and f-structure annotation all suffer to varying degrees. Both c-structure and f-structure evaluations start to decline when less than 70% of the treebank is included in the training set. Grammar coverage proves to be less affected in this case: it does not decline significantly until the amount of treebank WSJ training material falls below 20%. Nevertheless, the system is capable of achieving coverage in the region of 99%, a c-structure f-score of over 85%, and f-structure f-scores of over 88% (all grammatical functions) and over 82% (preds-only), when the c-structure parser is trained on 90% of the ATIS corpus and only 10% of the Penn-II Treebank.

6.3 Punctuation

The Penn-II Treebank Wall Street Journal sections used for training the c-structure parser contains properly punctuated text. On the other hand, the ATIS strings are unpunctuated. This is another factor that could possibly explain the underperformance of the c-structure parser and (consequently) annotation algorithm in our earlier experiments, as we would expect grammars trained on Penn-II Treebank sections to perform better on punctuated text.[6]

To test this with the ATIS corpus, we added basic punctuation to each of the ATIS sentences. Each of the 213 questions had a question mark added, the remaining sentences had a fullstop added, and the sub-sentential fragments were left unpunctuated. We then reran the parsing experiments with both the baseline WSJ-only trained grammar, and also the improved WSJ and 90% ATIS trained grammar in a 10-fold cross validation experiment.

	WSJ			WSJ + ATIS 90%		
	Unpunctuated	Punctuated	Diff	Unpunctuated	Punctuated	Diff
Coverage	100	99.83	-0.17	100	99.83	-0.17
F-Score(Trees)	71.42	71.31	-0.11	86.07	85.36	-0.71

Table 7: Parsing results for punctuated ATIS sentences

Table 7 shows the evaluation results for c-structure analysis of the 578 ATIS sentences with basic punctuation added. The table shows the coverage and f-scores for both the baseline grammar, trained on sections 02-21 of the Penn-II Treebank WSJ, and the grammar retrained with added ATIS sentences, and the difference between these scores and those for parsing the ATIS sentences without punctuation. It is interesting to note that all of the scores have decreased slightly as a result of adding punctuation, when the naive assumption, stated above, would be that the parser should perform better given that its training data is punctuated. This emphasises the effect of the domain difference between the ATIS corpus and the Penn-II Treebank.

[6] This was pointed out to us by Tracy King (p.c.).

6.4 Question vs Non-Question

The ATIS corpus contains both question and non-question data. Our 100-sentence gold standard is taken from the ATIS corpus and so comprises both question and non-question sentences. Table 8 shows the breakdown of the upper bound (established following the procedure detailed in Section 6.1) for both question and non-question sentences in the gold standard.

	Non-question	Question
All GFs	94.82	90.77
Preds-only	92.94	86.81

Table 8: Question and non-question f-score upper bounds

The upper bound breakdown shows a slight leaning towards a higher upper bound for non-question sentences, but the upper bound for questions is still quite high.

Table 9 gives the breakdown of the scores for question and non-question sentences in the 100 sentence gold standard parsing evaluations.

	WSJ Trained		WSJ + ATIS Trained			
	Non-Question	Question	Non-Question		Question	
	F-Score	F-Score	F-Score	Diff	F-score	Diff
Trees	74.75	61.92	80.55	+5.8	88.35	+26.43
All GFs	77.40	70.52	82.62	+5.22	84.38	+13.86
Preds-only	68.96	54.12	76.28	+7.32	77.56	+23.44

Table 9: Question and non-question scores for the annotation algorithm

The breakdown in Table 9 clearly shows the effect of both the domain variance and the retraining in the earlier experiments. The left of the table shows the breakdown for the baseline experiments before the parser was retrained. In this experiment it is clear that both the c-structure parser and the f-structure annotation algorithm are underperforming on questions as opposed to non-question sentences. The right of the table shows the same breakdown, but for the experiments with the parser retrained on both Penn-II Treebank WSJ and ATIS sentences. It is clear that this retraining has benefited both the c-structure and f-structure evaluations for the questions in particular. The c-structure tree evaluation has improved over 26% with an f-score of 88.35, likewise the f-structure evaluations have improved for evaluations of all grammatical functions and preds-only, improving by 13.86% and 23.44% respectively. It is also interesting to note that none of the scores have decreased as a result of this retraining, the results for the non-question sentences have also improved (albeit to a lesser extent).

6.5 Back-Testing the Retrained Grammar

The experiments above show that retraining the c-structure parser for the new domain has allowed us to adapt the treebank-based LFG resources to a new domain and achieve similar f-scores in c- and f-structure evaluations on data from a new domain compared to in-domain results. In order to ensure that this retraining process has not adversely affected the overall system performance, we back-test the retrained parser and annotation algorithm on sentences from the original WSJ domain (the DCU 105 gold standard). We parsed the 105 sentences with each of the 10 retrained grammars from the 10-fold cross validation experiment in Section 6.1, then evaluated both c- and f-structures against the DCU 105 gold standard. The averaged results are shown in Table 10 (a), along with the results for the grammar trained only on sections 02-21 of the Penn-II Treebank in the same evaluation (b).

WSJ 02-21 trained		Precision	Recall	F-Score
Trees		86.56	85.59	86.07
F-Structures	All GFs	83.45	78.95	81.14
	Preds-Only	76.32	72.0	74.10

(a)

WSJ 02-21 + 90% ATIS trained		Precision	Recall	F-Score
Trees		87.05	86.10	86.57
F-Structures	All GFs	83.92	79.34	81.56
	Preds-Only	77.32	72.85	75.02

(b)

Table 10: Results for backtesting retrained grammar and baseline grammar on DCU 105

The results show that the retraining process has resulted in no loss of accuracy at either c- or f-structure level. The scores have in fact improved slightly as a result of the retraining; however the improvements, when tested, were not statistically significant (paired t-test). From this we conclude that there has been no significant negative effect on the LFG parsing resources of Cahill et al. (2004) on WSJ material as a result of retraining the c-structure grammar to adapt the treebank-based LFG resources to a new domain.

7 Conclusions

Our experiments have shown that treebank induced LFG resources underperform when the domain is varied from that of the training material. This holds for both c-structure and f-structure analyses. To adapt the treebank-based LFG resources of Cahill et al. (2004) to a new domain, all that was necessary was to retrain the c-structure parser. The f-structure annotation module is able to handle the domain variance without modification. We have also shown that the f-structure annotation algorithm is general: given high-quality c-structure trees, it can achieve a high upper bound for f-structures in a new domain. More generally, our experiments support the claim that the f-structures generated are a more normalised linguistic representation which are less affected by domain variance than the level of c-structure representation.

In our experiments we have adapted our LFG parsing resources to a new domain with a c-structure labelled f-score of 86.07 and an f-structure all grammatical functions f-score of 88.11 in a CCG-style experiment. This constitutes an improvement of over 14% on c-structure parsing, and over 7% on f-structure annotation compared to unadapted parsing and annotation with the same system.

We plan to extend our work by developing a larger question corpus. With such a resource we will be able to parameterise the amount of question data needed in retraining the c-structure parser to reach an optimal result.

References

Bies, A., Ferguson, M., Katz, K., and MacIntyre, R. (1995). Bracketing guidelines for Treebank II style Penn Treebank Project. Technical report, University of Pennsylvania, Philadelphia, PA.

Bikel, D. M. (2002). Design of a multi-lingual, parallel-processing statistical parsing engine. In *Proceedings of Human Language Technology (HLT) 2002*, pages 24–27, San Diego, CA.

Black, E., Abney, S., Flickenger, D., Gdaniec, C., Grishman, R., Harrison, P., Hindle, D., Ingira, R., Jelinek, F., Klavans, J., Liberman, M., Marcus, M., Roukos, S., Santorinim, B., and Strzalkowski, T. (1991). A Procedure for Quantatively Comparing the Coverage of English Grammars. In *Proceedings of the February 1991 DARPA Speech and Natural Language Workshop*, pages 306–311, Pacific Grove, CA.

Burke, M., Cahill, A., O'Donovan, R., van Genabith, J., and Way, A. (2004). The Evaluation of an Automatic Annotation Algorithm against the PARC 700 Dependency Bank . In *Proceedings of the Ninth International Conference on LFG*, pages 101–121, Christchurch, New Zealand.

Cahill, A., Burke, M., O'Donovan, R., van Genabith, J., and Way, A. (2004). Long-Distance Dependency Resolution in Automatically Acquired Wide-Coverage PCFG-Based LFG Approximations. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 320–327, Barcelona, Spain.

Charniak, E. (2000). A maximum entropy inspired parser. In *Proceedings of the First Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL 2000)*, pages 132–139, Seattle, WA.

Clark, S., Steedman, M., and Curran, J. R. (2004). Object-extraction and question-parsing using ccg. In Lin, D. and Wu, D., editors, *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 111–118, Barcelona, Spain.

Collins, M. (1999). *Head-Driven Statistical Models for Natural Language Parsing*. PhD thesis, University of Pennsylvania, Philadelphia, PA.

Crouch, R., Kaplan, R. T., King, T., and Riezler, S. (2002). A comparison of evaluation metrics for a broad coverage parser . In *Beyond PARSEVAL Workshop, Language Resources and Evaluation (LREC)*, pages 67–74, Las Palmas, Canary Islands, Spain.

Gildea, D. (2001). Corpus variation and parser performance. In Lee, L. and Harman, D., editors, *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 167–202, Pittsburgh, PA.

Hemphill, C. T., Godfrey, J. J., and Doddington, G. R. (1990). The ATIS Spoken Language Systems pilot corpus. In *Proceedings of DARPA Speech and Natural Language Workshop*, pages 96–101, Hidden Valley, PA.

Hockenmaier, J. (2003). Parsing with generative models of predicate-argument structure. In *Proceedings of 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 359–366, Sapporo, Japan.

Magerman, D. (1994). *Natural Language Parsing as Statistical Pattern Recognition*. PhD thesis, Department of Computer Science, Stanford University, CA.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1994). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

O'Donovan, R., Burke, M., Cahill, A., van Genabith, J., and Way, A. (2004). Large-Scale Induction and Evaluation of Lexical Resources from the Penn-II Treebank . In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 368–375, Barcelona, Spain.

Pasca, M. and Harabagiu, S. M. (2001). High Performance Question/Answering. In *Research and Development in Information Retrieval*, pages 366–374, New Orleans, LA.