Exploiting Semantic Dependencies in Parsing

William Schuler
Computer and Information Science Dept.
University of Pennsylvania
Philadelphia, PA 19103
schuler@linc.cis.upenn.edu

Abstract

In this paper we describe a semantic dependency model for estimating probabilities in a stochastic TAG parser (Resnik, 1992) (Schabes, 1992), and we compare it with the syntactic dependency model inherent in a TAG derivation using the flat treatment of modifiers described in (Schabes and Shieber, 1994).

1 Introduction

The use of syntactic dependencies to estimate parser probabilities is not uncommon (Eisner, 1996) (Collins, 1997) (Charniak, 1997). Typically, a maximum probability parse is estimated from bigram statistics of lexical items that participate in head-modifier or head-complement dependencies with other lexical items. These dependencies can be characterized as ⟨ head, label, modifier ⟩ triples and ⟨ head, label, complement ⟩ triples – or as labeled directed arcs in a graph – which have the property that each lexical item may participate as a modifier or a complement in no more than one dependency. Using a TAG derivation tree (Joshi, 1987) with a flat treatment of modifiers (Schabes and Shieber, 1994), it is possible to capture the long distance dependencies of wh-extractions and relative clauses as adjacent arcs in a dependency structure, making them available for probability estimates within the parser as well. In this case, the head-complement dependencies for a sentence correspond to a set $\mathcal{S}$ of substitution triples $\langle \gamma, \eta, \alpha \rangle$ (where tree α substitutes into tree γ at note address η), and the head-modifier dependencies correspond to a set $\mathcal{A}$ of adjunction triples $\langle \gamma, \eta; \beta \rangle$ (where tree β adjoins into tree γ at node address η), in a probabilistic TAG (Resnik, 1992).[1]

Although the TAG-based syntactic dependency model has the necessary domain of locality (in terms of adjacent arcs on the derivation tree) to accurately guide a statistical parser, it is still susceptible to sparse data effects, in part because it does not generalize attachment statistics across syntactic transformations. An adjective used as a declarative predicate, for example, could not draw on attachment statistics for the same adjective used as a modifier, or as a predicate in a relative clause, and vice versa, because each transformation uses a different syntactic dependency structure. The triples in the syntactic dependency sets $\mathcal{S}$ and $\mathcal{A}$ for the sentences, "The damaged handle is attached to the drawer," and "The handle attached to the drawer is damaged," are represented as arcs in Figure 1.

In order to group these attachment statistics into denser pools of data, we need to abstract a common semantic structure from the various syntactic structures, effectively adopting a common argument frame for each transformation. This means that each auxiliary tree must have an argument position corresponding to the subject substitution site in its predicative transformation if it is a modifier auxiliary, or corresponding to the wh-object substitution site in its object-extraction transformation if it is a predicative auxiliary.[2] For convention, we place

[1] Although Resnik uses a direct function $S(\gamma, \eta, \alpha)$ to the $[0-1]$ interval where we use a probability of set membership $\mathcal{P}(\langle \gamma, \eta, \alpha \rangle \in \mathcal{S})$. Also note that this correspondence between head-complement dependencies and substitution dependencies is not strictly true in the case of predicative auxiliaries (Schabes and Shieber, 1994), which are handled by adjunction in TAG.

[2] See (Schabes and Shieber, 1994) for a description of the distinction between modifier and predicative auxiliaries.

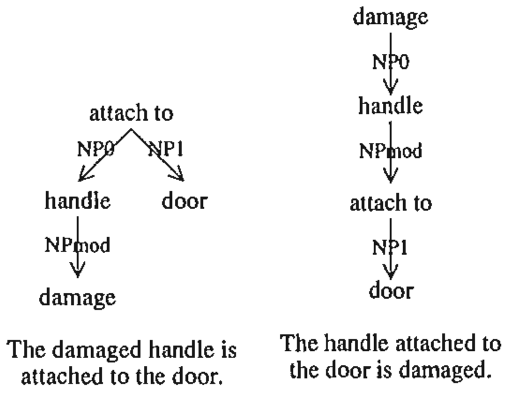

Figure 1: Syntactic dependencies in TAG

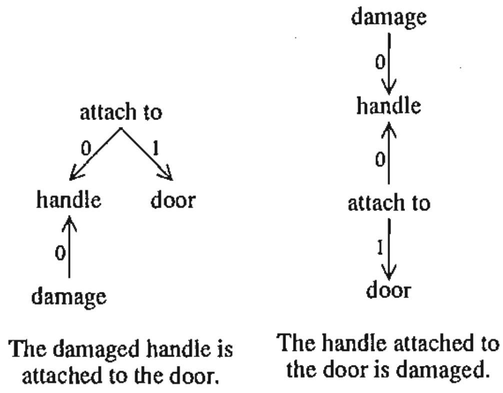

Figure 2: Semantic dependencies

this extra argument position at the foot node of the auxiliary tree, so the auxiliary takes the tree it adjoins into as an argument. This means that our semantic dependency model effectively reverses the direction of dependencies involved in adjunction from the syntactic model.The triples in the semantic dependency set $\mathcal{D}$ for the sentences, "The damaged handle is attached to the drawer," and "The handle attached to the drawer is damaged," are represented as arcs in Figure 2.

Formally, we augment the syntactic dependency sets $\mathcal{S}$ and $\mathcal{A}$ with a semantic dependency set $\mathcal{D}$ of $\langle$ predicate, label, argument $\rangle$ triples defined as follows:

- For every substitution (head-complement) dependency $\langle \gamma, \eta, \alpha \rangle$ in $\mathcal{S}$ add a predicate-argument dependency $\langle anchor(\gamma), argnum(\gamma, \eta), anchor(\alpha) \rangle$ to $\mathcal{D}$; and
- For every adjunction (head-modifier) dependency $\langle \gamma, \eta, \beta \rangle$ in $\mathcal{A}$ add a predicate-argument dependency $\langle anchor(\beta), argnum(\beta, foot(\beta)), anchor(\gamma) \rangle$ to $\mathcal{D}$;

where $anchor(\alpha)$ returns the lexical anchor of tree α, and $argnum(\alpha, \eta)$ returns the semantic argument position corresponding to node η in tree α. In this way we can combine argument attachment distributions for initial tree transformations and auxiliary tree transformations into a common attachment distribution for the underlying predicate.

2 Parsing

Parsing proceeds in three passes of $O(n^6)$ complexity. First, the chart is filled in from the bottom up, as described in (Schabes et al., 1988), and the input is recognized or rejected. The parser then constructs a shared forest (Vijay-Shanker and Weir, 1993) top-down from the elements in the chart, ignoring those items on bottom-up dead ends. Finally, the parser proceeds with the more expensive operations of feature unification and probability estimation on the reduced set of nodes in the shared forest. The chart consists of a set of items that each specify a node address η in an elementary tree α, a top ($\top$) or bottom ($\perp$) marker denoting the phase of operation on the node, and four indices i,j,k, and l, composing the extent of the node's coverage in the sentence: $\langle \alpha, \eta, \top, i, j, k, l \rangle$. The shared forest consists of an and/or graph, with 'or' arcs from each non-dead-end chart item to instantiations of the parser productions that could have produced it, and 'and' arcs from each instantiation of a parser production to the chart items it would have required.

In order to select a most-preferred parse for an ambiguous input, a highest-probability item is selected from the top node in the shared forest, and a parse is read off below it by traversing the subordinate items with the most probable dependencies. The probability of each shared forest item is computed as the maximum of the probabilities of its 'or'-adjacent parser productions. The probability of each instantiation of a parser production is computed as the probability of the relevant dependency for that production multiplied by the probabilities of the chart items that production required. Finally, the probability of each parse must be multiplied by the probability of each elementary tree given a lexical item in the input.

The probability model is adapted from (Resnik, 1992), which assigns a probability to any arc $\langle \alpha, \eta, \beta \rangle$ (where tree β is attached to tree α at node address η) being in the set of substitutions $\mathcal{S}$ or adjunctions $\mathcal{A}$ in a derivation. The root of the derivation tree is represented as $\langle MAIN, 0, \alpha \rangle$ in $\mathcal{S}$, and null adjunctions (which terminate the adjunction of modifiers at a node) are represented as $\langle \alpha, \eta, \epsilon \rangle$ in $\mathcal{A}$. Finally, the probability of a tree α is represented as the probability of the double $\langle anchor(\alpha), tree(\alpha) \rangle$ being in the set $\mathcal{T}$ of elementary trees used in a parse.

Probabilities for the dependencies in a parser production are estimated from observed frequencies that a child predicate c (the base-form anchor of a tree) occurs in argument position a of a parent predicate p (the base-form anchor of another tree), within some training set $\mathcal{D}$ of dependency structures: $\hat{F}(\langle a, p, c \rangle \in \mathcal{D})$. The top-level dependency is represented in $\mathcal{D}$ as $\langle MAIN, 0, c \rangle$, and null adjunctions are represented as $\langle NULL, 0, c \rangle$.[3] Note that we use the same dependencies as Resnik (the syntactic dependency sets $\mathcal{S}$ and $\mathcal{A}$) in describing the probability model, and use the semantic dependencies ($\mathcal{D}$) only in the estimation of those probabilities.

Probabilities are estimated as follows:

- For any topmost item in a derivation tree: $\langle \alpha, 0, \top, 0, -, -, n \rangle$
  the initial probability would be:
  $\mathcal{P}(\langle MAIN, 0, \alpha \rangle \in \mathcal{S} \mid \langle MAIN, 0, \_ \rangle \in \mathcal{S})$
  which we estimate as:
  $\frac{\hat{F}(\langle MAIN, 0, anchor(\alpha) \rangle \in \mathcal{D})}{\hat{F}(\langle MAIN, 0, \_ \rangle \in \mathcal{D})}$
- For any chart production for the substitution of initial tree α into γ at node address η, where i and j are indices, and η is a substitution site in γ with the same label as the root of α:
  $\frac{\langle \alpha, 0, \top, i, -, -, j \rangle}{\langle \gamma, \eta, \top, i, -, -, j \rangle}$
  the probability would be:
  $\mathcal{P}(\langle \gamma, \eta, \alpha \rangle \in \mathcal{S} \mid \langle \gamma, \eta, \_ \rangle \in \mathcal{S})$
  which we estimate as:
  $\frac{\hat{F}(\langle anchor(\gamma), argnum(\gamma,\eta), anchor(\alpha) \rangle \in \mathcal{D})}{\hat{F}(\langle anchor(\gamma), argnum(\gamma,\eta), \_ \rangle \in \mathcal{D})}$
- For any chart production for adjunction of auxiliary tree β into γ at node address η, where i, j, i', j', p and q are indices, and η is an adjunction site in γ with the same label as the root of β:
  $\frac{\langle \gamma, \eta, \bot, i', p, q, j' \rangle \quad \langle \beta, 0, \top, i, i', j', j \rangle}{\langle \gamma, \eta, \bot, i, p, q, j \rangle}$
  the probability would be:
  $\mathcal{P}(\langle \gamma, \eta, \beta \rangle \in \mathcal{A} \mid \langle \gamma, \eta, \_ \rangle \in \mathcal{A})$
  which we estimate as:
  $\frac{\hat{F}(\langle anchor(\beta), argnum(\beta, foot(\beta)), anchor(\gamma) \rangle \in \mathcal{D})}{\hat{F}(\langle \_, \_, anchor(\gamma) \rangle \in \mathcal{D})}$
- For any chart production for closing adjunction at a node address η in tree γ:
  $\frac{\langle \gamma, \eta, \top, i, j, k, l \rangle}{\langle \gamma, \eta, \bot, i, j, k, l \rangle}$
  the probability would be:
  $\mathcal{P}(\langle \gamma, \eta, \epsilon \rangle \in \mathcal{A} \mid \langle \gamma, \eta, \_ \rangle \in \mathcal{A})$
  which we estimate as:
  $\frac{\hat{F}(\langle NULL, 0, anchor(\gamma) \rangle \in \mathcal{D})}{\hat{F}(\langle \_, \_, anchor(\gamma) \rangle \in \mathcal{D})}$
- For any other chart production, the probability would be 1.
- Finally, the probability that each elementary tree α is in the set of trees $\mathcal{T}$ used in the parse, given a lexical item is:
  $\mathcal{P}(\langle anchor(\alpha), tree(\alpha) \rangle \in \mathcal{T} \mid \langle anchor(\alpha), \_ \rangle \in \mathcal{T})$
  which we estimate as:
  $\frac{\hat{F}(\langle anchor(\alpha), tree(\alpha) \rangle \in \mathcal{T})}{\hat{F}(\langle anchor(\alpha), \_ \rangle \in \mathcal{T})}$

3 Practical Issues

The extended goal of this project was to provide a natural language interface for "Jack" (Badler et al., 1993), a human-like agent that answers questions and carries out instructions in a virtual 3-D environment. The system's restricted domain makes unknown words and unknown syntactic structures unlikely, and the goal of translating inputs into a formal language for the agent avoids the danger of modifier scoping

[3] Although since the null-adjunction probability only conditions on the parent tree, it will be a constant in every case, and can be ignored in estimating the maximum probability.

ambiguity (which our model does not evaluate), since the scoping of modifier adjuncts can usually be ignored in transfer. It is for this reason that we concentrate our attention on parsing attachment ambiguity at the expense of other problems which might seem more relevant in free text applications.

We consider our approach orthogonal to statistical smoothing techniques such as (Charniak, 1997) for addressing the sparse data problem, and for this reason do not discuss them.

References

Badler, N., Phillips, C., and Webber, B. (1993). *Simulating humans: Computer graphics animation and control.* Oxford University Press, New York, NY.

Charniak, E. (1997). Statistical parsing with a context-free grammar and word statistics. In *Fourteenth National Conference on Artificial Intelligence*, Providence, Rhode Island.

Collins, M. (1997). Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL '97).*

Eisner, J. (1996). Three new probabilistic models for dependency grammar. In *Proceedings of the Sixteenth International Conference on Computational Linguistics (COLING '96).*

Joshi, A. K. (1987). An introduction to tree adjoining grammars. In Manaster-Ramer, A., editor, *Mathematics of Language.* John Benjamins, Amsterdam.

Resnik, P. (1992). Probabilistic tree-adjoining grammar as a framework for statistical natural language processing. In *Proceedings of the Fourteenth International Conference on Computational Linguistics (COLING '92)*, Nantes, France.

Schabes, Y. (1992). Stochastic lexicalized tree-adjoining grammars. In *Proceedings of the Fourteenth International Conference on Computational Linguistics (COLING '92)*, Nantes, France.

Schabes, Y., Abeillé, A., and Joshi, A. K. (1988). Parsing strategies with lexicalized grammars: Applications to tree adjoining grammars. In *Proceedings of the 12th International Conference on Computational Linguistics (COLING '88)*, Budapest, Hungary.

Schabes, Y. and Shieber, S. M. (1994). An alternative conception of tree-adjoining derivation. *Computational Linguistics*, 20(1):91–124.

Vijay-Shanker, K. aud Weir, D. (1993). The use of shared forests in tree adjoining grammar parsing. In *Proceedings of EACL '93*, pages 384–393.