THE PENN TREEBANK: ANNOTATING PREDICATE ARGUMENT STRUCTURE

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz,
Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, Britta Schasberger

Department of Computer and Information Science
University of Pennsylvania
Philadelphia, PA, USA

ABSTRACT

The Penn Treebank has recently implemented a new syntactic annotation scheme, designed to highlight aspects of predicate-argument structure. This paper discusses the implementation of crucial aspects of this new annotation scheme. It incorporates a more consistent treatment of a wide range of grammatical phenomena, provides a set of coindexed null elements in what can be thought of as "underlying" position for phenomena such as wh-movement, passive, and the subjects of infinitival constructions, provides some non-context free annotational mechanism to allow the structure of discontinuous constituents to be easily recovered, and allows for a clear, concise tagging system for some semantic roles.

1. INTRODUCTION

During the first phase of the The Penn Treebank project [10], ending in December 1992, 4.5 million words of text were tagged for part-of-speech, with about two-thirds of this material also annotated with a skeletal syntactic bracketing. All of this material has been hand corrected after processing by automatic tools. The largest component of the corpus consists of materials from the Dow-Jones News Service; over 1.6 million words of this material has been hand parsed, with an additional 1 million words tagged for part of speech. Also included is a skeletally parsed version of the Brown corpus, the classic million word balanced corpus of American English [5, 6]. hand-retagged using the Penn Treebank tagset.

The level of syntactic analysis annotated during this phase of this project was an extended and somewhat modified form of the skeletal analysis which has been produced by the treebanking effort in Lancaster, England [7]. The released materials in the current Penn Treebank, although still in very preliminary form, have been widely distributed, both directly by us, on the ACL/DCI CD-ROM, and now on CD-ROM by the Linguistic Data Consortium; it has been used for purposes ranging from serving as a gold-standard for parser testing to serving as a basis for the induction of stochastic grammars to serving as a basis for quick lexicon induction.

Many users of the Penn Treebank now want forms of annotation richer than provided by the project's first phase, as well as an increase in the consistency of the preliminary corpus. Some would also like a less skeletal form of annotation, expanding the essentially context-free analysis of the current treebank to indicate non-contiguous structures and dependencies. Most crucially, there is a strong sense that the Treebank could be of much more use if it explicitly provided some form of predicate-argument structure. The desired level of representation would make explicit at least the logical subject and logical object of the verb, and indicate, at least in clear cases, how subconstituents are semantically related to their predicates. Such a representation could serve as both a starting point for the kinds of SEMEVAL representations now being discussed as a basis for evaluation of human language technology within the ARPA HLT program, and as a basis for "glass box" evaluation of parsing technology.

The ongoing effort [1] to develop a standard objective methodology to compare parser outputs across widely divergent grammatical frameworks has now resulted in a widely supported standard for parser comparison. On the other hand, many existing parsers cannot be evaluated by this metric because they directly produce a level of representation closer to predicate-argument structure than to classical surface grammatical analysis. Hand-in-hand with this limitation of the existing Penn Treebank for parser testing is a parallel limitation for automatic methods for parser training for parsers based on deeper representations. There is also a problem of maintaining consistency with the fairly small (less than 100 page) style book used in the the first phase of the project.

2. A NEW ANNOTATION SCHEME

We have recently completed a detailed style-book for this new level of analysis, with consensus across annotators about the particulars of the analysis. This project has taken about eight months of ten-hour a week effort across a significant subset of all the personnel of the Penn Treebank. Such a stylebook, much larger, and much more fully specified than our initial stylebook, is a prerequisite for high levels of inter-annotator agreement. It is our hope that such a stylebook will also alleviate much of the need for extensive cross-talk between annotators during the annotation task, thereby increasing throughput as well. To ensure that the rules of this new stylebook remain in force, we are now giving annotators about 10% overlapped material to evaluate inter-annotator consistency throughout this new project.

We have now begun to annotate this level of structure editing the present Penn Treebank; we intend to automatically extract a bank of predicate-argument structures intended at the very least for parser evaluation from the resulting annotated corpus.

The remainder of this paper will discuss the implementation of each of four crucial aspects of the new annotation scheme, as well as notational devices to allow predicate-argument structure to be recovered in the face of conjoined structure involving gapping, where redundant syntactic structure within a conjoined structure is deleted. In particular, the new scheme:

1. Incorporates a consistent treatment of related grammatical phenomena. The issue here is not that the representation be "correct" given some theoretical analysis or other, but merely that instances of what are descriptively the same phenomenon be represented similarly. In particular, the notation should make it easy to automatically recover predicate-argument structure.
2. Provides a set of null elements in what can be thought of as "underlying" position for phenomena such as wh-movement, passive, and the subjects of infinitival constructions. These null elements must be co-indexed with the appropriate lexical material.
3. Provides some non-context free annotational mechanism to allow the structure of discontinuous constituents to be easily recovered.
4. Allows for a clear, concise distinction between verb arguments and adjuncts where such distinctions are clear, with some easy-to-use notational device to indicate where such a distinction is somewhat murky.

Our first step, just now complete, has been to produce a detailed style-book for this new level of analysis, with consensus across annotators about the particulars of the analysis. This project has taken about eight months of ten-hour a week effort across a significant subset of all the personnel of the Penn Treebank. It has become clear during the first stage of the project that a much larger, much more fully specified stylebook than our initial stylebook is a prerequisite for high levels of inter-annotator agreement. It is our hope that such a stylebook will also alleviate much of the need for extensive cross-talk between annotators during the annotation task, thereby increasing throughput as well. To ensure that the rules of this new stylebook remain in force, we intend to give annotators about 10% overlapped material to evaluate inter-annotator consistency throughout this new project.

The remainder of this paper discusses the implementation of each of the four points above, as well as notational devices to allow predicate-argument structure to be recovered in the face of conjoined structure involving gapping, where redundant syntactic structure within a conjoined structure is deleted.

3. CONSISTENT GRAMMATICAL ANALYSES

The current treebank materials suffer from the fact that differing annotation regimes are used across differing syntactic categories. To allow easy automatic extraction of predicate-argument structure in particular, these differing analyses must be unified. In the original annotation scheme, adjective phrases that serve as sentential predicates have a different structure than VPs, causing sentential adverbs which occur after auxiliaries introducing the ADJP to attach under VP, while sentential adverbs occurring after auxiliaries introducing VPs occur under S. In the current treebank, copular *be* is treated as a main verb, with predicate adjective or prepositional phrases treated as complements to that verb.

In the new stylebook, the predicate is either the lowest (right-most branching) VP or the phrasal structure immediately under copular BE. In cases when the predicate cannot be identified by those criteria (e.g. in "small clauses" and some inversion structures), the predicate phrase is tagged *-PRD* (PReDicate).

```
(S (NP-SBJ I)
   (VP consider
       (S (NP-SBJ Kris)
          (NP-PRD a fool))))

(SQ Was
    (NP-SBJ he)
    (ADVP-TMP ever)
    (ADJP-PRD successful)
    ?)
```

Note that the surface subject is always tagged *-SBJ* (SuBJect), even though this is usually redundant because the subject can be recognized purely structurally. The *-TMP* tag here marks time (TeMPoral) phrases. Our use of "small clauses" follows one simple rule: every S maps into a single predication, so here the predicate-argument structure would be something like

```
consider(I, fool(Kris)).
```

4. ARGUMENT-ADJUNCT STRUCTURE

In a well developed predicate-argument scheme, it would seem desirable to label each argument of a predicate with an appropriate semantic label to identify its role with respect to that predicate. It would also seem desirable to distinguish between the *arguments* of a predicate, and *adjuncts* of the predication. Unfortunately, while it is easy to distinguish arguments and adjuncts in simple cases, it turns out to be very difficult to consistently distinguish these two categories for many verbs in actual contexts. It also turns out to be very difficult to determine a set of underlying semantic roles that holds up in the face of a few paragraphs of text. In our new annotation scheme, we have tried to come up with a middle ground which allows annotation of those distinctions that seem to hold up across a wide body of material. After many attempts to find a reliable test to distinguish between arguments and adjuncts, we have abandoned structurally marking this difference. Instead, we now label a small set of clearly distinguishable roles, building upon syntactic distinctions only when the semantic intuitions are clear cut. Getting annotators to consistently apply even the small set of distinctions we will discuss here is fairly difficult.

In the earlier corpus annotation scheme, We originally used only standard syntactic labels (e.g. NP, ADVP, PP, etc.)

Tag	Marks:
	Text Categories
-HLN	headlines and datelines
-LST	list markers
-TTL	titles
	Grammatical Functions
-CLF	true clefts
-NOM	non NPs that function as NPs
-ADV	clausal and NP adverbials
-LGS	logical subjects in passives
-PRD	non VP predicates
-SBJ	surface subject
-TPC	topicalized and fronted constituents
-CLR	closely related - see text
	Semantic Roles
-VOC	vocatives
-DIR	direction & trajectory
-LOC	location
-MNR	manner
-PRP	purpose and reason
-TMP	temporal phrases

Figure 1: Functional Tags

for our constituents – in other words, every bracket had just one label. The limitations of this became apparent when a word belonging to one syntactic category is used for another function or when it plays a role which we want to be able to identify easily. In the present scheme, each constituent has at least one label but as many as four tags, including numerical indices. We have adopted the set of functional tags shown in Figure 2 for use within the current annotation scheme. NPs and Ss which are clearly arguments of the verb are unmarked by any tag. We allow an open class of other cases that individual annotators feel strongly should be part of the VP. These cases are tagged as -*CLR* (for CLosely Related); they are to be semantically analyzed as adjuncts. This class is an experiment in the current tagging; constituents marked -*CLR* typically correspond to Quirk et al's [11] class of predication adjuncts. At the moment, we distinguish a handful of semantic roles: direction, location, manner, purpose, and time, as well as the syntactic roles of surface subject, logical subject, and (implicit in the syntactic structure) first and second verbal objects.

5. NULL ELEMENTS

One important way in which the level of annotation of the current Penn Treebank exceeds that of the Lancaster project is that we have annotated null elements in a wide range of cases. In the new annotation scheme, we co-index these null elements with the lexical material for which the null element stands. The current scheme happens to use two symbols for null elements: **T**, which marks WH-movement and topicalization, and *** which is used for all other null elements, but this distinction is not very important. Co-indexing of null elements is done by suffixing an integer to non-terminal categories (e.g. `NP-10, VP-25`). This integer serves as an id number for the constituent. A null element itself is followed by the id number of the constituent with which it is co-indexed. We use *SBARQ* to mark WH-questions, and *SQ* to mark auxiliary inverted structures. We use the WH-prefixed labels, *WHNP, WHADVP, WHPP,* etc., only when there is WH-movement; they always leave a co-indexed trace. Crucially, the predicate argument structure can be recovered by simply replacing the null element with the lexical material that it is co-indexed with:

```
(SBARQ (WHNP-1 What)
       (SQ is
           (NP-SBJ Tim)
           (VP eating
               (NP *T*-1)))
       ?)

Predicate Argument Structure:
eat(Tim, what)
```

In passives, the surface subject is tagged -*SBJ*, a passive trace is inserted after the verb, indicated by (NP *), and co-indexed to the surface subject (i.e. logical object). The logical subject by-phrase, if present, is a child of VP, and is tagged -*LGS* (LoGical Subject). For passives, the predicate argument structure can be recovered by replacing the passive null element with the material it is co-indexed with, and treating the NP marked -*LGS* as the subject.

```
(S (NP-SBJ-1 The ball)
   (VP was
       (VP thrown
           (NP *-1)
           (PP by
               (NP-LGS Chris)))))

Predicate Argument Structure:
throw(Chris, ball)
```

The interpretation rules for passives and WH-phrases interact correctly to yield the predicate argument structures for complex nestings of WH-questions and passives.

```
(SBARQ (WHNP-1 Who)
       (SQ was
           (NP-SBJ-2 *T*-1)
           (VP believed
```

```
            (S (NP-SBJ-3 *-2)
               (VP to
                   (VP have
                       (VP been
                           (VP shot
                               (NP *-3))))))))
      ?)
```

```
Predicate Argument Structure:
believe(*someone*, shoot(*someone*, Who))
```

A null element is also used to indicate which lexical NP is to be interpreted as the null null subject of an infinitive complement clause; it is co-indexed with the controlling NP, based upon the lexical properties of the verb.

```
(S (NP-SBJ-1 Chris)
   (VP wants
       (S (NP-SBJ *-1)
          (VP to
              (VP throw
                  (NP the ball))))))
```

```
Predicate Argument Structure:
wants(Chris, throw(Chris, ball))
```

We also use null elements to allow the interpretation of other grammatical structures where constituents do not appear in their default positions. Null elements are used in most cases to mark the fronting (or "topicalization" of any element of an S before the subject (except in inversion). If an adjunct is topicalized, the fronted element does not leave a trace since the level of attachment is the same, only the word order is different. Topicalized arguments, on the other hand, always are marked by a null element:

```
(S (NP-TPC-5 This)
   (NP-SBJ every man)
   (VP contains
       (NP *T*-5)
       (PP-LOC within
               (NP him))))
```

Again, this makes predicate argument interpretation straightforward, if the null element is simply replaced by the constituent to which it is co-indexed.

Similarly, if the predicate has moved out of VP, it leaves a null element *T* in the VP node.

```
(SINV (VP-TPC-1 Marching
                (PP-CLR past
                        (NP the reviewing stand)))
      (VP were
          (VP *T*-1))
      (NP-SBJ 500 musicians))
```

TAG	Mnemnonic
ICH	Interpret Constituent Here
PPA	Permanent Predictable Ambiguity
RNR	Right Node Raising
EXP	EXPletive

Figure 2: The four forms of pseudo-attachment

Here, the *SINV* node marks an inverted S structure, and the *-TPC* tag (ToPiC) marks a fronted (topicalized) constituent; the *-CLR* tag is discussed below.

6. DISCONTINUOUS CONSTITUENTS

Many otherwise clear argument/adjunct relations in the current corpus cannot be recovered due to the essentially context-free representation of the current Treebank. For example, currently there is no good representation for sentences in which constituents which serve as complements to the verb occur after a sentential level adverb. Either the adverb is trapped within the VP, so that the complement can occur within the VP, where it belongs, or else the adverb is attached to the S, closing off the VP and forcing the complement to attach to the S. This "trapping" problem serves as a limitation for groups that currently use Treebank material to semiautomatically derive lexicons for particular applications.

To solve "trapping" problems and annotation of non-contiguous structure, a wide range of phenomena of the kind discussed above can be handled by simple notational devices that use co-indexing to indicate discontinuous structures. Again, an index number added to the label of the original constituent is incorporated into the null element which shows where that constituent should be interpreted within the predicate argument structure.

We use a variety of null elements to show show how non-adjacent constituents are related; we refer to such constituents as "pseudoattached". There are four different types of pseudo-attach, as shown in Figure 1; the use of each will be explained below:

The **ICH** pseudo-attach is used for simple extraposition, solving the most common case of "trapping":

```
(S (NP-SBJ Chris)
   (VP knew
       (SBAR *ICH*-1)
       (NP-TMP yesterday)
       (SBAR-1 that
               (S (NP-SBJ Terry)
                  (VP would
                      (VP catch
                          (NP the ball)))))))
```

Here, the clause *that Terry would catch the ball* is to be interpreted as an argument of *knew.*

The **PPA** tag is reserved for so-called "permanent predictable ambiguity", those cases in which one cannot tell where a constituent should be attached, even given context. Here, annotators attach the constituent at the more likely site (or if that is impossible to determine, at the higher site) and pseudo-attach it at all other plausible sites using using the **PPA** null element. Within the annotator workstation, this is done with a single mouse click, using pseudo-move and pseudo-promote operations.

```
(S (NP-SBJ I)
       (VP saw
           (NP (NP the man)
               (PP *PPA*-1))
           (PP-CLR-1 with
                     (NP the telescope))))
```

The **RNR** tag is used for so-called "right-node raising" conjunctions, where the same constituent appears to have been shifted out of both conjuncts.

```
(S But
   (NP-SBJ-2 our outlook)
   (VP (VP has
           (VP been
               (ADJP *RNR*-1)))
       ,
       and
       (VP continues
           (S (NP-SBJ *-2)
              (VP to
                  (VP be
                      (ADJP *RNR*-1)))))
       ,
       (ADJP-1 defensive)))
```

So that certain kinds of constructions can be found reliably within the corpus, we have adopted special marking of some special constructions. For example, extraposed sentences which leave behind a semantically null "it" are parsed as follows, using the **EXP** tag:

```
(S (NP-SBJ (NP It)
           (S *EXP*-1))
   (VP is
       (NP a pleasure))
   (S-1 (NP-SBJ *)
        (VP to
            (VP teach
                (NP her)))))

Predicate Argument Structure:
pleasure(teach(*someone*, her))
```

Note that "It" is recognized as the surface subject, and that the extraposed clause is attached at S level and adjoined to "it" with what we call **EXP**-attach. The **EXP** is automatically co-indexed by our annotator workstation software to the postposed clause. The extraposed clause is interpreted as the subject of *a pleasure* here; the word *it* is to be ignored during predicate argument interpretation; this is flagged by the use of a special tag.

7. CONJUNCTION AND GAPPING

In general, we use a Chomsky adjunction structure to show coordination, and we coordinate structures as low as possible. We leave word level conjunction implicit; two single word NP's or VP's will have only the higher level of structure. If at least one of the conjoined elements consists of more than one word, the coordination is made explicit. The example that follows shows two conjoined relative clauses; note that relative clauses are normally adjoined to the antecedent NP.

```
(S (NP-SBJ Terry)
   (VP knew
       (NP (NP the person)
           (SBAR (SBAR (WHNP-1 who)
                       (S (NP-SBJ T-1)
                          (VP threw
                              (NP the ball))))
                 and
                 (SBAR (WHNP-2 who)
                       (S (NP-SBJ T-2)
                          (VP caught
                              (NP it))))))

Predicate Argument Structure:
(knew Terry (person (and (threw *who* ball)
                         (caught *who* it))))
```

Conditional, temporal, and other such subordinate clauses, like other adjuncts, are normally attached at S-level.

The phenomenon of gapping provides a major challenge to our attempt to provide annotation which is sufficient to allow the recovery of predicate argument for whatever structure is complete within a sentence. We have developed a simple notational mechanism, based on structural *templates*, which allows the predicate argument structure of gapped clauses to be recovered in most cases when the full parallel structure is within the same clause. In essence, we use the complete clause as a template and provide a notation to allow arguments to be mapped from the gapped clause onto that template. In the template notation, we use an equal sign to indicate that constituent NP=1 should be mapped over NP-1 in the largest conjoined structure that NP-1 and NP=1 both occur in. A variety of simple notational devices, which we will not discuss here, extend this notation to handle constituents that occur in one branch of the conjunct, but not the other.

```
(S (S (NP-SBJ-1 Mary)
      (VP likes
          (NP-2 Bach)))
   and
   (S (NP-SBJ=1 Susan)
      ,
      (NP=2 Beethoven)))
```

```
Predicate Argument Structure:
like (Mary, Bach) and like (Susan,Beethoven)

(S (S (NP-SBJ John)
      (VP gave
          (NP-1 Mary)
          (NP-2 a book)))
    and
    (S (NP=1 Bill)
       (NP=2 a pencil)))

(S (S (NP-SBJ I)
      (VP eat
          (NP-1 breakfast)
          (PP-TMP-2 in
                    (NP the morning))))
   and
   (S (NP=1 lunch)
      (PP-TMP=2 in
          (NP the afternoon))))
```

We do not attempt to recover structure which is outside a single sentence. We use the tag FRAG for those pieces of text which appear to be clauses, but lack too many essential elements for the exact structure to be easily determined. Obviously, predicate argument structure cannot be extracted from FRAG's.

```
Who threw the ball?  Chris, yesterday.

(FRAG (NP Chris)
      ,
      (NP-TMP yesterday))

What is Tim eating? Mary Ann thinks chocolate.

(S (NP-SBJ Mary Ann)
   (VP thinks
       (SBAR 0
             (FRAG (NP chocolate)))))
```

8. CONCLUSION

We are now beginning annotation using this new scheme. We believe that this revised form of annotation will provide a corpus of annotated material that is useful for training stochastic parsers on surface syntax, for training stochastic parsers that work at one level of analysis beyond surface syntax, and at the same time provide a consistent database for use in linguistic research.

References

1. Black, E., Abney, S., Flickenger, F., Grishman, R., Harrison, P., Hindle, D., Ingria, R., Jelinek, F., Klavans, J., Liberman, M., Marcus, M., Roukos, S., Santorini, B., and Strzalkowski, T., 1991. A procedure for quantitatively comparing the syntactic coverage of English grammars. In *Proceedings of the Fourth DARPA Speech and Natural Language Workshop, February 1991.*
2. Black, E., Jelinek, F., Lafferty, J., Magerman, D.M., Mercer, R., and Roukos, S. 1992. Towards history-based grammars: Using Richer Models for Probabilistic parsing. In *Proceedings of the 31th Annual Meeting of the Association for Computational Linguistics.*
3. Brill, E., Marcus, M., 1992. Automatically acquiring phrase structure using distributional analysis. In *Proceedings of the DARPA Speech and Natural Language Workshop, February 1992.*
4. Brill, E., 1993. Automatic grammar induction and parsing free text: a transformation-based approach. In *Proceedings of the 31th Annual Meeting of the Association for Computational Linguistics.*
5. Francis, W., 1964. *A standard sample of present-day English for use with digital computers. Report to the U.S Office of Education on Cooperative Research Project No. E–007.* Brown University, Providence.
6. Francis, W. and Kučera, H., 1982. *Frequency analysis of English usage. Lexicon and grammar.* Houghton Mifflin, Boston.
7. Garside, R., Leech, G., and Sampson, G., 1987. *The computational analysis of English. A corpus-based approach.* Longman, London.
8. Hindle, D., and Rooth, M., 1993. Structural Ambiguity and Lexical Relations. *Computational Linguistics*, Vol 19.
9. D. Magerman and M. Marcus, 1991. PEARL — A Probabilistic Chart Parser, In *Proceedings, Fifth Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, Berlin, April 1991.
10. Marcus, M., Santorini, B., Marcinkiewicz, M.A., 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, Vol 19.
11. Quirk, R., Greenbaum, S., Leech, G., and Svartvik, J., 1985. *A comprehensive grammar of the English language*, Longman, London.