

Language Understanding as Recognition and Transduction of Numerous Overlaid Patterns

Lenhart Schubert

Department of Computer Science, University of Rochester

Introduction

Few computational linguists would dispute that language understanding crucially involves pattern recognition, in some sense, and transduction to an internal representation. However, theories differ on the sorts of patterns and processes they posit as underlying this recognition and transduction process.

I would characterize the traditional view, dominant for several decades (and decidedly still part of the common coin), in terms of the following central ingredients:

“concise syntax + concise interpretive rules + parsing + choices based on preferences and context”;

and more recent statistically inspired approaches in terms of “larger syntax + distributional properties + statistical parsing (+ concise interpretive rules, if any)”.

I want to suggest here that it is time to extrapolate beyond the traditional and statistical views, elaborating basic phrase (or dependency) structure and statistically weighted features into multiple kinds of overlaid syntactic, semantic, epistemic, and discourse patterns. These patterns jointly determine a (more or less) unambiguous consensus structure for the utterances that are processed, and a (more or less) unambiguous interpretation. Implementing such an approach would set us on a course to human-like learning by reading.

Augmenting basic phrase structure with overlaid patterns

Traditional syntax entails that we seek maximum coverage with a minimal rule set. In a computational setting, given a sentence admitting multiple structural analyses according to those rules, we then attempt to select a unique constituent structure as the one that the author had in mind and intends the reader to recognize.¹ This selection is assumed to involve various syntactic preferences and preferences of verbs, nouns and adjectives for particular semantic types of arguments, as well as pragmatic, contextual factors such as the match-up of referring expressions with contextually salient entities, and the compatibility of resulting sentence

Copyright © 2009, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

¹Of course this applies to spoken language as well, but in the context of “learning by reading”, we set aside speech recognition issues.

meanings with world knowledge and contextual presuppositions. Statistical NLP allows for frequency information about phrase structure, and in addition tries to capture some types of syntactic and semantic preferences through local features of a text (such as local lexical phrase heads, function words, part-of-speech patterns, syntactic dependency paths, or the presence of words falling under certain high-level hypernyms or into certain learned clusters), and embeds these features in a probabilistic language model.

Yet statistical parsers still misanalyze most longer sentences. (A phrasal accuracy of 90% or so still entails at least one error per sentence for average-length sentences, e.g., in the WSJ or Brown corpora.) What is striking in the misanalyzed sentences is that even allowing for the lack of context, the parser seems to have missed the obvious: it has combined constituents in an odd way, when the habituated, “natural” combination should have been obvious. Similarly errors in statistical word sense disambiguation run against our intuitions about the most natural, most familiar configurations of meanings in language.

My hypothesis is that statistical (and of course classical) parsers fall far short in their ability to take into account multiple, overlaid patterns, and only an approach that is responsive to these patterns (encoding them in some explicit or implicit way) can hope to match human performance in sentence analysis and interpretation. For example, consider the sentence

“Throwing money at the problem won’t solve it”.

This instantiates very general phrase structure patterns such as the following (selectively annotated with examples from the above sentence and a few others):

S → NP VP
NP[gerund] → VP[-ing]
VP → V[_np_pp:goal] NP PP[goal]
 “throwing money at the problem”
 “took the dog into the car”
VP → V[_np] NP
 “solve it”
 “eat the pizza”
VP → V[aux,_vp:base] ADV⁻ VP[base]
 “won’t solve it”
 “does occasionally exercise”

But it also instantiates other familiar, more specific patterns

that we might write in an *ad hoc* way as

- VP → V[_np_pp:at] NP PP[goal]
“point/aim/throw/hurl/launch/swing an object
at something”
- VP → V[throw,_np:hand-loftable-obj_pp:at]
NP[hand-loftable-obj] PP[at+target]
“throw/toss/hurl an object at something”
- VP → V[throw,_np:money_pp:at] NP[money]
PP[at+problem]
Google shows 89,500 hits for “throwing money
at the/a problem”
- VP → V[solve,_np:problem] NP[problem]
- VP → V[aux,_vp:base,-n’t] VP[base]
“won’t/can’t/doesn’t ... solve it”

I believe we recognize all of these patterns (and probably more), even if only unconsciously; i.e., we recognize “throwing money at the problem” as instantiating the general V[_np_pp:goal] NP PP[goal] pattern, we further recognize the instantiation of the idea of throwing a loftable object at a target, and (most saliently) we recognize the particular idea of throwing money at a problem, as a stock metaphor for wanton expenditures on the problem. Similarly we recognize the VP “solve it” as an instance of a broad V[_np] NP pattern, but we also recognize “solving a problem” as a strongly habituated specific version. In fact, it is partly because of the familiarity of this pattern that the choice of “the problem” (rather than “money”) as referent of “it” is so compelling here.

Note that these patterns blur the boundary between syntax and semantics. The three types of patterns above spanning the phrase “throwing money at a problem” have been rendered as syntactic rules, but the *ad hoc* features employed in them can probably be viewed more naturally as semantic rather than syntactic features. We could in fact cast at least the two more specific patterns instantiated by the phrase as patterns of predication something like

- throw-toward-goal(agent, loftable-obj, entity),
throw-at(agent, money, problem).

But certainly stock metaphors and idioms are typically constrained syntactically as well. In the present example, passivization and modification of the stock metaphor are possible (“Money was thrown liberally at the problem”), but other examples are more resistant to such variation, as in the case of “turning over a new leaf” or “kicking the bucket”.

In addition the sentence as a whole instantiates the familiar idea that providing ample funds for solving a problem may be ineffectual. This can be viewed as a pattern at the *epistemic* (knowledge) level. Patterns are also important at the discourse level, as seen for example in the formulaic nature of a greeting exchange, in question/answer structure, and more subtly in narrative and rhetorical structure such as general observance of chronological order or, in descriptive passages, moving from an encompassing perspective to a more detailed one. These patterns are key to recovery of the temporal, causal, and part structure underlying the explicitly mentioned events and entities.

The challenge, and an approach

Our recognition of these multiple, overlaid, often very specialized patterns secures and particularizes our understanding of the text segments they span. If our computational mechanisms are to achieve human-level precision and comprehension, they will have to match our capacity for this type of integrated pattern recognition. Since the patterns we have been discussing are generally aligned with basic phrase structure, a natural way of allowing for them is to have them contribute various “weights” to hypothesized constituents, adding these up in a bottom-up sweep, much as in log-linear probabilistic approaches.

The accumulation of the requisite patterns is a formidable challenge. In our example we started with 5 general phrasal patterns, and added 5 more specialized ones. Such specialized patterns typically involve two or more relatively narrow word meaning categories or even particular word senses. Since there are many thousands of verbs, and their complements can involve many thousands of open-class head words (and semantic categories of such words), the potential number of VP patterns is enormous. Similar considerations apply to the complement patterns of other types of phrases. Moreover, we can understand modification patterns such as N N, A N, N PP, and VP PP combinations in some very generic ways, but we also are attuned to myriad specific combinations. Thus the requisite pattern base for human-like parsing may well be several orders of magnitude larger than grammars used in current parsers (which may contain a few thousand or a few tens of thousands of syntactic rules).

At the University of Rochester, we have been working for a number of years on a system called KNEXT (KNOWledge EXtraction from Text) with two goals in mind: populating a knowledge base for a commonsense reasoner, and providing patterns of predication to guide a parser (see the listed references by Schubert and collaborators). KNEXT applies compositional semantic rules and some simple semantic abstraction methods to statistically parsed text, and in this way derives logically represented general “factoids” (automatically rendered back into English), such as that “A PERSON MAY BELIEVE A PROPOSITION”, “A COMEDY CAN BE DELIGHTFUL”, or “PEOPLE MAY WANT TO BE RID OF A DICTATOR”. More than 10 million such factoids (of variable quality) have been extracted from the British National Corpus and web documents.

Concerning parser guidance, our impression from scrutinizing these factoids is that their quantity and quality is at a useful level, but that we will require still more patterns, and will need to disambiguate, merge and adjust many of them to yield patterns at the “right” level of abstraction. This is a focus of our current work. The usefulness of our pursuit is bolstered by a recent study at Boeing Company (Clark & Harrison 2008) showing that “tuples” similar to our factoids could be used to improve parsing accuracy.

As a final remark, we expect the kinds of overlaid patterns we have discussed to constrain not only parser choices, but also word sense (or “phrase sense”) disambiguation. For an approach to this issue based on factoid abstraction via WordNet, see (Van Durme, Michalak, and Schubert 2008).

References

- Clark, P., and Harrison, P. 2008. Large-scale extraction and use of knowledge from text. Submitted.
- Schubert, L.K. 2002. Can we derive general world knowledge from texts? In *Proc. of the 2nd Int. Conf. on Human Language Technology Research (HLT 2002)*, March 24-27, San Diego, CA, 94-97.
- Schubert, L.K., and Tong, M. 2003. Extracting and evaluating general world knowledge from the Brown Corpus. In *Proc. of the HLT-NAACL Workshop on Text Meaning*, May 31, Edmonton, Alberta, 7-13.
- Schubert, L.K. 2006. Turing's dream and the knowledge challenge. In *Proc. of the 21st Nat. Conf. on Artificial Intelligence (AAAI'2006)*, July 16-20, Boston, MA, AAAI Press, Menlo Park, CA, 1534-8.
- Schubert, L.K., and Van Durme, B. 2008. Open extraction of general knowledge through compositional semantics. *Notebook of the NSF Symposium on Semantic Knowledge Discovery, Organization and Use*, New York University, Nov. 14-15.
- Van Durme, B., Michalak, P., and Schubert, L.K. 2009. Deriving generalized knowledge from corpora using WordNet abstraction. *12th Conf. of the European Chapter of the Assoc. for Computational Linguistics (EACL-09)*, Mar. 30 - Apr. 3, Athens, Greece.
- Van Durme, B., and Schubert, L.K. 2008. Open Knowledge Extraction through Compositional Language Processing. *Symposium on Semantics in Systems for Text Processing (STEP 2008)*, September 22-24, Venice, Italy.
- Van Durme, B.; Qian, T; and Schubert, L.K. 2008. Class-driven attribute extraction. In *Proc. of COLING'08*, Aug. 18-22, Manchester, UK.