Semi-automatic syntactic and semantic corpus annotation with a deep parser *

Mary D. Swift, Myroslava O. Dzikovska, Joel R. Tetreault, James F. Allen

Department of Computer Science
University of Rochester
Rochester, NY 14627 USA
{swift, myros, tetreaul, james}@cs.rochester.edu

Abstract

We describe a semi-automatic method for linguistically rich corpus annotation using a broad-coverage deep parser to generate syntactic structure, semantic representation and discourse information for task-oriented dialogs. The parser-generated analyses are checked by trained annotators. Incomplete coverage and incorrect analyses are addressed through lexicon and grammar development, after which the dialogs undergo another cycle of parsing and checking. Currently we have 85% correct annotations in our emergency rescue task domain and 70% in our medication scheduling domain. This iterative process of corpus annotation allows us to create domain-specific gold-standard corpora for test suites and corpus-based experiments as part of general system development.

1. Introduction

Linguistically annotated corpora are valuable resources for computational linguistics research, but time-consuming and costly to create so it is desirable to automate the annotation process while retaining as much linguistic detail as possible. We address this problem with a method for semi-automatic, linguistically rich corpus annotation using a broad-coverage deep parser.

Our annotation method has two stages: transcription preprocessing for input to the parser, followed by a checking and update loop. The preprocessing stage begins with transcriptions from task-oriented spoken dialogs, which are prepared for parser input as described in Section 3.1 and then parsed to generate linguistic analyses for both syntax and semantics, as detailed in Section 3.2. In the second stage the parser-generated analyses are checked by trained annotators, the lexicon and grammar are updated to handle analyses judged to be incorrect, and the dialogs are parsed and checked again.

Our approach gives a methodology for quickly developing linguistically annotated corpora in new domains as part of general system development. Our lexicon includes a mechanism to easily and transparently augment the entries with domain-specific restrictions, allowing the correct parse to be selected automatically most of the time, improving disambiguation accuracy compared to a parser without domain information (Dzikovska et al., 2003). Using domain-specific parsing to automatically select between alternative analyses helps automate this part of the annotation process, which has traditionally relied on human annotators. Our corpus coverage and annotation improves as the grammar develops, so work on parser development simultaneously contributes to producing an annotated corpus.

2. Background

Manual corpus annotation is time-consuming, costly and prone to inconsistencies even from a single annotator. To improve efficiency, corpus builders have tried to automate corpus annotation as much as possible. The annotation tasks that lend themselves most readily to automation are structural, so parsers are often used to perform superficial syntactic analysis. The Penn Treebank (Marcus et al., 1993) and the Italian Syntax-Semantics Corpus (ISST) (Barsotti et al., 2000) use a shallow parser to generate phrase-structure analyses that are then checked by trained annotators. Both corpora add some deeper analysis manually, such as syntactic function labels (e.g., subject, object) and syntactic dependencies.

Interactive approaches use a parser to automatically generate possible analyses from which an annotator selects, such as the German TiGer corpus (Brants and Plaehn, 2000), the Redwoods corpus, providing HPSG analyses in the Verbmobil domain (Oepen et al., 2002), and LFG analyses for the Penn Treebank, e.g., (Cahill et al., 2002).

The less structural the analysis, the more difficult it is to generate automatically, so semantic, pragmatic and discourse information is less commonly provided in corpora. The Penn Treebank distinguishes some hand-annotated semantic roles under the Treebank II annotation style (Marcus et al., 1994), and ISST includes hand-annotated word senses from ItalWordNet. Only the Redwoods corpus provides automatically generated semantic annotation with an underspecified Minimal Recursion Semantics (Copestake et al., 1997) representation.

3. Corpus building method

Our method is designed to build and improve the parsed corpus as the grammar develops and improves, with diminishing amount of hand-checking necessary. There are two stages: preparation of the transcripts for parser input (Section 3.1) and a checking and update loop (Section 3.2).

Our corpora consist of transcribed dialogs between human interlocutors engaged in collaborative problem solving tasks. We report here on our results from two domains. The Monroe domain (Stent, 2001) consists of task-oriented dialogs between human participants designed to encourage collaborative problem-solving and mixed-initiative interaction in simulated rescue operation situations in which a controller receives emergency calls and is assisted by a system or another person in formulating a plan to handle emergencies ranging from weather emergencies to civil disorder. The medication scheduling domain (Ferguson et al.,

* In Proceedings of LREC'04, Lisbon

2002) consists of dialogs in which a participant interacts with another person playing the role of the system to work out medication scheduling based on different patient needs and situation parameters.

3.1. Transcription preparation

To prepare the transcribed speech for parser input we have to do some utterance resegmentation followed by manual annotation for disfluencies and ungrammatical or incomplete utterances.

The segmentation in the original dialog transcripts separates utterances by user turn. Further segmentation is needed on utterances that are conjunctions of several main clauses that are coherent by themselves. Utterances are split if there is an explicit coordinating conjunction between two or more complete clauses, that is, each clause has subject-verb or subject-verb-object format and is joined by coordinating conjunctions *and*, *but*, *or*, etc. Only clauses with a subject of *you* or *we* are exempt, so there can be implicit subjects in preceding clauses. In the case of utterances without explicit coordinating conjunctions, each utterance must be able to stand by itself. Typically these are complete sentences accompanied by short acknowledgment phrases such as *[Wait a minute] [I misunderstood]* and *[Right] [We still have a road crew at the airport]*.

Handling speech repairs and incomplete utterances presents additional difficulty to the problem of parsing human speech. We annotated our corpus with information about disfluencies as follows. Words or phrases are marked as disfluencies (indicated in square brackets) if the phrase is repaired (e.g., *[We're gonna send the digging crew] we're gonna send the road crew from RGE to Elmwood bridge*) or repeated (e.g., *[Can you] can you go over the thing for me again?*) immediately after in the same utterance. The original utterance is retained in the corpus, but the input to the parser does not include the disfluencies (as shown in the Corpus Tool display in Figure 2).

We label utterances that are incomplete or ungrammatical but do not currently parse them. Incomplete utterances meet one or more of the following criteria: 1) both subject and verb are present, but at least one more constituent is required to make sense, e.g., *It and that is*; 2) the utterance consists of a single word that is not an acknowledgment and does not qualify as an elliptical response to a previous utterance; 3) the utterance ends with a half-finished word, e.g., *Actually it's right a ab*. Ungrammatical utterances do not fit the above criteria for incomplete utterances but meet one of the following criteria: 1) required constituents are missing, such as articles or prepositions e.g., *So ambulance sends generator*; 2) wording is incorrect, e.g., *As terms of crews I have two road crews*; 3) there are morphological errors, such as incorrect agreement marking.

The prepared utterances are placed in a format that includes fields such as the utterance to be parsed; the original utterance (complete with disfluencies); whether the utterance is incomplete or ungrammatical; speaker label; whether the utterance is part of a series of conjoined main clauses; and fields that will be automatically filled in by the parser for the syntactic and semantic analyses and the timestamp of the parse.

3.2. Automatic generation of linguistic analyses

We use a broad-coverage deep parser based on a bottom-up algorithm and an augmented context-free grammar with hierarchical features. The parser-generated syntactic representation consists of a derivation tree structure labeled with phrasal categories and terminal nodes, as shown in Figure 1.[1] The tree also includes word senses and their base forms in the TYPE feature. The word senses come from a domain-independent ontology developed together with the grammar. Our ontology for actions and states (expressed by verbs and nominalizations) has a top-level structure consistent with FrameNet (Johnson and Fillmore, 2000), though we use a smaller set of semantic roles to make syntactic linking easier (Dzikovska, 2004). The domain-independent ontology is combined with a domain model that adds selectional restrictions to the semantic role arguments, allowing the parser to choose the correct interpretation among available alternatives. Domain-independent semantic features are associated with each word to differentiate between e.g., at the top level, physical objects, abstract objects, situations and times, each of which has a set of basic semantic characteristics (see Figure 3 for examples of the :SEM feature vector).

The parser generates a semantic representation that is a flat unscoped logical form with events and labeled semantic arguments. An abbreviated form is shown in Figure 2; excerpts of the full terms from this form, which show the semantic feature representation, are in Figure 3. Each term includes an identifying variable (:VAR), indicating dependency relations between terms; the semantic class; semantic features (:SEM); the term type; and any modifications (:MODS). Some information is specific to certain kinds of terms, e.g., tense, modality and aspect (:TMA) is generated for terms obtained from verbs, and referential information (:CONTEXT-REL) for pronouns.The semantic representation also includes terms for null elements, such as implied subjects in imperatives, and speech act information in the form of a classification of the utterance as a statement, query, acknowledgment, rejection, etc.[2]

3.3. Checking and updating the corpus

The parser-generated analyses are automatically checked in a post-processing stage as a rough guide for annotators until the parses are checked by hand. The analyses are hand-checked by trained annotators using a software tool developed for this purpose, shown in Figure 2. Annotators use the tool to display the syntactic and semantic analyses [3] for each utterance and assign a status of GOOD or BAD. Comments on status assignments can be entered, using keywords if applicable, e.g., BAD-SENSE if an incorrect word sense appears in the analyses. Recording the reasons for status assignment speeds up subsequent cycles of annotation because known past problems can be quickly checked. Initially, checking took between 2 and 4 hours (depending on the annotator) for a 300-utterance

[1] Some leaf nodes are omitted due to space constraints.

[2] Details of the parser's semantic representation can be found at http://www.cs.rochester.edu/research/cisd/resources/parser/lf.

[3] Dependencies are highlighted with matching colors on the :VAR to help the checking process.

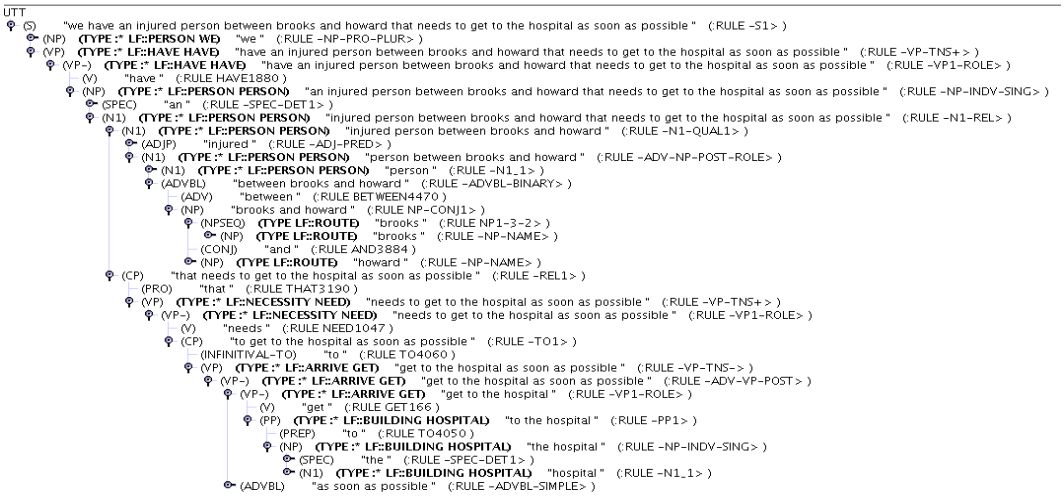

Figure 1: Sample parse tree for s16 utterance 297

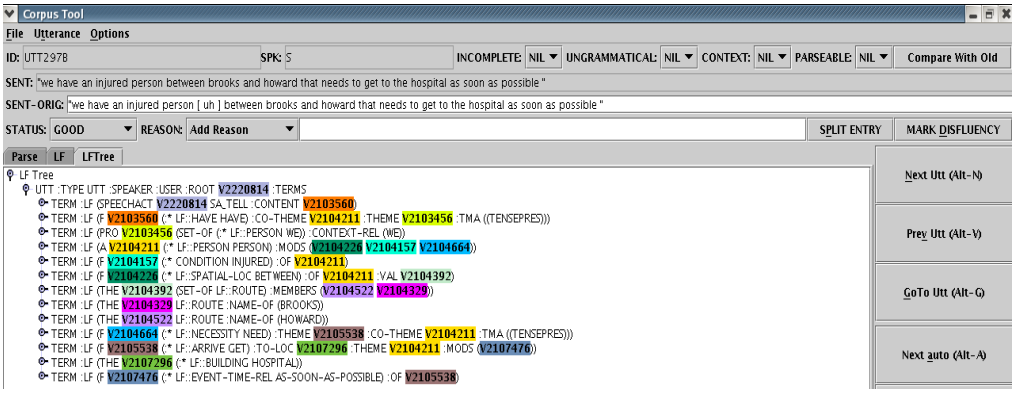

Figure 2: Top-level display of the logical form with the corpus tool

dialog. After a month of training, updated dialogs can be checked at a speed of about 200 utterances per hour.

If multiple analyses are possible for a given utterance due to lexical or syntactic ambiguity, and the top analysis selected by the parser at run-time is not the correct one, alternate parses can be generated and the correct one selected and saved in the gold-standard file. These analyses have the status HAND-GEN, indicating that they are not the parser's first choice and had to be generated by iteratively parsing the same utterance to obtain an alternate structure.

The hand-checked parses are used to identify problems with the parser, which are addressed through lexicon and grammar development, after which the dialogs are parsed again. New parse results are automatically merged with the old hand-checked parses. If the parse is unchanged, the new parse is kept, with the old hand-checked status. Otherwise, the new parses with their automatically assigned status replace the old ones, so they must be hand-checked. If the parse changes and the old status is GOOD, the old entry is retained for comparison. This iterative process is useful for lexicon and grammar development, since the amount of work spent on this phase diminishes as lexical and grammatical coverage increases.

4. Results

For each of the dialogs, Table 1 shows the number of parser-generated analyses that are good, hand-generated, or bad, together with the number of utterances that are either incomplete or ungrammatical (hence no parse attempted), total utterances, and finally the percent of correct analyses out of the utterances for which parses were generated.[4]

Our parser currently generates correct analyses for 85% of the utterances in the Monroe corpus and 70% in the MedAdvisor corpus. We continue to expand our coverage to handle difficult phenomena common in spoken language, such as elliptical responses e.g., *So two stretcher and three walking*, uttered in reference to accident victims.

5. Conclusions and future work

With this technique we will create annotated corpora for multiple domains and use them to train a statistical parser. We also plan to use the corpora to train the parser to automatically detect disfluencies and unparseable sentences to save time in manual annotation. Additionally, since our

[4]Medication domain analyses are from (Dzikovska, 2004).

```
(TERM :VAR V2105538
      :LF (F V2105538 (:* LF::ARRIVE GET) :TO-LOC V2107296 :THEME V2104211 :MODS (V2107476))
      :SEM ($ F::SITUATION (F::LOCATIVE F::ANY-LOCATIVE) (F::CAUSE -) (F::TIME-SPAN F::ATOMIC)
           (F::ASPECT F::BOUNDED) (F::INTENTIONAL -) (F::INFORMATION F::PROPOSITION)
           (F::CONTAINER -) (F::KR-TYPE KR::MOVE) (F::TRAJECTORY -) (F::TYPE F::EVENTUALITY)))
(TERM :VAR V2107296
      :LF (THE V2107296 (:* LF::BUILDING HOSPITAL))
      :SEM ($ F::PHYS-OBJ (F::OBJECT-FUNCTION F::PLACE) (F::ORIGIN F::ARTIFACT)
           (F::FORM F::GEOGRAPHICAL-OBJECT) (F::MOBILITY F::FIXED) (F::GROUP -)
           (F::SPATIAL-ABSTRACTION F::SPATIAL-POINT) (F::INTENTIONAL -) (F::INFORMATION -)
           (F::CONTAINER -) (F::KR-TYPE KR::HOSPITAL) (F::TRAJECTORY -) (F::TYPE F::ANY-TYPE)))
```

Figure 3: Excerpt from full logical form for s16 utterance 297

Dialog	s2	s4	s12	s16	s17	Total	med1	med2	med3	med4	med5	Total
Good	325	246	151	298	311	1331	47	23	30	67	32	199
Hand-Gen	9	2	0	0	1	12	0	0	0	0	0	0
Bad	34	78	17	56	54	239	12	16	18	20	19	85
Incomplete	35	51	19	21	23	149	1	0	0	3	4	8
Ungramm	2	10	2	8	3	25	0	0	0	0	2	2
Total Utts	405	387	189	383	392	1756	60	39	48	90	57	294
% Correct	90.8	76.1	89.9	84.2	85.3	84.9	79.7	58.9	62.5	77	61.5	70.1

Table 1: Statistics for Monroe and MedAdvisor dialogs

corpora contain original utterances as well as their counterparts cleaned of disfluencies, it can be used for testing error-correcting parsers such as (Core and Schubert, 1999). The rich semantics of the corpus will also facilitate reference resolution studies and research on implicit roles, which requires a corpus with labeled thematic roles.

Our method builds linguistically annotated corpora semi-automatically by generating syntactic, semantic and discourse information with a broad-coverage, deep parser. We use domain-specific semantic constraints to find the correct analysis, eliminating the need to select the best parse by hand from a set of alternatives. Incorrect analyses serve as input to lexical and grammatical development. In this way, a linguistically annotated corpus is generated as part of general system development, while our grammar and lexicon improve in coverage and robustness.

6. References

Barsotti, F., R. Basili, M. Battista, O. Corazzari N. Calzolari, R. del Monte, F. Fanciulli, N. Mana, M. Massetani, S. Montemagni, M. Pazienza, F. Pianesi, R. Raffaelli, D. Saracino, A. Zampolli, and F. Zanzotto, 2000. The Italian Syntactic-Semantic Treebank. In A. Abeille (ed.), *Building and using syntactically annoted corpora*. Kluwer.

Brants, T. and O. Plaehn, 2000. Interactive corpus annotation. In *Proceedings of LREC2000*.

Cahill, A., M. McCarthy, J. van Genabith, and A. Way, 2002. Automatic annotation of the Penn-Treebank with LFG F-structure information. In *Proceedings of LREC2002 workshop on Linguistic Knowledge Acquisition and Representation*.

Copestake, A., D. Flickinger, and I. A. Sag, 1997. Minimal Recursion Semantics: An introduction. Technical report, CSLI, Stanford University, CA.

Core, M. G. and L. K. Schubert, 1999. A model of speech repairs and other disruptions. In S. E. Brennan, A. Giboin, and D. Traum (eds.), *Working Papers of the AAAI Fall Symposium on Psychological Models of Communication in Collaborative Systems*. Menlo Park, CA.

Dzikovska, M. O., 2004. *A Practical Semantic Representation for Natural Language Parsing*. Ph.D. thesis, U. Rochester.

Dzikovska, M. O., M. D. Swift, and J. F. Allen, 2003. Constructing custom semantic representations from a generic lexicon. In *Proceedings of IWCS5*. Tilburg, Netherlands.

Ferguson, G., J. F. Allen, N. Blaylock, D. Byron, N. Chambers, M. Dzikovska, L. Galescu, X. Shen, R. Swier, and M. Swift, 2002. The Medication Advisor Project. Technical Report 766, U. Rochester, Computer Science Department.

Johnson, C. and C. J. Fillmore, 2000. The FrameNet tagset for frame-semantic and syntactic coding of predicate-argument structure. In *Proceedings ANLP-NAACL 2000*.

Marcus, M. P., G. Kim, M. A. Marcinkiewicz, R. MacIntyre, A. Bies, M. Ferguson, K. Katz, and B. Schasberger, 1994. The Penn Treebank: Annotating predicate argument structure. In *Proceedings of ARPA Human Language Technology Workshop*.

Marcus, M. P., B Santorini, and M. A. Marcinkiewicz, 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19:313–330.

Oepen, S., D. Flickinger, K. Toutanova, and C. D. Manning, 2002. LinGO Redwoods: A rich and dynamic treebank for HPSG. In *Proceedings of First Workshop on Treebanks and Linguistic Theories (TLT2002)*.

Stent, A J., 2001. *Dialogue Systems as Conversational Partners*. Ph.D. thesis, U. Rochester.