From Deep-Linguistic Dependency Parsing to Biomedical Text Mining

Gerold Schneider,
Fabio Rinaldi, Manfred Klenner, Kaarel Kaljurand

Institute of Computational Linguistics, University of Zurich
gschneid@ifi.uzh.ch

March 9, 2007
Abstract

We present a deep-linguistic, practical system for the discovery of protein interactions in biomedical scientific literature. Our approach is based on probabilistic deep-syntactic parsing for dependency relations and a complex pattern search over the dependency trees to normalize alternations, diathesis, idiomatic expressions, and allowing semantic restrictions.

We show that if reliable term recognition is available, performance on parsing biomedical texts is as accurate as on general text and present a successful Machine-Learning adaptation to the specific genre.

A system for editing and testing KB queries formulated as lexico-syntactic patterns is presented. We conclude with an evaluation of the linguistic performance of the parser, a user-based practical evaluation and its evaluation in recent text mining competition.
Contents

• Motivation

• Probabilistic Dependency Parsing

• Domain Adaptation

• Relation Mining

• Evaluation

• Conclusions
1 Motivation

- Information overload is one of the most widely felt problems in our modern society.

- One domain where such a need is particularly pressing is that of the Biomedical Scientific Literature. The current rapid growth in the amount of published papers makes it increasingly difficult to filter the core information.

Researchers explore new results using PubMed\(^a\). It offers a keyword search over the published articles, and/or interfaces to manually compiled ontologies, such as the Gene Ontology\(^b\), The creation of such resources is a very labour intensive process.

→ A (partial) automation of this activity is therefore highly desirable.

Understanding the relationships between biological entities is central to biology research and drug design. → OntoGene\(^c\)

\(^a\)http://www.ncbi.nlm.nih.gov/entrez
\(^b\)http://www.geneontology.org/
\(^c\)http://www.ontogene.org/
2 Probabilistic Dependency Parsing

2.1 The Need for Parsing

- Parsing possibilities mutually constrain each other: The parsing context weeds out *most* locally possible analyses e.g. *Experts fear the virus will spread*.

- Remote dependencies stretch further than finite-state observation windows

- Non-parsing approaches allow pathological dependencies (verb with several subjects, subject with several verbs)

- Long-distance dependencies can more easily be recognised in a parsing approach
2.2 Fast Robust Dependency Parsing

Advanced parsing techniques combining statistics and human knowledge of linguistics have matured enough to be successfully applied in real settings.

We have parsed the 100 mio. words BNC corpus and over 50 mio. of biomedical texts with our parser Pro3Gres.
3 Pipeline and Statistical Disambiguation

3.1 Pipeline

Figure 1: Pro3Gres flowchart
1. Sentence splitting (using MXTERMINATOR)
2. Tokenization (using the Penn Treebank tokenizer)
3. Part-of-speech tagging (using MXPOST)
4. Lemmatization (using morpha)
5. Terminology detection and detection of term heads (using own tools)
6. Noun and verb group chunking (using LTCHUNK) and detection of chunk heads
7. Dependency parsing (Pro3Gres)
A large subset of syntactic relations, the ones which are considered to be most relevant for argument structure, are modeled.

<table>
<thead>
<tr>
<th>Relation</th>
<th>Label</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>verb–subject</td>
<td>subj</td>
<td>he sleeps</td>
</tr>
<tr>
<td>verb–direct object</td>
<td>obj</td>
<td>sees it</td>
</tr>
<tr>
<td>verb–second object</td>
<td>obj2</td>
<td>gave (her) kisses</td>
</tr>
<tr>
<td>verb–adjunct</td>
<td>adj</td>
<td>ate yesterday</td>
</tr>
<tr>
<td>verb–subord. clause</td>
<td>sentobj</td>
<td>saw (they) came</td>
</tr>
<tr>
<td>verb–pred. adjective</td>
<td>predadj</td>
<td>is ready</td>
</tr>
<tr>
<td>verb–prep. phrase</td>
<td>pobj</td>
<td>slept in bed</td>
</tr>
<tr>
<td>noun–prep. phrase</td>
<td>modpp</td>
<td>draft of paper</td>
</tr>
<tr>
<td>noun–participle</td>
<td>modpart</td>
<td>report written</td>
</tr>
<tr>
<td>verb–complementizer</td>
<td>compl</td>
<td>to eat apples</td>
</tr>
<tr>
<td>noun–preposition</td>
<td>prep</td>
<td>to the house</td>
</tr>
</tbody>
</table>
### 3.2 Lexicalized Label Probability Model

We use a hand-written competence grammar, combined with performance-driven disambiguation obtained from the Penn Treebank (Marcus, Santorini, and Marcinkiewicz, 1993). The MLE probability of generating a dependency relation $R$ given lexical heads ($a$ and $b$) is calculated.

\[
p(R|a, b) \approx \frac{\#(R, a, b)}{\sum_{i=1}^{n} \#(R_i, a, b)}
\]  

(1)

E.g. for the Verb-PP attachment relation $pobj$ (following Collins including the desc. noun = noun inside PP)

\[
p(pobj|\text{verb}, \text{prep}, \text{desc.noun}) \approx \frac{\#(pobj, \text{verb}, \text{prep}, \text{desc.noun})}{\sum_{i=1}^{n} \#(R_i, \text{verb}, \text{prep}, \text{desc.noun})}
\]

(2)

Distance (in chunks) and direction is also included. The counts are backed off (Collins, 1996; Merlo and Esteve Ferrer, 2006). The backoff levels include semantic classes from WordNet (Fellbaum, 1998).
4 Domain Adaptation

Texts from the Biomedical domain are very different from the training domain (newspapers)

- Parsing different domains affects performance (Gildea, 2001)

- Lexicalization is only partly domain-specific: verbs and prepositions vary far less than nouns

- Lexicalization data is sparse anyway, back-offs are used. Heads (e.g. verbs) are more decisive

- Difficult domain nouns (e.g. gene names) also affect tagger and chunker → named entity recognition is key

- Relational nouns (e.g. transcription, down-regulation) are learnt unsupervisedly from domain corpora
4.1 The Importance of Named Entity Recognition

Comparing general chunker to near-perfect terminology annotation on 100 manually annotated sentences from the GENIA corpus, which is manually annotated for term tags and multi-word term boundaries.

<table>
<thead>
<tr>
<th>GENIA “dirty”</th>
<th>Subject</th>
<th>Object</th>
<th>noun-PP</th>
<th>verb-PP</th>
<th>subord. clause</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>83</td>
<td>70</td>
<td>68</td>
<td>67</td>
<td>63</td>
</tr>
<tr>
<td>Recall</td>
<td>74</td>
<td>77</td>
<td>64</td>
<td>68</td>
<td>60</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GENIA “clean”</th>
<th>Subject</th>
<th>Object</th>
<th>noun-PP</th>
<th>verb-PP</th>
<th>subord. clause</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>90</td>
<td>94</td>
<td>83</td>
<td>82</td>
<td>71</td>
</tr>
<tr>
<td>Recall</td>
<td>86</td>
<td>95</td>
<td>82</td>
<td>84</td>
<td>75</td>
</tr>
</tbody>
</table>

Table 1: Evaluation comparing LTChunk chunking ("dirty") and near-perfect term knowledge ("clean") on GENIA corpus

Performance with near-perfect term knowledge ("clean") is similar to performance on training domain texts.
4.2 Unsupervised Machine Learning Domain Adaptations

- Learning relational nouns unsupervisedly from domain corpora: \(NP\ PP\ PP\ PP\)
  based on (Hindle and Rooth, 1993). Sentence-initial NPs are not ambiguous with verb-attachment. They can be cascaded, however. Excluded \(of\ ...\ of\) sequences. This approximation has been shown to improve parsing.

- Learning multi-word terms as collocations:
  using e.g. (Smadja, 2003), multi-word terms are discovered from large domain corpora. Statistical measures such as likelihood ratio can be used to detect collocations (Wilks, 1994)
5 Relation Mining

Our approach to relation mining aims at flexibility and portability, we aim at providing an environment where simple rules can lead to very powerful retrieval.

Our approach is based on 3 levels of rules. On the first level, we exploit syntactic patterns detected in the data, (“macros”).

On the second level we combine various macros into a single semantic rule, which normalises syntactic alternations (e.g. active, passive, nominalizations).

On the third level we combine semantic rules with lexical and ontological constraints to obtain very specialized patterns that detect a given domain-specific relation.
First level: syntactic patterns. For example the macro for the simple passive:

```
macro(simple_passive,"", [X2,X1,X3],
    [  dep(subj,X2,X1),dep(pobj,X2,X3),dep(prep,X3,By),
        pos(X2,'VBN'),lemma(By,['by','through','via'])
    ]).
```
Second level: combining patterns: tree-rewriting procedure

semRel(xrel([H,A,B]), direct_transitive([H,A,B])).
semRel(xrel([H,A,B]), simple_passive([H,B,A])).
semRel(xrel([H,A,B]), nominalization([H,B,A])).

A regulates B
B is regulated by A
the regulation of B by A
Third level: combinations, with lexical constraints (here with keyword \textit{trigger})

\[
domRel(trigger3([H,A,B]),
    [xrel([\text{`trigger'},A,H]),
    \text{nominalisation}(H,Prep),
    \text{deprel}([H,B,Prep])]).
\]

\textit{A triggers the H of B}

\textit{the H of B is triggered by A}

similar third-level patterns for:

\textit{A is involved in the H of B}

\textit{A is under the control of B}
Third level, semantic constraints. For example, the following is a query which restricts the type of the agent to be “protein_molecule”:

\[
xrel(['control',type:'G#protein_molecule',_])
\]

While the following allows it to be “amino_acid” (a more generic term. according to the Genia Ontology):

\[
xrel(['control',type:'G#amino_acid',_])
\]

Because “amino_acid” is a supertype of “protein_molecule”, the results of the latter include (and expand) the results of the former.
<table>
<thead>
<tr>
<th>sid</th>
<th>Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>m92013023-s1</td>
<td>Anti-CD2 receptor antibodies activate the HIV long terminal repeat in T lymphocytes.</td>
</tr>
<tr>
<td>SVG</td>
<td></td>
</tr>
<tr>
<td>m91355651-s5</td>
<td>We found that in both cell lines, both phorbol ester and TNF alpha were able to activate NF-kappa B.</td>
</tr>
<tr>
<td>SVG</td>
<td></td>
</tr>
<tr>
<td>m91355651-s5</td>
<td>We found that in both cell lines, both phorbol ester and TNF alpha were able to activate NF-kappa B.</td>
</tr>
<tr>
<td>SVG</td>
<td></td>
</tr>
<tr>
<td>m94148994-s9</td>
<td>These data suggest that interferon regulatory factor 1 not only triggers the activation of the interferon signal transduction pathway, but also may play a role in limiting the duration of this response by activating the transcription of IRF-2.</td>
</tr>
<tr>
<td>SVG</td>
<td></td>
</tr>
<tr>
<td>m92107162-s5</td>
<td>The simian virus 40 early promoter is also synergistically activated by the Z/c-myb combination.</td>
</tr>
<tr>
<td>SVG</td>
<td></td>
</tr>
<tr>
<td>m91237803-s2</td>
<td>Human herpesvirus 6 (HHV-6) can activate the human immunodeficiency virus (HIV) promoter and accelerate cytopathic effects in HIV-infected human T cells.</td>
</tr>
</tbody>
</table>
6 Evaluation

6.1 Application-Oriented Expert Evaluation

We asked domain experts to evaluate each relation. Example:

<table>
<thead>
<tr>
<th>relation</th>
<th>agent</th>
<th>agent type</th>
<th>agent eval</th>
<th>target</th>
<th>target type</th>
<th>target eval</th>
</tr>
</thead>
<tbody>
<tr>
<td>activate</td>
<td>Interleukin-2 (IL-2)</td>
<td>protein</td>
<td>Y</td>
<td>Stat5 in fresh PBL, and Stat3 and Stat5 in preactivated PBL</td>
<td>protein</td>
<td>A+</td>
</tr>
<tr>
<td>activate</td>
<td>IL-5</td>
<td>protein</td>
<td>Y</td>
<td>the Jak 2-STAT 1 signaling pathway</td>
<td>other name</td>
<td>Y</td>
</tr>
<tr>
<td>bind</td>
<td>Spi-B</td>
<td>protein</td>
<td>Y</td>
<td>DNA sequences</td>
<td>nucleic acid</td>
<td>A-</td>
</tr>
</tbody>
</table>

Y if the relation is correct and biologically significant mark as correct
A if the relation is correct and biologically significant, but includes too much or too little information, mark as correct
P if the relation appears correct, but an anaphora needs to be resolved, mark as incorrect
N if the relation is completely incorrect mark as incorrect
Table 2: Analysis of user evaluation precision on GENIA

In the absence of a gold standard, only approximative recall values of about 60 % can be reported.
6.2 Participation in BioCreAtIvE

BioCreAtIvE is an international competition for text mining systems over biomedical literature. There are a number of different subtasks. We participated in two subtasks in 2006.

- IPS: identification of protein-protein interactions from Medline papers
- IMS: identification of the method used by the authors to experimentally verify an interaction

According to preliminary information by the organisers, our system is the best performing in IMS and probably among the best performing in IPS.
Procedure for IPS task:

1. Detection and disambiguation of protein names

2. High recall step: generate candidate interactions by combining all proteins belonging to the same sentence

3. Disambiguation step 1:
   Detection of ‘curatable’ sentences. Based on keywords and on salience

   Here we report the isolation of a new DLG-interacting protein, GUK-holder, that interacts with the GUK domain of DLG and which is dynamically expressed during synaptic bouton budding.

   Previous studies have revealed a genetic interaction between DLG and another PDZ scaffolding protein, SCRIBBLE (SCRIB), during the establishment of cell polarity in developing epithelia.

   We have used an SVM classifier on a bag-of-lemmas representation for each sentence containing proteins mentioned in the abstract.

Only proteins related by a licensing chain of syntactic relations are considered.

*Daxx binds to Mdm2 and the deubiquitinase Hausp*

Three possible interactions can be considered:

1. **Daxx – Mdm2**  
2. **Daxx – Hausp**  
3. **Mdm2 – Hausp**

On syntactic grounds, only the first two interactions are licensed.

The syntactic relation chain uses the three-level patterns presented previously.
7 Conclusions

Large-scale full parsing of Biomedical texts is feasible.

We have shown a language that allows users to write complex patterns collecting various alternations.

We have conducted a user-based evaluation and participated in a biomedical competition. Recall and especially precision results are very encouraging.
References


IPS results

<table>
<thead>
<tr>
<th></th>
<th>Average per article</th>
<th>Average all interactions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Precision</td>
<td>Recall</td>
</tr>
<tr>
<td>Ontogene</td>
<td>0.263</td>
<td>0.248</td>
</tr>
<tr>
<td>average</td>
<td>0.106</td>
<td>0.186</td>
</tr>
<tr>
<td>Ontogene</td>
<td>0.275</td>
<td>0.274</td>
</tr>
<tr>
<td>average</td>
<td>0.116</td>
<td>0.200</td>
</tr>
</tbody>
</table>

Table 3: BiocreAtIvE results for the protein-protein interaction task. Each table compares Ontogene results with average results of all participants. The first row (UniProt) shows results computed taking into account all interactions, the second row considers only interactions involving Swiss-Prot proteins (Swiss-Prot is a subset of UniProt). The results on the left are average of results on each article, while results on the right are results obtained considering all interactions in all articles as a single pool.

Results might appear low in absolute terms: the setup of the task was extremely complex: the participating systems had to identify proteins, normalize them to UniProt identifiers, find interactions, and distinguish novel from known interactions.
Dependency Grammar is a constituent grammar in which words are the only permissible constituents.

Constituency and Dependency are largely equivalent.
8 The Need for Parsing

- Parsing possibilities mutually constrain each other: The parsing context weeds out *most* locally possible analyses → far fewer global analyses
- Remote dependencies stretch further than finite-state observation windows
- Non-parsing approaches allow pathological dependencies (verb with several subjects, subject with several verbs)
- Long-distance dependencies can more easily be recognised in a parsing approach
Example of a parse containing a long-distance dependency
Research on PCFG and PP-attachment has shown the importance of probabilizing on lexical heads (a and b).

\[
p(R|A \rightarrow AB, a, b) \approx \frac{\#(R, A \rightarrow AB, a, b)}{\sum_{i=1}^{n} \#(R_i, A \rightarrow AB, a, b)}
\]  
(3)

All that \(A \rightarrow AB\) expresses is that in the dependency relation the dependency is towards the right.

\[
p(R|\text{right}, a, b) \approx \frac{\#(R, \text{right}, a, b)}{\sum_{i=1}^{n} \#(R_i, \text{right}, a, b)}
\]  
(4)

e.g. for the Verb-PP attachment relation \(pobj\) (following (Collins and Brooks, 1995) including the desc. noun = noun inside PP)

\[
p(pobj|\text{right}, \text{verb}, \text{prep}, \text{desc.noun}) \approx \frac{\#(pobj, \text{right}, \text{verb}, \text{prep}, \text{desc.noun})}{\#([pobj, \text{modpp}], \text{right}, \text{verb}, \text{prep}, \text{desc.noun})}
\]  
(5)
(Collins, 1996) MLE estimation: \( P(R|\langle a, \text{atag}\rangle, \langle b, \text{btag}\rangle, \text{dist}) \approx \)
\[
\frac{\#(R, \langle a, \text{atag}\rangle, \langle b, \text{btag}\rangle, \text{dist})}{\#(\langle a, \text{atag}\rangle, \langle b, \text{btag}\rangle, \text{dist})}
\]
\( (6) \)

(Schneider, 2003b) MLE estimation: \( P(R, \text{dist}|a, b) \approx p(R|a, b) \cdot p(\text{dist}|R) \approx \)
\[
\frac{\#(R, a, b)}{\sum_{i=1}^{n} \#(R_i, a, b)} \cdot \frac{\#(R, \text{dist})}{\#R}
\]
\( (7) \)

- licencing, rule-based hand-written grammar over Penn tags
- back-off to semantic classes (WordNet)
- real distance, measured in chunks
- co-occurrence in denominator is not sentence-context, but \( \sum \) of competing relations (e.g. object/adjunct or subject/modpart) \( \rightarrow \) decision probabilities
- Relations (R) have a Functional Dependency Grammar definition (overleaf)
Reduced, chunked Tree representation for the sentence *This man eats bananas with a fork* leads to the following Dependency Relations:

(Collins, 1996)

(Schneider, 2003b)
8.1 Backoffs

In addition to backing off to partly lexicalized counts semantic classes,

- for nouns \(a\): top Wordnet class of the most frequent sense \(\hat{a}\)
- for verbs \(b\): the Wordnet/Levin classes \(\hat{b}\)

**Example modpart**: always to right; added passive counts to fight sparse data

data

\[
\begin{align*}
\text{the report announced the deal (subj) } &\leftrightarrow \text{ the report announced yesterday (modpart)} \\
\text{the report announced yesterday (modpart) } &\approx \text{ the report has been announced (psubj)}
\end{align*}
\]

\[
p(\text{modpart}|a, b) &\approx\]

\[
\frac{\#(\text{modpart}, \text{right}, a, b) + \#(\text{psubj}, \text{left}, a, b)}{\#(\text{modpart}, \text{right}, a, b) + \#(\text{psubj}, \text{left}, a, b) + \#(\text{asubj}, \text{left}, a, b)} \quad \text{if} \gt 0, \text{else}
\]

\[
\frac{\#(\text{modpart}, \text{right}, \hat{a}, \hat{b}) + \#(\text{psubj}, \text{left}, \hat{a}, \hat{b})}{\#(\text{modpart}, \text{right}, \hat{a}, \hat{b}) + \#(\text{psubj}, \text{left}, \hat{a}, \hat{b}) + \#(\text{asubj}, \text{left}, \hat{a}, \hat{b})} \quad \text{if} \gt 0, \text{else}
\]

\[
\frac{\#(\text{modpart}, \text{right}, b) + \#(\text{psubj}, \text{left}, b)}{\#(\text{modpart}, \text{right}, b) + \#(\text{psubj}, \text{left}, b) + \#(\text{asubj}, \text{left}, b)} \quad \text{if} \gt 0, \text{else}
\]

\[
\frac{\#(\text{modpart}, \text{right}, a) + \#(\text{psubj}, \text{left}, a) + \#(\text{asubj}, \text{left}, a)}{\#(\text{modpart}, \text{right}, a) + \#(\text{psubj}, \text{left}, a) + \#(\text{asubj}, \text{left}, a)}
\]

9 Extraction of Lexicalized Dependencies

Active subject relation has the head of an arbitrarily nested ("@") NP as dependent, and the head of an arbitrarily nested VP as head.

Passive subject and control subject:

```
NP-SBJ-X@               NP-SBJ-X@  
  noun                  noun
  V                     V
  passive verb          control-verb
  -NONE-                -NONE-
  *-X                   *-X

```

"local", fixed dependencies across several subtrees (extended domain of locality) → can be reduced to really local dependency with a dedicated label. This applies to most LDDs except WH (Carroll et al., 1999; Schneider, 2003a)
A large subset of syntactic relations, the ones which are considered most relevant for argument structure and which are most ambiguous, are modeled. Some use functional labels, several levels of subtrees and empty nodes as integral parts.

<table>
<thead>
<tr>
<th>Relation</th>
<th>Label</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>verb–subject</td>
<td>subj</td>
<td>he sleeps</td>
</tr>
<tr>
<td>verb–direct object</td>
<td>obj</td>
<td>sees it</td>
</tr>
<tr>
<td>verb–second object</td>
<td>obj2</td>
<td>gave (her) kisses</td>
</tr>
<tr>
<td>verb–adjunct</td>
<td>adj</td>
<td>ate yesterday</td>
</tr>
<tr>
<td>verb–subord. clause</td>
<td>sentobj</td>
<td>saw (they) came</td>
</tr>
<tr>
<td>verb–pred. adjective</td>
<td>predadj</td>
<td>is ready</td>
</tr>
<tr>
<td>verb–prep. phrase</td>
<td>obj</td>
<td>slept in bed</td>
</tr>
<tr>
<td>noun–prep. phrase</td>
<td>modpp</td>
<td>draft of paper</td>
</tr>
<tr>
<td>noun–participle</td>
<td>modpart</td>
<td>report written</td>
</tr>
<tr>
<td>verb–complementizer</td>
<td>compl</td>
<td>to eat apples</td>
</tr>
<tr>
<td>noun–preposition</td>
<td>prep</td>
<td>to the house</td>
</tr>
</tbody>
</table>

Verb–subject has a different probability model for active and passive
<table>
<thead>
<tr>
<th>#</th>
<th>Antecedent</th>
<th>POS</th>
<th>Label</th>
<th>Count</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NP</td>
<td>NP</td>
<td>*</td>
<td>22,734</td>
<td>NP trace</td>
<td><em>Sam was seen</em></td>
</tr>
<tr>
<td>2</td>
<td>NP</td>
<td>*</td>
<td>PRO</td>
<td>12,172</td>
<td>to sleep is nice</td>
<td>* to sleep is nice</td>
</tr>
<tr>
<td>3</td>
<td>WHNP</td>
<td>NP</td>
<td><em>T</em></td>
<td>10,659</td>
<td>WH trace</td>
<td>the woman who you saw <em>T</em></td>
</tr>
<tr>
<td>4</td>
<td>WHNP</td>
<td><em>U</em></td>
<td></td>
<td>9,202</td>
<td>Empty units</td>
<td>$25 <em>U</em></td>
</tr>
<tr>
<td>5</td>
<td>S</td>
<td>S</td>
<td>0</td>
<td>7,057</td>
<td>Empty complementizers</td>
<td>Sam said 0 Sasha snores</td>
</tr>
<tr>
<td>6</td>
<td>WHADVP</td>
<td>ADVP</td>
<td><em>T</em></td>
<td>3,181</td>
<td>WH-trace</td>
<td>Sam explained how to leave <em>T</em></td>
</tr>
<tr>
<td>7</td>
<td>S</td>
<td>S</td>
<td><em>U</em></td>
<td>2,513</td>
<td>Empty clauses</td>
<td>*Sam had to go, Sasha said <em>T</em></td>
</tr>
<tr>
<td>8</td>
<td>WHNP</td>
<td>0</td>
<td></td>
<td>2,139</td>
<td>Empty relative pronouns</td>
<td>the woman who we saw</td>
</tr>
<tr>
<td>9</td>
<td>WHADVP</td>
<td>0</td>
<td></td>
<td>726</td>
<td>Empty relative pronouns</td>
<td>the reason who to leave</td>
</tr>
</tbody>
</table>

Table 4: The distribution of the 10 most frequent types of empty node and their antecedents in the Penn Treebank (adapted from Johnson2002). Bracketted lines designate long-distance dependencies that are local in DG
10 DG, TAG and LFG

- According to TAG, only mild context-sensitivity is needed for NL expressiveness.
- TAG Adjoining is the only operation needed for mild context-sensitivity.
- What would Adjoining correspond to in other formal grammars?
- LFG functional uncertainty has been shown to follow as a corollary from TAG Adjoining (Joshi and Vijay-Shanker, 1989).
10.1 DG Preliminaries

- Nuclei and elementary trees are similar: we “assume that elementary trees are built around a single lexical element, that is, a semantically contentful word like a noun, verb or adjective” (Frank, 2004, p. 11).

- (Grimshaw, 1991), (Carroll et al., 1999), (Frank, 2002) discuss that in head-movement the base position and the ultimate landing site lie within a single extended projection. This entails that head-movement generally is not unbounded. (Schneider, 2003a) shows that finite-state patterns (instead of costly movement as in TAG) can be used.

- Functional words are attached as dependents in DG, thus DG equivalents of functional projections are all projections of the head word.
10.2 Adjoining in DG

The only possible foot node $N$ in DG is a projection $W$ of the head word $W$ (endocentricity). Adjoining inserts a recursive structure at some projection $N$ (TAG foot node). The head of the inserted structure is $N$, and the part of the elementary tree that appeared below $N$ occurs below the recursive $N$. Since the foot node $N$ of the inserted auxiliary tree appears above the $N$ of the original elementary tree, Adjunction inserts new governors into an existing structure and thus breaks the context-freeness. The DG difference between Substitution and Adjoining is:
Substitution inserts dependents, Adjoining inserts governors.

If every projective DG relation corresponds to a TAG elementary tree and every trigger for a non-projective DG relation corresponds to a TAG auxiliary tree then they are equivalent.
11 Evaluation

Subject

| Precision: | subj | \(\rightarrow\) | ncssubj\(_C\) OR cmod\(_C\)(with rel.pro) |
| Recall: | ncssubj\(_C\) | \(\rightarrow\) | subj OR modpart |

\(ncsubj\(_C\)=\) non-clausal subject
\(cmod\(_C\)=\) clausal modification, used for relative clauses (but not all \(cmod\(_C\)\) are relative pronouns)

Object

| Precision: | obj OR obj\(_2\) | \(\rightarrow\) | dobj\(_C\) OR obj\(_2\)\(_C\) |
| Recall: | dobj\(_C\) OR obj\(_2\)\(_C\) | \(\rightarrow\) | obj OR obj\(_2\) |

\(dobj\(_C\)=\)first object
\(obj\(_2\)\(_C\)=\)second object

noun-PP

| Precision: | modpp | \(\rightarrow\) | ncmmod\(_C\)(with prep) OR xmod\(_C\)(with prep) |
| Recall: | ncmmod\(_C\)(with prep) OR xmod\(_C\)(with prep) | \(\rightarrow\) | modpp |

\(ncmmod\(_C\)=\)non-clausal modification
\(xmod\(_C\)=\)clausal modification for verb-to-noun translations
### 11.1 General Evaluation and Comparison

<table>
<thead>
<tr>
<th></th>
<th>Percentage Values for</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Subject</td>
<td>Object</td>
<td>noun-PP</td>
</tr>
<tr>
<td>Precision</td>
<td>91</td>
<td>89</td>
<td>73</td>
</tr>
<tr>
<td>Recall</td>
<td>81</td>
<td>83</td>
<td>67</td>
</tr>
</tbody>
</table>

Comparison to Lin (on the whole Susanne corpus):

<table>
<thead>
<tr>
<th></th>
<th>Subject</th>
<th>Object</th>
<th>PP-attachment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>89</td>
<td>88</td>
<td>78</td>
</tr>
<tr>
<td>Recall</td>
<td>78</td>
<td>72</td>
<td>72</td>
</tr>
</tbody>
</table>

Comparison to Buchholz (Buchholz, 2002); and to Charniak (Charniak, 2000), according to Preiss:

<table>
<thead>
<tr>
<th></th>
<th>ncs\textit{subj}</th>
<th>do\textit{obj}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>86; 82</td>
<td>88; 84</td>
</tr>
<tr>
<td>Recall</td>
<td>73; 70</td>
<td>77; 76</td>
</tr>
</tbody>
</table>
### 11.2 Selective LDD evaluation (as far as the annotations permit)

<table>
<thead>
<tr>
<th></th>
<th>LDD relations results for</th>
</tr>
</thead>
<tbody>
<tr>
<td>WH-Subject Precision</td>
<td>57/62 92%</td>
</tr>
<tr>
<td>WH-Subject Recall</td>
<td>45/50 90%</td>
</tr>
<tr>
<td>WH-Object Precision</td>
<td>6/10 60%</td>
</tr>
<tr>
<td>WH-Object Recall</td>
<td>6/7 86%</td>
</tr>
<tr>
<td>Anaphora of the rel. clause subject Precision</td>
<td>41/46 89%</td>
</tr>
<tr>
<td>Anaphora of the rel. clause subject Recall</td>
<td>40/63 63%</td>
</tr>
<tr>
<td>Passive subject Recall</td>
<td>132/160 83%</td>
</tr>
<tr>
<td>Precision for subject-control subjects</td>
<td>40/50 80%</td>
</tr>
<tr>
<td>Precision for object-control subjects</td>
<td>5/5 100%</td>
</tr>
<tr>
<td>Precision of modpart relation</td>
<td>34/46 74%</td>
</tr>
<tr>
<td>Precision for topicalized verb-attached PPs</td>
<td>25/35 71%</td>
</tr>
</tbody>
</table>
11.3 Comparison to Collins / Distances and Lexicalisation

<table>
<thead>
<tr>
<th>Relation</th>
<th>Collins Model 1</th>
<th>Pro3Gres</th>
</tr>
</thead>
<tbody>
<tr>
<td>subj_prec</td>
<td>657 of 728</td>
<td>90.25%</td>
</tr>
<tr>
<td>subj_recall</td>
<td>627 of 956</td>
<td>92.38%</td>
</tr>
<tr>
<td>obj_prec</td>
<td>403 of 452</td>
<td>89.16%</td>
</tr>
<tr>
<td>obj_recall</td>
<td>297 of 391</td>
<td>83.89%</td>
</tr>
<tr>
<td>nounpp_prec</td>
<td>306 of 388</td>
<td>74.39%</td>
</tr>
<tr>
<td>verbpp_prec</td>
<td>293 of 401</td>
<td>72.43%</td>
</tr>
<tr>
<td>ncmod_recall</td>
<td>443 of 801</td>
<td>65.54%</td>
</tr>
<tr>
<td>iobj_recall</td>
<td>112 of 157</td>
<td>85.99%</td>
</tr>
<tr>
<td>argmod_recall</td>
<td>30 of 41</td>
<td>80.49%</td>
</tr>
</tbody>
</table>

Table 5: Comparison of parsing accuracy between Collins Model 1 and Pro3Gres
Figure 2: Comparison between Baseline, Distance Measure Only, Full Model without Distance Measure, and Full Model
## 11.4 Detailed Analysis of PP-Attachment Errors

<table>
<thead>
<tr>
<th>Attachment Error</th>
<th>Head Extraction Error</th>
<th>Chunking or Tagging Error</th>
<th>compl/prep Error</th>
<th>Grammar Mistake or incompl. Parse</th>
<th>Grammar Assumption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noun-PP Attachment Precision ($modpp$)</td>
<td>22</td>
<td>1</td>
<td>8</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Verb-PP Attachment Precision ($pobj$)</td>
<td>12</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Noun-PP Attachment Recall ($ncmod_{c}$)</td>
<td>25</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>Some Verb-PP Attachment Recall ($iobj_{c}$)</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>Partial Total</strong></td>
<td><strong>61</strong></td>
<td><strong>3</strong></td>
<td><strong>28</strong></td>
<td><strong>1</strong></td>
<td><strong>16</strong></td>
</tr>
<tr>
<td><strong>51 %</strong></td>
<td><strong>3 %</strong></td>
<td><strong>24 %</strong></td>
<td><strong>1 %</strong></td>
<td><strong>13 %</strong></td>
<td><strong>12 %</strong></td>
</tr>
</tbody>
</table>
11.5 Comparison to Carroll

11.5.1 Chunk-internal relations

Pre-processing module to recover relations inside chunks

1. Non-head words in a chunk modify the head. 3 exceptions:
   - In verb chunks, every verb modifies the succeeding verb (example: *would have been going*).
   - If noun chunk contains adjective plus > 1 noun: if more often before non-head noun than head noun in BNC \(\rightarrow\) adjective modifies non-head noun (example: *conventional forces strengthening*).
• If noun chunk contains > 2 nouns: if 1st noun more often before the non-head noun than head noun lemma in BNC → 1st noun modifies succeeding non-head noun (example: *Eisenhower administration effort*).

• If noun chunk contains > 2 proper names: every proper name modifies succeeding proper name (example: *Fulton County Grand Jury*).

2. Modification type follows from POS tag of head and modifier. Exception:

• if either determiner or adjective (*several, many, few, one*) *detmod* if 1st word in chunk, *ncmod* otherwise.
11.5.2 Complement/Adjunct distinction

Small correlation between attachment and argumenthood. We use a post-processing approach to PPs as arguments \((iobj_c)\) or as adjuncts \((ncmod_c)\). NB: inter-annotator agreement is low, no reliable training data exists.

Conservative approach: verbal PPs with label \(CLR\) are argument, all others adjunct.
Carroll’s own GREVAL evaluation reports the numbers in 6. Comparison to Pro3Gres performance:

<table>
<thead>
<tr>
<th>Relation</th>
<th>RASP</th>
<th>Pro3Gres</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Precision</td>
<td>Recall</td>
</tr>
<tr>
<td></td>
<td>%</td>
<td>%</td>
</tr>
<tr>
<td>ncmmod</td>
<td>78</td>
<td>73</td>
</tr>
<tr>
<td>xmod</td>
<td>70</td>
<td>52</td>
</tr>
<tr>
<td>cmod</td>
<td>67</td>
<td>48</td>
</tr>
<tr>
<td>arg_mod</td>
<td>84</td>
<td>41</td>
</tr>
<tr>
<td>ncsbj</td>
<td>85</td>
<td>88</td>
</tr>
<tr>
<td>xsbj</td>
<td>100</td>
<td>40</td>
</tr>
<tr>
<td>csbj</td>
<td>14</td>
<td>100</td>
</tr>
<tr>
<td>dobj</td>
<td>86</td>
<td>84</td>
</tr>
<tr>
<td>obj2</td>
<td>39</td>
<td>84</td>
</tr>
<tr>
<td>iobj</td>
<td>42</td>
<td>65</td>
</tr>
<tr>
<td>xcomp</td>
<td>84</td>
<td>79</td>
</tr>
<tr>
<td>ccomp</td>
<td>72</td>
<td>75</td>
</tr>
</tbody>
</table>

Table 6: RASP evaluation results compared to Pro3Gres
11.6 Parsing the terminology-annotated GENIA corpus
12 Conclusions

- fast (~ 300,000 words/h), lexicalized broad-coverage parser with grammatical relation (GR) output

- Parser’s performance is state-of-the-art.

- Most non-local dependencies can be treated as local dependencies by using and modelling dedicated patterns across several levels of constituency subtrees, by lexicalized post-processing rules and because some non-local dependencies are artifacts of the grammatical representation. For the remaining non-local dependencies, mild context-sensitivity is enough.

- A version of Adjoining has been implemented in DG. DG and TAG are probably equivalent, but DG parsing complexity is lower → representationally minimal grammar.
13 Experiments

High Recall on Carroll Corpus

<table>
<thead>
<tr>
<th>Percent</th>
<th>Subject</th>
<th>Object</th>
<th>noun-PP</th>
<th>verb-PP</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td></td>
<td></td>
<td>64.9</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td></td>
<td></td>
<td>70.4</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td></td>
<td></td>
<td>73.9</td>
<td></td>
</tr>
<tr>
<td>75</td>
<td></td>
<td></td>
<td>75.2</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td></td>
<td></td>
<td>81.4</td>
<td></td>
</tr>
<tr>
<td>85</td>
<td></td>
<td></td>
<td>81.6</td>
<td></td>
</tr>
<tr>
<td>90</td>
<td></td>
<td></td>
<td>81.8</td>
<td></td>
</tr>
<tr>
<td>95</td>
<td></td>
<td></td>
<td>81.9</td>
<td></td>
</tr>
</tbody>
</table>
High Recall on GENIA Corpus

<table>
<thead>
<tr>
<th>Percent</th>
<th>Subject</th>
<th>Object</th>
<th>noun-PP</th>
<th>verb-PP</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>65</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>70</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>75</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>80</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>85</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>95</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- 1 analysis
- max. 2 analyses
- max. 4 analyses
- max. 8 analyses
- max. 16 analyses

Graph showing precision and recall for different analysis levels.
Table 7: Percentage results of Experiment 2: keeping only agreeing relations arising from parsing with two taggers, on Carroll’s test corpus on subject, object and PP-attachment relations.

<table>
<thead>
<tr>
<th>Experiment 2</th>
<th>Subject</th>
<th>Object</th>
<th>noun-PP</th>
<th>verb-PP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>94.1</td>
<td>93.0</td>
<td>73.3</td>
<td>75.4</td>
</tr>
<tr>
<td>Recall</td>
<td>76.4</td>
<td>78.8</td>
<td>60.5</td>
<td>80.3</td>
</tr>
</tbody>
</table>
Experiment 4 Precision on Carroll

Distance 1-2 Precision
Distance 1-3 Precision
Distance 1-4 Precision
Distance 1-5 Precision

Percent

Subject
Object
noun-PP
verb-PP
<table>
<thead>
<tr>
<th>Distance 1-2 Precision</th>
<th>Distance 1-3 Precision</th>
<th>Distance 1-4 Precision</th>
<th>Distance 1-5 Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>92.3</td>
<td>89.5</td>
<td>90.2</td>
<td>90.9</td>
</tr>
<tr>
<td>92.9</td>
<td>92.9</td>
<td>92.9</td>
<td>92.9</td>
</tr>
<tr>
<td>88.1</td>
<td>87.2</td>
<td>86.8</td>
<td>85.6</td>
</tr>
<tr>
<td>95.5</td>
<td>87.6</td>
<td>87.5</td>
<td>85.6</td>
</tr>
</tbody>
</table>

Experiment 4 Precision on GENIA

- **Subject**
- **Object**
- **noun-PP**
- **verb-PP**
Experiment 4 Recall on GENIA

<table>
<thead>
<tr>
<th>Distance 1-2 Recall</th>
<th>Distance 1-3 Recall</th>
<th>Distance 1-4 Recall</th>
<th>Distance 1-5 Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>57.1</td>
<td>64.8</td>
<td>69.4</td>
<td>74.5</td>
</tr>
<tr>
<td>91.1</td>
<td>91.1</td>
<td>91.1</td>
<td>91.1</td>
</tr>
<tr>
<td>79</td>
<td>79</td>
<td>79.5</td>
<td>80</td>
</tr>
<tr>
<td>64.7</td>
<td>74.1</td>
<td>77.7</td>
<td>79.1</td>
</tr>
<tr>
<td>50</td>
<td>55</td>
<td>60</td>
<td>65</td>
</tr>
<tr>
<td>70</td>
<td>75</td>
<td>80</td>
<td>85</td>
</tr>
<tr>
<td>80</td>
<td>85</td>
<td>90</td>
<td>95</td>
</tr>
<tr>
<td>90</td>
<td>95</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>

Legend:
- **Subject**
- **Object**
- **noun-PP**
- **verb-PP**
Precision-corrected Experiment 5 on Carroll: Precision Values

Percent

Threshold Threshold Threshold Threshold Threshold Threshold Threshold Threshold Threshold Threshold Threshold Threshold

0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 0.95 0.99

- nounpp_prec

- verbpp_prec

75.29 75.23 75.52 75.70 76.00 77.54 78.80 79.95 80.49 84.19 85.22 85.23

72.77 72.93 72.81 72.85 73.15 73.38 73.99 74.75 74.07 76.76 77.27 82.69 82.69
Precision-corrected Experiment 5 on Carroll: Recall Values

The graph shows recall values for different thresholds. The x-axis represents the percent (50.00 to 100.00) and the y-axis represents the threshold (0.0 to 0.99). The recall values are indicated by symbols: △ for {ncmod_recall}, × for {iobj_recall}, and * for {argmod_recall}. The specific recall values at different thresholds are not detailed in the image.
PP-Attachment Precision Values by Back-off Level

Numbers of [Noun,Verb] occurrences returned by the parser in angular brackets

Percent nounpp verbpp

Levels:
- Level 0 [24,26]
- Level 2 [254,279]
- Level 3 & 3.1 [21,20]
- Level 4 & 4.1 [99,74]
- Levels 5 & 6 [26,18]

Graph showing precision values for different back-off levels.
14 Corpus Linguistics with Large Syntactically Annotated Corpora

14.1 Syntactically annotating the BNC?

In a realistic annotation or filtering scenario using the high recall parsing method with the annotator selecting the best of top 16 analyses, over subject, object and PP-attachment relations, GENIA recall is $\frac{564}{618} = 91.3\%$.

Deducting spurious errors, long-distance dependencies (LDDs, often left underspecified by statistical parsers), bearing in mind that the parser is affected by tagging and chunking mistakes, actual GENIA micro-average high recall parsing performance for the evaluated relations can be confirmed to reach $95\%$.

BNC tagging quality is much higher than automatic, possibly higher than GENIA.
14.2 Using High Precision / High Recall Parsing Results

If for the phenomenon under investigation

1. automatically parsed results generally achieve high precision (and at least moderate recall) or

2. representative, high-precision samples can be identified or

3. researchers are ready to do e.g. high recall filtering (16 best)

then also descriptive corpus linguistics can profit from very large, syntactically automatically annotated corpora.
15 Related Approaches

- (Collins, 1997), (Dubey and Keller, 2003)
- (Jijkoun, 2003) extends Johnson’s patterns, statistical approach
- Dependency-based statistical parsing (Eisner, 1996; Nivre, 2003). No LDDs, sometimes unlabeled dependencies
- Memory-Based Grammatical Relation Finding (Daelemans, Buchholz, and Veenstra, 1999; Buchholz, 2002). Non-parsing approach
- Weighted grammars, Maximum Entropy Modeling: Hand-written, usually Formal Grammars, with added lexicalized probabilities. Recently, for the first time, LFG grammars managed to parse the Treebank (Riezler et al., 2002; Burke et al., 2004).
- “Finite-state LDD approximations”: work in broad-coverage LFG (Burke et al., 2004)
- Pro3Gres is about 10 times faster than the fastest such LFG approach (van Genabith, p.c.)
- Similar work, automatically extracting and weighting grammars from Penn Treebank exists for HPSG (Miyao) and for CCG (Hockenmaier and Steedman, 2002)
## Frequency Analysis of Empty Nodes

Distribution of the 10 most frequent types of empty nodes and their antecedents in the Penn Treebank (adapted from (Johnson, 2002))

<table>
<thead>
<tr>
<th></th>
<th>Antecedent</th>
<th>POS</th>
<th>Label</th>
<th>Count</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NP</td>
<td>NP</td>
<td>*</td>
<td>22,734</td>
<td>NP trace</td>
<td><em>Sam was seen</em></td>
</tr>
<tr>
<td>2</td>
<td>NP</td>
<td>*</td>
<td></td>
<td>12,172</td>
<td>NP PRO</td>
<td>*to sleep is nice</td>
</tr>
<tr>
<td>3</td>
<td>WHNP</td>
<td>NP</td>
<td><em>T</em></td>
<td>10,659</td>
<td>WH trace</td>
<td><em>the woman whom you saw</em></td>
</tr>
<tr>
<td>4</td>
<td><em>U</em></td>
<td></td>
<td></td>
<td>9,202</td>
<td>Empty units</td>
<td><em>$25</em></td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td></td>
<td></td>
<td>7,057</td>
<td>Empty complementizers</td>
<td><em>Sam said 0 Sasha snores</em></td>
</tr>
<tr>
<td>6</td>
<td>S</td>
<td>S</td>
<td><em>T</em></td>
<td>5,035</td>
<td>Moved clauses</td>
<td><em>Sam had to go, Sasha said</em></td>
</tr>
<tr>
<td>7</td>
<td>WHADVP</td>
<td>ADVP</td>
<td><em>T</em></td>
<td>3,181</td>
<td>WH-trace</td>
<td><em>Sam explained</em> how to leave*</td>
</tr>
<tr>
<td>8</td>
<td>SBAR</td>
<td></td>
<td></td>
<td>2,513</td>
<td>Empty clauses</td>
<td><em>Sam had to go, said Sasha</em> (SBAR)</td>
</tr>
<tr>
<td>9</td>
<td>WHNP</td>
<td>0</td>
<td></td>
<td>2,139</td>
<td>Empty relative pronouns</td>
<td><em>the woman 0 we saw</em></td>
</tr>
<tr>
<td>10</td>
<td>WHADVP</td>
<td>0</td>
<td></td>
<td>726</td>
<td>Empty relative pronouns</td>
<td><em>the reason 0 to leave</em></td>
</tr>
</tbody>
</table>

Empty elements [rows 4,5,9,10] → non-nucleus material
Moved clauses[6], subj – utterance-verb inversion[8] → change of canonical direction


## 16.1 NP Traces

Coverage of the patterns for the most frequent NP traces [row 1]

<table>
<thead>
<tr>
<th>Type</th>
<th>Count</th>
<th>prob-modeled</th>
<th>Treatment</th>
</tr>
</thead>
<tbody>
<tr>
<td>passive subject</td>
<td>6,803</td>
<td>YES</td>
<td>local relation</td>
</tr>
<tr>
<td>indexed gerund</td>
<td>4,430</td>
<td>NO</td>
<td>Tesnière translation</td>
</tr>
<tr>
<td>control, raise, semi-aux</td>
<td>6,020</td>
<td>YES</td>
<td>post-parsing processing (see below)</td>
</tr>
<tr>
<td>others / not covered</td>
<td>5,481</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td>22,734</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Minimal predicate-argument structure output for sentence 36: “Asked to elaborate, Pierre Salinger, White House press secretary, replied, I would say *it’s got to go* thru several more drafts.”
16.2 NP PRO

12,172 NP PRO [row 2] in the Treebank. 5,656 are modpart, 3,095 non-indexed gerunds, 1,598 adverbial phrases of verbs, 268 adverbial phrases of nouns.

16.3 WH trace

113 of the 10,659 WHNP antecedants [row 3] are question pronouns. Over 9,000 are relative pronouns → change of direction if subject or infinitive [example of row 3] is present

But non-subject WH-question pronouns and support verbs need to be treated as “real” non-local dependencies.

Before main parsing is started, the support verb is attached to any lonely participle chunk in the sentence, the WH-pronoun pre-parses with any verb.
17 More Aspects

- Lower Bound
- Robustness
- A Hybrid System
- Functional Dependency Grammar
- More on PP-attachment
### 17.1 Lower Bound

<table>
<thead>
<tr>
<th>Precision and recall measures</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>subj_prec</td>
<td>801 of 936</td>
<td>85.5 %</td>
</tr>
<tr>
<td>subj_recall</td>
<td>752 of 956</td>
<td>78.6 %</td>
</tr>
<tr>
<td>obj_prec</td>
<td>333 of 398</td>
<td>83.6 %</td>
</tr>
<tr>
<td>obj_recall</td>
<td>257 of 391</td>
<td>65.7 %</td>
</tr>
<tr>
<td>nounpp_prec</td>
<td>155 of 227</td>
<td>68.2 %</td>
</tr>
<tr>
<td>verbpp_prec</td>
<td>275 of 722</td>
<td>38.0 %</td>
</tr>
<tr>
<td>ncmo-mod recall</td>
<td>610 of 801</td>
<td>76.1 %</td>
</tr>
<tr>
<td>iobj_recall</td>
<td>87 of 157</td>
<td>55.4 %</td>
</tr>
<tr>
<td>argmod_recall</td>
<td>21 of 41</td>
<td>51.2 %</td>
</tr>
</tbody>
</table>

Table 8: Results of evaluating the Baseline System output on Carroll’s test suite on subject, object and PP-attachment relations
17.2  Robustness

17.2.1  Minimal Parsing

- Least parsing: use finite state technology where reliable: tagging, chunking, head-extraction. Parse only among heads of chunks

- Least long-distance dependencies (LDD): passives, copulas, yes/no-questions need not involve LDDs

- Self-projection: isomorphism of head and projection (A $\rightarrow$ AB or B $\rightarrow$ AB)

- Least nestedness:
  - no unnecessary recursion of same phrase type (e.g. VP $\rightarrow$ VP),
  - express LDD subtrees by a local dedicated dependency relation
17.2.2 Near-full Parsing

- Hand-written grammar: focus on important and manageable distinctions (bottom-up development)

- wide-coverage tag sequence grammar (Abney, 1995; Briscoe and Carroll, 2002),

- exclude or restrict rare, marked and error-prone phenomena. E.g.
  - only nouns seen in the Treebank with several PPs are allowed to have several PPs.
  - subject to the immediate right of a verb (said she) only allowed for verbs seen with a subject to the right in the training corpus (typically verbs of utterance) and only in a comma-delimited or sentence-final context.

- the parser profits from a lean grammar but finds a complete structure spanning the entire sentence in the majority of real-world sentences
Near-full parsing only leads to a very small loss. If an analysis consists of two partial parses, on the dependency relation level only the one, usually high-level relation between the heads of the two partial parses remains unexpressed. The risk of returning “garden path”, locally correct but globally wrong, analyses diminishes with increasing span length.

Already early “naïve” statistical approaches have shown that the problem of grammar size is not solved but even aggravated by a naive probabilistic parser implementation, in which e.g. all CFG rules permitted in the Penn Treebank are extracted. From his 300,000 words training part of the Penn Treebank (Charniak, 1996) obtains more than 10,000 CFG rules, of which only about 3,000 occur more than once. It is therefore necessary to either discard infrequent rules, do manual editing, use a different rule format such as individual dependencies (Collins, 1996) or gain full linguistic control and insight by using a hand-written grammar – each of which sacrifices total completeness.
17.2.3 Underspecification and Disambiguation

Not reductio ad absurdum, but in few, carefully selected areas where

- distinctions do not matter for the task at hand, e.g. intra-base NP ambiguities, such as quantifier scope ambiguities do not matter for a parser aiming at predicate-argument structure
- the disambiguation task is particularly unreliable, e.g. the distinction between verbal particles and prepositions
- inter-annotator agreement is very low (same example)

Conversely, the Penn Treebank annotation is sometimes not specific enough. The parser distinguishes between the reading of the tag \textit{IN} as a complementizer or as a preposition, and disambiguates commas as far as it can, between apposition, subordination and conjunction.

Some typical tagging errors can be robustly corrected by the hand-written grammar. For example, the distinction between verb past tense \textit{VBD} and participle \textit{VBN} is unreliable, but can usually be disambiguated in the parsing process by leaving this tag distinction underspecified for a number of constructions.
17.2.4 Aggressive Pruning

Hard Local Cut

- Very unlikely local structures rarely form part of the most likely global structure
- If a very unlikely local structure forms part of the correct global structure, it chances of getting among the most likely parses are very low
- Very simple to implement
- Biggest gain in complexity reduction, no partial structures at all are built

Fixed Beam Pruning

- Keep a maximum amount of readings for every span
- Only chart entries with equal span can be compared
- “Vorsicht ist die Mutter der Porzellankiste”: do not discard too much
Complexity-Dependent Pruning

- Pruning is only useful from a certain level of ambiguity on. Prune more and more strongly
- Again, only chart entries with equal span can be compared
- In practice, fixed beam pruning performs slightly better → preferred

High Complexity Graceful Degradation
Above a certain complexity threshold measured in total chart entries, only promising paths are pursued.
- It is accepted that some permissible spans will never be found
- Increasing severity based on span length and number of total chart entries
17.2.5 Collecting Partial Results

In a (large) minority of real-world sentences the parser needs to resorts to collecting partial parses in the remaining minority.

- Starting from the most probable longest span, recursively the most probable longest span to left and right is searched.

- “Longest span”: Full span $>$ two subspans $>$ two subspans starting too late / ending too early
17.3 A Hybrid System

**probabilistic vs. statistical: Decision-based Parsing**  Unlike Collins, Charniak, etc. this parser is not probabilistic, but based on decision probabilities.

Not the sum of $p$ of possible parses, but the sum of $p$ of possible decisions at a decision point add to 1. Whether to attach or not (in shift/reduce parlance: to reduce or to shift) is e.g. a decision.

The probability-based score of a parse is the product of the (normalized) decisions taken during parsing. From a practical viewpoint, knowing the probability of a certain rule expansion per se is of little interest.

**statistical vs. rule-based**  the most obvious way in which Pro3Gres is a hybrid
Unlike formal grammars to which post-hoc statistical disambiguators can be added, Pro3Gres has been designed to be hybrid, carefully distinguishing between tasks that can best be solved by finite-state methods, rule-based methods and statistical methods.
shallow vs. deep the designing philosophy for Pro3Gres has been to stay as shallow as possible to obtain reliable results at each level.

Treebank constituency vs. DG the observation that a DG that expresses grammatical relations is more informative, but also more intuitive to interpret for a non-expert, and that Functional DG can avoid a number of LDD types has made DG the formalism of our choice. For lexicalizing the grammar, a partial mapping from the largest manually annotated corpus available, the Penn Treebank, was necessary, exhibiting a number of mapping challenges.

history-based vs. mapping-based Pro3Gres is not a parse-history-based approach. Instead of manually selecting what goes into the history, as is usually done, we manually select how to linguistically meaningfully map Treebank structures onto dependency relations by the use of mapping patterns adapted from (Johnson, 2002).
**local subtress vs. DOP**  psycholinguistic experiments and Data-Oriented Parsing (DOP) (Bod, Scha, and Sima’an, 2003) suggest that people store subtrees of various sizes, from two-word fragments to entire sentences. But (Goodman, 2003) suggests that the large number of subtrees can be reduced to a compact grammar that makes DOP parsing computationally tractable. In Pro3Gres, a subset of non-local fragments which, based on linguistic intuition are especially important, are used.

**generative vs. structure-generating**  DG generally, although generative in the sense that connected complete structures are generated, is not generative in the sense that it is always guaranteed to terminate if used for random generation of language. Since a complete or partial hierarchical structure that follows CFG assumptions due to the employed grammar is built up for each sentence. Pro3Gres’ constraint to allow each complement dependency type only once per verb can be seen as a way of rendering it generative in practice.
syntax vs. semantics  instead of using a back-off to tags (Collins, 1999), semantic
classes, Wordnet for nouns and Levin classes for verbs, are used, in the hope that they
better manage better to express selectional restrictions than tags. Practical
experiments have shown, however, that, in accordance to (Gildea, 2001) on
head-lexicalisation, there is almost no increase in performance.
17.4 Functional Dependency Grammar

- Integration of chunking and parsing (Abney, 1995), where a chunk largely corresponds to a *nucleus* (Tesnière, 1959)

- In an endocentric theory projection can never fail

- No Empty Nodes: The fact that traditional DG does not know empty nodes allows a DG parser to use the efficient $O(n^3)$ CYK algorithm.

- Only Content Words are Nuclei: Only content words can be nuclei in a traditional DG. This means that empty units, empty complementizers and empty relative pronouns pose no problem for DG as they are optional, non-head material.

- No External Argument, ID/LP: Moved clauses are mostly PPs or clausal complements of verbs of utterance. Only verbs of utterance allow subject-verb inversion in affirmative clauses. Our hand-written grammar provides rules with appropriate restrictions for them, allowing an inversion of the “canonical” dependency direction under well-defined conditions, distinguishing between
ordre linéaire (linear precedence(LP)) and ordre structural (immediate dominance(ID)). Fronted positions are available locally to the verb in a theory that does not posit a distinction between internal and external arguments.

• Exploiting Functional DG Labels: The fact that dependencies are often labeled is a main difference between DG and constituency. We exploit this by using dedicated labels to model a range of constituency LDDs, relations spanning several constituency levels, including empty nodes and functional Penn Treebank labels, by a purely local DG relation.

• Monostratalism and Functionalism While multistratal DGs exist and several dependency levels can be distinguished (Mel’čuk, 1988) we follow a conservative view close to the original (Tesnière, 1959), which basically parses directly for a simple LFG f-structure without needing a c-structure detour.

• Graphs DG theory often conceives of DG structures as graphs instead of trees (Hudson, 1984). A statistical lexicalized post-processing module in Pro3Gres transforms selected subtrees into graphs, (e.g. control).
• Transformation to Semantic Layer Pro3Gres is currently being adapted to a Question Answering system specifically targeted at technical domains (Rinaldi et al., 2004). Mapping from the syntactic layer to a semantic layer is partly simplified (Rinaldi et al., 2002).

• Tesnière’s Translations The possible functional changes of a word called translations (Tesnière, 1959) are an exception to endocentricity. Participles may function as adjectives upwards in the tree (Western industrialized/VBN countries), gerunds may function as nouns (after winning/VBG the race), or in nounless NPs such as the poor, adjectives function as nouns.
17.5 More on PP-attachment

17.5.1 Integrating Hindle and Rooth

\[ LA = \log_2 \left( \frac{\#(v,p) + \#(V,p)}{\#(v) + 1} \cdot \frac{\#(n, NULL) + \#(N, NULL)}{\#(n) + 1} \right) \]  

integrating into the backoff hierarchy \(\rightarrow\) drop the smoothing factors

\[ LA' = \log_2 \left( \frac{\#(v,p)}{\#(v)} \cdot \frac{\#(n, NULL)}{\#(n)} \right) \]  

\[ p(p|v) \approx \frac{\#(v,p)}{\#(v)} \cdot \frac{\#(n, NULL)}{\#(n)} \]  

\[ p(p|n) \approx \frac{\#(n,p)}{\#(n)} \]
But \( p(attach(v, p)|v, p) \neq p(p|v) \)

\[ p(attach(v, p)|v, p) = 1 \text{ in Hindle & Rooth’s unambiguous data} \]

\[ [Schneider] \quad p(attach(v, p)|v, p) = \frac{\#(attach(v, p), v, p)}{\#(v, p)} \quad (13) \]

\[ [Hindle & Rooth] \quad p(p, attach(v, p)|v) = \frac{\#(attach(v, p), v, p)}{\#(v)} \quad (14) \]

... “divided by too big a factor”

\[ \#(v, p) = \#(v) \cdot p(p|v) = \#(v) \cdot \frac{\#(v, p)}{\#(v)} \quad (15) \]

\[ p(attach(v, p)|v, p) = \frac{\#(attach(v, p), v, p)}{\#(v, p)} = \frac{\#(attach(v, p), v, p)}{\#(v) \cdot \frac{\#(v, p)}{\#(v)}} = 1 \quad (16) \]
Solutions:

- Either estimate $\frac{\#(v,p)}{\#(v)}$ and $\frac{\#(n,p)}{\#(n)}$ from a different, ambiguous corpus

- Or rely on the fact that also $p(p|n)$ is “divided by too big a factor”:

  If we assume that $p(attach(n, prep) + p(attach(v, prep)) = 1$:

  \[
p(attach(v, p)) = \frac{p_x(attach(v, p))}{p_x(attach(v, p) + p_x(attach(n, p))}, p_x = \text{non-normalised prob.} \tag{17}
  \]

  \[
p(attach(n, p)) = \frac{p_x(attach(n, p))}{p_x(attach(v, p) + p_x(attach(n, p))}, p_x = \text{non-normalised prob.} \tag{18}
  \]
\( p_x(n) \) is available at parse time when the decision about the probability for verb-attachment arises, as \( n \) is the object of the verb. \( :-( \)

But \( p_x(v) \) and \( v \) are unknown when a probability for noun-attachment is needed \( :-( \)

Therefore, it is summed over all verbs for calculating the noun-attachment probabilities
APPENDIX: tgrep sample query

```
VP << '/VB/ < (/NP$/|NP-PRD <- '/NN/|PRP|WDT|WP|CD))
VP << '/VB/ < (/NP$/|NP-PRD < 1 (NP <- '/NN/|PRP|WDT|WP|CD))
VP << '/VB/ < (/NP$/|NP-PRD < 1 (NP<1(NP <- '/NN/|PRP|WDT|WP|CD)))
VP << '/VB/ < (/NP$/|NP-PRD < 1 (NP<(NP<1(NP <- '/NN/|PRP|WDT|WP|CD))))

(VP << '/VB/ < (S <(NP-SBJ < /-NONE-/) <(NP-PRD< ( '/NN/|PRP|WDT|WP|CD))))
(VP << '/VB/ < (S <(NP-SBJ < /-NONE-/) <(NP-PRD<1(NP< ( '/NN/|PRP|WDT|WP|CD))))
(VP << '/VB/ < (S <(NP-SBJ < /-NONE-/) <(NP-PRD<1(NP<1(NP< ( '/NN/|PRP|WDT|WP|CD))))
(VP << '/VB/ < (S <(NP-SBJ < /-NONE-/) <(NP-PRD<1(NP<1(NP< ( '/NN/|PRP|WDT|WP|CD))))
```

Figure 3: Currently used tgrep queries for extracting the verb–object relation
• DG is binary & in Chomsky Normal Form → CYK → $0(n^3)$

• CYK Parsing: bottom-up parallel processing

• my “chartdata-driven” CYK implementation
  1. Add all terminals to chart
  2. Loop: foreach chart entry $X\lambda_i.\lambda_k.[i - k]$
     foreach chart entry $Y\lambda_j.[k - j]$ # adjacent
     if ¬ tried($X, Y$)
     foreach $Z \rightarrow X, Y$ assert $Z[i - j]$ to chart (for next Loop)
     else assert tried($X, Y$)
  3. If any rule was successful, prune and then Loop again, else terminate.

• pruning: If in a Loop more than $m$ chart entries are created, then for every span with more than $n$ readings in the chart, only keep the most probable $n/2$ entries.

• auxiliary charts: remember all tried chart pairs. Remember all computed probabilities.