Data-driven learning in an incremental grammar framework

Matthew Purver,[1] *Arash Eshghi*[2] *& Julian Hough*[1]
[1]School of Electronic Engineering and Computer Science, Queen Mary University of London
[2]School of Mathematical and Computer Sciences, Heriot-Watt University

Overview Incremental processing of both syntax and semantics, both in parsing and generation, is of significant interest for modelling the human language capability, and for building systems which interact with it. Formal linguistics has made significant contributions to this; one example is the framework Dynamic Syntax, which provides an inherently word-by-word incremental grammatical framework. However, making this practical for computational models or systems involves building grammars with broad coverage on real data – a significant challenge. Here, we describe a method for inducing such a grammar from a corpus in which sentences are paired with semantic logical forms. By taking a probabilistic view, we hypothesise possible lexical entries – including entries for anaphoric elements – and learn a lexicon from their observed distributions without requiring annotation at the word level. The resulting grammar provides a resource for incremental semantic processing with good coverage, while learning grammatical constraints similar to a hand-crafted version.

Background Many dialogue phenomena demonstrate the *incremental* nature of human language processing in interaction; in particular, our ability to produce and understand *compound contributions* – units in which a possibly incomplete contribution by one interlocutor can be queried, continued or completed by another. These impose some strict requirements on the level of incrementality required:

(1) A: Before that then if they were ill B: they got nothing.

(2) A: They took an x-ray. The doctor B: Chorlton? A: Uh-huh – examined me thoroughly.

(3) A: I'm afraid I haven't read B: any of my papers at all.

(4) A: I smell burning. Have you B: burned myself? No.

As (1)-(4) show, speaker changes can happen apparently at any point within a sentence; semantic processing must proceed incrementally, with partial interpretations being available for comprehension and clarification; and syntactic processing must be incremental, with dependencies (e.g. negative polarity items, reflexives) respected across speaker changes even though these changes do not respect constituent boundaries. Several grammatical frameworks might fulfil these conditions (including categorial grammars, tree-adjoining grammars and minimalist grammars) given a suitable approach to incremental parsing (see e.g. Demberg and Keller, 2008; Stabler, 2013). However, as (1)-(4) also show, these incremental processes must be able to switch seamlessly at any point between parsing and generation, as interlocutors switch from speaker to hearer and vice versa. Parser and generator states must be entirely compatible at all stages, with intermediate semantic and syntactic representations interchangeable. While this could of course be stipulated of suitable incremental parsing and generation frameworks, more explanatory power would be provided by a grammatical framework which itself ensures this must be the case.

Dynamic Syntax (DS) is one such framework: an inherently incremental semantic grammar formalism (Kempson et al., 2001; Cann et al., 2005) in which semantic representations are projected on a word-by-word basis. It recognises no intermediate layer of syntax (see below), but instead reflects grammatical constraints via constraints on the incremental construction of partial logical forms (LFs). Parsing and generation are defined in terms of these same incremental construction processes; it is therefore in principle capable of modelling and providing semantic interpretations for phenomena such as (1)-(4), not licensed directly by standard grammar formalisms but important for dialogue systems.

The output for any given string of words is a purely *semantic* tree representing its predicate-argument structure; tree nodes correspond to terms in the lambda calculus, decorated with labels expressing their semantic type and formula, with beta-reduction determining the type and formula at a mother node from those at its daughters (Figure 1). These trees can be *partial*, containing unsatisfied requirements for node labels (e.g. $?Ty(e)$ is a requirement for future development to $Ty(e)$), and a *pointer*

♢ on the node currently under development. The grammar is defined in terms of *actions*, both *lexical actions* associated with words and generally available *computational actions*, which specify monotonic tree updates (Figure 1). Grammaticality is defined as parsability: the successful incremental construction of a tree with no outstanding requirements (a *complete* tree) using all information given by the words in a sentence.

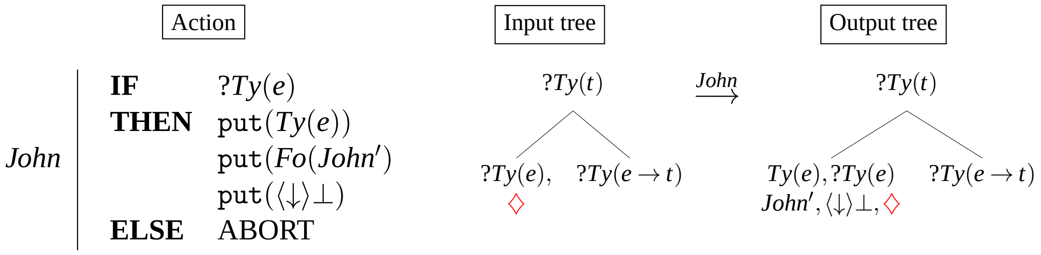

Figure 1: Lexical action for the word ‘John’

Induction However, its definition in terms of semantics (rather than the more familiar syntactic phrase structure) makes it hard to define or extend broad-coverage grammars: expert linguists are required. It is also not directly suitable for existing syntactic approaches to grammar learning and induction, which either learn grammars from treebanks annotated with syntactic trees at the word and phrase level (e.g. Charniak, 1996) or induce such grammars from lexical co-occurrence (e.g. Klein and Manning, 2005). However, recent research in has shown that lexicalised grammars can be learned using lightly supervised learning, guided by semantic annotation using sentence-level propositional logical form rather than detailed word-level annotation (e.g. Kwiatkowski et al., 2010). Here we take a similar approach, but apply it within DS’s strictly incremental, semantic formalism. The grammar learned is therefore inherently incremental, and can successfully treat items such as pronouns whose grammatical constraints depend on semantic context.

Approach We assume the availability of a LF in the form of a complete *target tree* for the sentence, containing information about its predicate-argument structure and how this is composed, but not about the relation of this to words or word order – see Figure 2, right. By hypothesising possible monotonic extensions of any partial tree which subsume this target tree, together with possible constraints on their application, we build a graph containing all possible parse paths for any sentence which lead to the target tree. A general anaphoric action can be hypothesised at any point which copies semantic information from some existing relative position on the current tree.

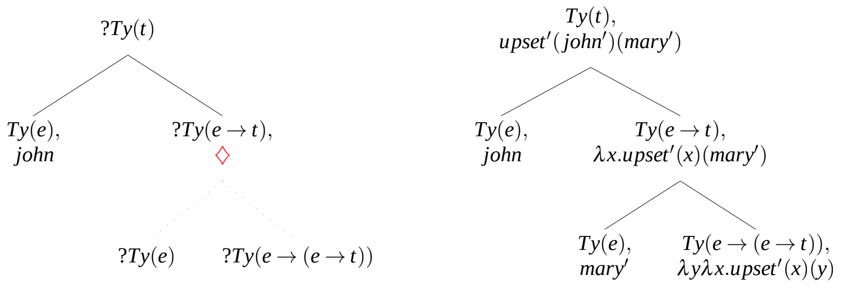
?Ty(t)
Ty(e), john
?Ty(e → t), ♢
?Ty(e)
?Ty(e → (e → t))
Ty(t), upset′(john′)(mary′)
Ty(e), john
Ty(e → t), λx.upset′(x)(mary′)
Ty(e), mary′
Ty(e → (e → t)), λyλx.upset′(x)(y)

Figure 2: Hypothesizing extension of tree under development T_{cur} (left) to target tree T_t (right)

Given these hypothesis graphs, we can now estimate a probability distribution θ_w over hypotheses for each word w, where $\theta_w(h)$ is the posterior probability $p(h|w)$ of a given word hypothesis h being used to parse w. For this we use an incremental version of the Expectation-Maximisation algorithm (Dempster et al., 1977): in the Expectation step we populate the hypothesis graph with the current estimates of these probabilities (assuming a uniform distribution over a held-out probability mass for as yet unseen hypotheses); the Maximisation step then re-estimates individual lexical hypothesis probabilities based on the probabilities of graph paths containing them. The most probable hypotheses then form the induced grammar, providing lexical actions with associated constraints (including, for words likely to be associated with an anaphoric copying action, constraints learnt on relative antecedent position).

We test this approach in terms of its ability to learn a grammar compatible with a known, manually defined grammar: we generate a semantically annotated corpus from this known grammar, induce a grammar from this corpus, and compare with the original. The corpus was generated by choosing words randomly from an existing defined lexicon, with probabilities following the broad part-of-speech type and token frequency distributions observed in the maternal utterance data from the CHILDES corpus (MacWhinney, 2000). The manual lexicon contained 156 lexical entries, and the resulting corpus contains 200 sentences with average length 3.7 words, paired with their semantic trees. 90% of these sentences were then used to induce a new grammar, and the remaining 10% used to evaluate its accuracy. Semantic accuracy on this test set, assessed as the availability of the correct LF within the top 2 derivations for successful parses, reaches 80%. Lexical ambiguity posed a challenge: 10% of original lexical entries were ambiguous between 2 or 3 different syntactic categories; in the induced grammar, only 57% of these words had entries with both senses in the top 3 hypotheses. Induction of grammatical constraints on context-dependent elements was tested by including relative pronouns: the induced lexical entries exactly match manually defined versions.

Acknowledgements We would like to thank Ruth Kempson, Yo Sato and the anonymous reviewers for helpful comments and discussion. This work was supported by the RISER project (EPSRC ref EP/J010383/1), and in part by the SpaceBook project (EU FP7 grant 270019).

Cann, R., Kempson, R., and Marten, L. (2005). *The Dynamics of Language*. Elsevier, Oxford.

Charniak, E. (1996). *Statistical Language Learning*. MIT Press.

Demberg, V. and Keller, F. (2008). A psycholinguistically motivated version of TAG. In *Proceedings of the International Workshop on Tree Adjoining Grammars*.

Dempster, A., Laird, N., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

Kempson, R., Meyer-Viol, W., and Gabbay, D. (2001). *Dynamic Syntax: The Flow of Language Understanding*. Blackwell.

Klein, D. and Manning, C. D. (2005). Natural language grammar induction with a generative constituent-context mode. *Pattern Recognition*, 38(9):1407–1419.

Kwiatkowski, T., Zettlemoyer, L., Goldwater, S., and Steedman, M. (2010). Inducing probabilistic CCG grammars from logical form with higher-order unification. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1223–1233, Cambridge, MA. Association for Computational Linguistics.

MacWhinney, B. (2000). *The CHILDES Project: Tools for Analyzing Talk*. Lawrence Erlbaum Associates, Mahwah, New Jersey, third edition.

Stabler, E. (2013). Two models of minimalist, incremental syntactic analysis. *Topics in Cognitive Science*, to appear. Available at http://www.linguistics.ucla.edu/people/stabler/Stabler12-2models.pdf.