Plans and the (Predicate Argument) Structure of Behavior

Mark Steedman

19th April 2015
Outline

• Introduction

• I: Planning

• II: From Planning to Semantics

• III: The Problem of Content

• IV: Hanging Language onto (Human) Planning

• Conclusions
Introduction

• There is a long tradition associating language and other serial cognitive behavior with an underlying motor planning mechanism (Piaget 1936, Lashley 1951, Miller et al. 1960).

• The evidence is evolutionary, neurophysiological, and developmental.

• It raises the possibility that language is much more closely related to embodied cognition than current linguistic theories of grammar suggest.
Introduction

- I’m going to argue that practically every aspect of language reflects this connection transparently, and that both cognitive and linguistic theories should be adjusted accordingly.

- The talk discusses this connection in terms of planning as it is viewed in Robotics and AI, with some attention to applicable machine learning techniques (Steedman 2002a,b).

- Work In Progress under ERC Advanced Fellowship 249520 GRAMPLUS and EU grant Xperience
Introduction

• The paper will sketch a path between representations at the level of the grounded sensory manifold and perceptron learning to the mid-level of plans and explanation-based learning, and on up to the level of language grammar and parsing model learning.

• At the levels of planning and linguistic representation, two simple but very general combinatory rule types, Composition (the operator $B$) and Type-Raising (the operator $T$) will appear repeatedly.

\[
Bfg \equiv \lambda x.f(gx) \quad T_a \equiv \lambda f.fa
\]

• Human planning requires an additional element, in the form of plan variables, which also provides the basis for distinctively human language.
I: Plans and the Structure of Behavior

- Apes really can solve the “monkeys and bananas” problem, using tools like old crates to gain altitude in order to reach objects out of reach.
Figure 1: Köhler 1925
Figure 2: Köhler 1925
What does it Take to Plan?

• Such planning involves
  – Retrieving *appropriate actions* from memory (such as piling boxes on top of one another, and climbing on them),
  – *Sequencing them* in a way that has a reasonable chance of bringing about a desired state or goal (such as having the bananas).

It is qualitatively different from Skinnerian shaping of purely reactive behavior in animals like pigeons—cf. http://www.youtube.com/watch?v=mDntbGReE
What does it Take to Plan?

• Köhler showed that, in apes at least, such search seems to be
  – object-oriented—that is, reactive to the presence of the tool, and
  – forward-chaining, working forward breadth-first from the tool to the goal,
    rather than backward-chaining (working from goal to tool).

• The first observation implies that actions are accessed via perception of the
  objects that mediate them—in other words that actions are represented in
  memory associatively, as properties of objects—in Gibson’s 1966 terms, as
  affordances of objects.

• The second observation suggests that in a cruel and nondeterministic world it
  is better to identify reasonably highly valued states that you have a reasonable
  chance of getting to than to optimize complete plans.
What does it Take to Plan?

- The problem of planning can therefore be viewed as the problem of Search for a sequence of actions or affordances in a “Kripke model”:
- A Kripke model is a tree or more accurately a lattice, in which nodes are states, and arcs are actions.
- A plan is then a sequence of actions that culminates in a state that satisfies the goal of the plan.

Search for plans is intrinsically recursive, and requires a Push-Down Automaton (PDA) to keep track of alternative paths to some limited depth.
- It is interesting that a PDA is also necessary to process recursive languages.
- But a PDA clearly isn’t enough for human language, which animals lack.
Representing Actions

• We can think of actions as STRIPS operators or as finite-state transducers (FSTs) over (sparse) state-space vectors

• FSTs are closed under composition, and can be represented as simple neural computational devices such as Perceptrons, or the Associative Network or Willshaw Net (Willshaw 1981 cf. Marr 1969).

⚠️ We still need a stack memory to run the search for plans.
Reducing Complexity

• We need the Kernel Generalization of Perceptrons to learn STRIPS rules (and their more modern descendants) as FSTs (Mourão et al. 2009, 2010).

• This calls for a highly structured state representation (Hume, 1738; Kant, 1781, passim), of a kind that can only be developed by more than 500M years of chordate evolution, using resources on a scale that is completely beyond machine learning.

• Like everyone else, we have to define a state-description language by hand.

• Complexity is $O(n^2)$, so we still need to keep the state vector small.

• We do this via a via a “deictic” or location-based attention mechanism cf. Agre and Chapman (1987) and Pasula et al. (2007)
Mourão 2012: Predicting STRIPS Update

10-fold cross-validation results

- Learnt Model
- Hand-coded Model
- Extended Model

Number of training examples

F-score
II: From Planning to Semantics

- How do we get from seriation and affordance (which we share with other animals) to language (which is uniquely human)?

- Seriation of actions to form a plan is Composition of FSTs or functions of type $\text{state} \rightarrow \text{state}$

- The Affordance of a state is a function from all those actions that are possible in that state into their respective result states.

- States are defined by the objects they include, so this is like exchanging objects for Type-Raised functions that map states into other states resulting from actions on those objects.
Actions as Functions

• Thus, the affordance of a (state including a) box to an ape is a function from actions like the box falling, their climbing-on the box and their putting the box on another box into resulting states whose utility the ape can evaluate.

• The functions are of the following (Curried) types, where $e$ is the type of a state satisfying preconditions including the presence of an entity, and $t$ is a consequent state:

  – $\text{fall}_{e \rightarrow t}$,
  – $\text{climb-on}_{e \rightarrow (e \rightarrow t)}$
  – $\text{put-on}_{e \rightarrow (e \rightarrow (e \rightarrow t))}$
Objects as Affordances

• Thus the ape’s concept of a box is an object-oriented set of Type-Raised functions of type

  \[ \text{box1} : (e \rightarrow t) \rightarrow t \]
  \[ \text{box2} : (e \rightarrow (e \rightarrow t)) \rightarrow (e \rightarrow t) \]
  \[ \text{box3} : (e \rightarrow (e \rightarrow (e \rightarrow t))) \rightarrow (e \rightarrow (e \rightarrow t)) \]

• —that is, functions from the current situation to the results of the actions it affords.

• Planning is then object-oriented seriation of affordances

• So the only place for human planning to differ from animal planning in a way that supports language is in the event representation itself.
“Grounding” Actions and Affordances

• The fact that actions and objects have (fairly) simple types doesn’t mean that the actions themselves are simple.

• A box falling is not a volitional action, and has perceptual preconditions like a looming flow-field. The event is a complex conjunction of entailments of a box falling, such as a hurting event, and the consequent state concerns issues other than the mere lowering of the box’s position.

• The ramified nature of this dynamic event knowledge is the reason that languages can vary in the way they carve the conceptual representation at the joints to define their (much terser) lexical semantics.

• E.g. English run across the road vs. French traverser la rue à la course.

• To understand the connection between planning and semantics, we need to better understand the grounded event representation.
III: The Problem of Content

- Linguists and the Artificial Intelligence have notably failed to devise a semantics that captures this cross-linguistic variety.

(1) Thomason, 1974: $\forall x [\text{bug}'x \Rightarrow \exists y [\text{plants}(y) \land \text{kill}'y x]]$
McCawley, 1968: $[s \text{CAUSE BUGS}[s \text{BECOME}[s \text{NOT}[s \text{ALIVE PLANTS}] v] v]$
Dowty, 1979: $[\text{CAUSE}[\text{DO BUGS} \otimes][\text{BECOME} \neg[\text{ALIVE PLANTS}]] v]$
Talmy, 2000: Bugs ARE-the-AUTHOR''-OF [plants RESULT-TO-die]
Van Valin, 2005: $[do'(\text{bugs}', \otimes)[\text{CAUSE}[\text{BECOME}[\text{dead'}(\text{plants})]] v] v]$
Goddard, 2010: Bugs do something to PLANTS; because of this, something happens to PLANTS at the same time; because of this, something happens to PLANTS' body; because of this, after this PLANTS are not living anymore.

- Can we identify the primitive concepts automatically, as hidden variables?
Two Approaches

- Clustering by Collocation (Landauer and Dumais, 1997; Baroni and Zamparelli, 2010; Grefenstette and Sadrzadeh, 2011; Padó and Lapata, 2007; Mitchell and Lapata, 2008; Mikolov et al., 2013).
  - Composition via Linear Algebraic Operations
  - Good for underspecification and disambiguation

- Clustering by Denotation (Lin and Pantel, 2001; Hovy et al., 2001), using sentences involving identifiable Named Entities (Lewis and Steedman, 2013a; Reddy et al., 2014)
  - Composition via traditional Logical Operators
  - Good for inference.
Clustered Entailment Semantics

- We must distinguish paraphrase from entailment.

- \( X_{\text{person}} \text{ elected to } Y_{\text{office}} \) entails \( X_{\text{person}} \text{ ran for } Y_{\text{office}} \) but not vice versa.

- The paraphrase relation depends on global properties of the named entity relation graph.

- Lewis (2015); Lewis and Steedman (2014b) apply the entailment graphs of Berant et al. (2012) to generate more articulated entailment structures.
Local Entailment Probabilities

- The typed named-entity technique is applied to (errorfully) estimate local probabilities of entailments:
  a. \( p(\text{conquer } xy \Rightarrow \text{invade } xy) = 0.9 \)
  b. \( p(\text{invade } xy \Rightarrow \text{attack } xy) = 0.8 \)
  c. \( p(\text{conquer } xy \Rightarrow \text{attack } xy) = 0.4 \)
  d. \( p(\text{bomb } xy \Rightarrow \text{attack } xy) = 0.7 \)
    (etc.)
Global Entailments

- The local entailment probabilities are used to construct an entailment graph using integer linear programming with $\pm$ weights around $p = 0.5$ with the global constraint that the graph must be closed under transitivity.

- Thus, (c) will be included despite low observed frequency, while other low frequency spurious local entailments will be excluded.

- Cliques within the entailment graphs are collapsed to a single paraphase cluster relation identifier.
• A simple entailment graph for relations between countries.
Lexicon

- The lexicon obtained from the entailment graph
  
  \[
  \text{attack} := (S\backslash NP)/NP : \lambda x\lambda y\lambda e.\text{rel}_1 xy e \\
  \text{bomb} := (S\backslash NP)/NP : \lambda x\lambda y\lambda e.\text{rel}_1 xy e \land \text{rel}_4 xy e \\
  \text{invade} := (S\backslash NP)/NP : \lambda x\lambda y\lambda e.\text{rel}_1 xy e \land \text{rel}_2 xy e \\
  \text{conquer} := (S\backslash NP)/NP : \lambda x\lambda y\lambda e.\text{rel}_1 xy e \land \text{rel}_2 xy e \land \text{rel}_3 xy e \\
  \text{annex} := (S\backslash NP)/NP : \lambda x\lambda y\lambda e.\text{rel}_1 xy e \land \text{rel}_2 xy e \land \text{rel}_3 xy e \\
  \]

- These logical forms support correct inference under negation, such as that \text{conquered} entails \text{attacked} and \text{didn’t invade} entails \text{didn’t conquer}.

- To answer a question “Did X invade Y” we look for sentences which subsume the conjunctive logical form \text{rel}_2 \land \text{rel}_1, or satisfy its negation \neg \text{rel}_2 \lor \neg \text{rel}_1.

\begin{itemize}
  \item Note that if we know that \text{invasion-of} is a paraphrase of \text{invade} = \text{rel}_2, we also know \text{invasion-of} entails \text{attack} = \text{rel}_1.
\end{itemize}
Lexicon

- Primitives like $rel_3$ correspond to “hidden” semantic primitives that distinguish these concepts.

- If we do the machine-reading cross-linguistically (Lewis and Steedman, 2013b), we will see that some of them correspond to universal elements masked in English (see earlier remarks about run across the road).

- Others will be more arcane.
## Results (Lewis and Steedman, 2014b)

<table>
<thead>
<tr>
<th>System</th>
<th>Accuracy (all)</th>
<th>AUC (all)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Majority Class</td>
<td>56.8%</td>
<td>0.46</td>
</tr>
<tr>
<td>Non Compositional</td>
<td>57.4%</td>
<td>0.48</td>
</tr>
<tr>
<td>CCG Baseline</td>
<td>57.8%</td>
<td>0.46</td>
</tr>
<tr>
<td>CCG ChineseWhispers</td>
<td>58.0%</td>
<td>0.50</td>
</tr>
<tr>
<td>VectorMultiplicative</td>
<td>61.3%</td>
<td>0.51</td>
</tr>
<tr>
<td>VectorAdditive</td>
<td>63.5%</td>
<td>0.57</td>
</tr>
<tr>
<td><strong>CCG Entailment Graphs</strong></td>
<td><strong>64.0%</strong></td>
<td><strong>0.58</strong></td>
</tr>
<tr>
<td>CCG Entailment Graphs+ Implicative Verb Lexicon</td>
<td><strong>65.0%</strong></td>
<td><strong>0.59</strong></td>
</tr>
</tbody>
</table>
Philosophical Reflections

• Our hidden relations resemble “meaning postulates”, such as the one that says that in every model where X is a bachelor, X is also unmarried and male.

• Carnap (1952) introduced meaning postulates in support of Inductive Logic, including a model of Probability, basically to keep the model small and consistent.

• This suggests that our semantic representation expresses an a pragmatic empiricist view of “analytic meaning”, of the kind advocated by Quine (1951).

• It can also be viewed as a statistical and text-based approach to treating “meaning as use” (Wittgenstein, 1953).
IV: Hanging Language onto Planning

- We saw that (partially) searching the plan graph is an intrinsically recursive process.
- So we need at least a push-down automaton (PDA) to keep track of it.

❓ Is a PDA expressive enough?

- It depends on the class of plans
- If the set of plan- types is unbounded, then a PDA is not enough.
- (For the same reason that a PDA is not enough for a PS grammar with unboundedly many non-terminals.)
Language and Cooperative Planning

- Collaborative Plans are functions over arbitrary numbers of other agents:

  (2) a. Find someone to help to mind the baby
      b. Find someone to promise to help to mind the baby
      c. Find someone to ask to promise to help to mind the baby.
      (etc.)

Searching a graph with unboundedly many node-types needs an Embedded PDA (EPDA), in which the stack of the PDA can include stack-valued elements.

- Collaborative planning with other minds provides not only the only known motivation for language (Tomasello, 1999), but also the characteristic automaton that supports its use.

- So we should look at the grammar of sentences such as (2).
Combinatory Categorial Grammar

- CCG (Steedman, 2000; Bozşahin, 2012) eschews language-specific syntactic rules like (3) for English.

\[
\begin{align*}
S & \rightarrow NP \quad VP \\
VP & \rightarrow TV \quad NP \\
TV & \rightarrow \{proved, \; found, \; met, \; \ldots\}
\end{align*}
\]

- Instead, all language-specific syntactic information is lexicalized, via lexical entries like (4) for the English transitive verb, where \(\text{met}'\) is an abbreviation for some conjunction of clustered entailments of the kind discussed earlier:

\[
\text{met} := (S/\text{NP})/\text{NP} : \text{met}'
\]

- In CCG, syntactic projection from the lexicon is mediated by type-raising \(T\) and composition \(B\).
The Lexicon

- The syntactic “category” identifies the transitive verb as a function, and specifies the type and directionality of its arguments and the type of its result. For Turkish:

\[(5) \text{rastladı} := (S\backslash NP)\backslash NP : \text{met}'\]

This is a good example of the different ways languages carve meaning at the joints. \textit{rastladı} means something like “came across”, and is distinct from reciprocal “meet” \textit{taniştı} which is the same word in English.

- A cross-linguistic clustered entailment semantics, obtained from multilingual machine-reading, would split these meanings into two distinct clusters, rather than one \textit{met}'.

\[\text{Steedman} \quad \text{2nd Intl. Symp. on Brain and Cognitive Science, ODTÜ Ankara} \quad 19\text{th April 2015}\]
Type Raising as Case

- We will assume that type-raising in the form of case is a universal primitive of grammar, as it is for planning in the form of affordance.

- All noun-phrases (NP) like “Harry” are (polymorphically) type-raised.

- In Japanese and Latin this is the job of case morphemes like nominative -ga and -us. (Same for Turkish, except nominative is null.)

- In English NPs are ambiguous as to case, and must be disambiguated by the parsing model (a.k.a. “structural case”).
Syntactic Derivation

• (6)  \[ \frac{S/(S\backslash NP)^T}{T} \frac{(S\backslash NP)/NP}{T} \frac{(S\backslash NP)/(S\backslash NP)/NP)}{T} \]

\[ \frac{S\backslash NP}{<} \]

\[ \frac{S}{>} \]

• (7)  \[ \frac{S/(S\backslash NP)^T}{T} \frac{(S\backslash NP)/NP}{T} \frac{(S\backslash NP)/(S\backslash NP)/NP)}{T} \]

\[ \frac{S/NP}{>B} \]

\[ \frac{S}{>} \]
“Surface Compositional” Semantics

• (8) \[
\frac{S/(S\backslash NP)}{T} (S\backslash NP)/NP (S\backslash NP)/(S/(S\backslash NP)/NP)
\]
  \[\vdash \lambda p\cdot p \text{ harry}' : \text{met}' : \lambda p\cdot p \text{ sally}' \]
  \[\frac{}{S\backslash NP : \text{met}' \text{ sally}'} \frac{}{S : \text{met}' \text{ sally}' \text{ harry}'} \]

• (9) \[
\frac{S/(S\backslash NP)}{T} (S\backslash NP)/NP (S\backslash NP)/(S/(S\backslash NP)/NP)
\]
  \[\vdash \lambda p\cdot p \text{ harry}' : \text{met}' : \lambda p\cdot p \text{ sally}' \]
  \[\frac{}{S\backslash NP : \lambda x.\text{met}' x \text{ harry}'} \frac{}{S : \text{met}' \text{ sally}' \text{ harry}'} \]
Relativization

- (10) (The woman)  
  \[
  \lambda p \lambda n \lambda x. p x \land n x \quad \lambda p. h a r r y' \quad : \text{met}'
  \]
  
  \[
  S/NP \\
  \lambda x. \text{met}' x h a r r y' \\
  \]
  
  \[
  N/N : \lambda n \lambda x. \text{met}' x h a r r y' \land n x
  \]

- (11) (The woman)  
  \[
  \lambda p \lambda n \lambda x. p x \land n x \quad \lambda p. h a r r y' \quad : \text{met}'
  \]
  
  \[
  S/NP \\
  \lambda x. \text{met}' x h a r r y' \\
  \]
  
  \[
  S/S \\
  \]
  
  \[
  S/NP \\
  \]
  
  \[
  N/N
  \]
Coordination

- (12) **give** Harry a book **and** Sally a record

\[ \text{Coordination} \]

\[ \text{CCG reduces the linguists’ MOVE and COPY/DELETE to adjacent MERGE} \]
CCG is “Near Context-Free”

• The composition rules in CCG are generalized to $B^2$, and “crossed composition” $B \times$

• The combination of type-raising and generalized composition yields a permuting and rebracketing calculus closely tuned to the needs of natural grammar.

• CCG and TAG are provably weakly equivalent to Linear Indexed Grammar (LIG) Vijay-Shanker and Weir (1994).

• Hence they are not merely “Mildly Context Sensitive” (Joshi 1988), but rather “Near Context Free,” or “Type 1.9” in the Extended Chomsky Hierarchy.
## The Extended Chomsky Hierarchy

<table>
<thead>
<tr>
<th>Language Type</th>
<th>Automaton</th>
<th>Rule-types</th>
<th>Exemplar</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 0: RE</td>
<td>Universal Turing Machine</td>
<td>$\alpha \rightarrow \beta$</td>
<td></td>
</tr>
<tr>
<td>Type 1: CS</td>
<td>Linear Bound Automaton (LBA)</td>
<td>$\phi A \psi \rightarrow \phi \alpha \psi$</td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>Nested Stack Automaton (NSA)</td>
<td>$A_{[i,...]} \rightarrow \phi B_{[i,...]} \psi C_{[i,...]} \xi$</td>
<td>$a^{2n}$</td>
</tr>
<tr>
<td>LCFRS/MCF</td>
<td>$i$th-order EPDA</td>
<td>$A_{[[i,i...,]]} \rightarrow \phi B_{[[i...,]]} \psi$</td>
<td>$P(a^n b^n c^n)$</td>
</tr>
<tr>
<td>LI</td>
<td>Embedded PDA (EPDA)</td>
<td>$A_{[i,...]} \rightarrow \phi B_{[i,...]} \psi$</td>
<td>$a^n b^n c^n$</td>
</tr>
<tr>
<td>Type 2: CF</td>
<td>Push-Down Automaton (PDA)</td>
<td>$A \rightarrow \alpha$</td>
<td>$a^n b^n$</td>
</tr>
<tr>
<td>Type 3: FS</td>
<td>Finite-state Automaton (FSA)</td>
<td>$A \rightarrow {a B}$</td>
<td>$a^n$</td>
</tr>
</tbody>
</table>

*All higher language classes properly contain all lower except LCFRS and I, which properly intersect.*
Zürich German is Strongly Near Context-Free

(13) das mer em Hans es huus hälfed aastriiche
that we—NOM Hans—DAT the house—ACC helped paint

NP_{nom}^{\downarrow} \quad NP_{dat}^{\uparrow} \quad NP_{acc}\quad ((S_{+SUB}\backslash NP_{nom}\backslash NP_{dat}\backslash VP) / VP) / VP_{acc} \rightarrow B_{x}

(S_{+SUB}\backslash NP_{nom}\backslash NP_{dat}\backslash NP_{acc})

S_{+SUB}

“that we helped Hans paint the house”

• The following is correctly also allowed (Shieber, 1985):

(14) Das mer em Hans hälfed es huus aastriiche.
Zürich German is Strongly Near Context-Free

(15) das mer d‘chind em Hans es huus lönd hälfe aastriiche

\[
\begin{align*}
&\text{NP}_{\text{nom}}^\downarrow \quad \text{NP}_{\text{acc}}^\downarrow \quad \text{NP}_{\text{dat}}^\downarrow \quad \text{NP}_{\text{acc}}^\downarrow \\
&\text{NP}_{\text{acc}} \quad \text{NP}_{\text{acc}} \quad \text{NP}_{\text{dat}} / \text{VP} \quad \text{VP} / \text{VP} \\
&\text{B}_2 \\
&\text{B}_x \\
&\text{S}_{\text{SUB}} \quad \text{NP}_{\text{nom}} \quad \text{NP}_{\text{acc}} \\
&\text{S}_{\text{SUB}} \\
\end{align*}
\]

“that we let the children help Hans paint the house”

- Again, other word orders are correctly allowed.
- Constituents like “es huus lönd hälfe aastriiche” are homologous to collaborative plans like earlier “Find someone to let someone help someone mind the baby”.

Steedman
2nd Intl. Symp. on Brain and Cognitive Science, ODTÜ Ankara
19th April 2015
Conclusion (I)

- The lexicon is the only locus of language specific information in the grammar.

- The universal projective syntactic component of natural language grammar is based on the combinators $B, T$.

- In evolutionary terms, these combinators were provided ready-made, by a sensory motor planning mechanism most of which we share with a number of animals.

- The problem of parsing is automata-theoretically equivalent to the problem of planning for cooperation with other minds.

- Both of the latter abilities seem unique to humans.
Conclusion (II)

- The following progression over the last 200M years of vertebrate evolution may have resulted in an essentially instantaneous recent emergence of language:

1. Pure reactive planning with non-recursive KR (finite-state);
2. (Forward-chaining, breadth-first) deliberative planning with non-recursive KR requiring composition, type-raising, and a (simulated) PDA for search;

- A PDA also supports recursive concepts in KR. But a PDA alone isn’t enough to support human planning and humean language, which other animals lack.
Conclusion (III)

• We must postulate the following further developments:

3. Human planning is characterized by the use of plan variables corresponding to unknown provided by external agencies such as phone-books, Google search, or other human beings.
   - Planning with the particular recursive concepts that are necessary human collaboration for purposes like neotenic child-reading generates plans with unboundedly many plan variables (agents) (Hrdy, 2009; Steedman, 2014).

4. Such planning requires a (simulated) embedded PDA

5. The EPDA immediately supports near-context-free Natural Language Grammar, as attested by English, Turkish, and Zürich German
   - This can happen without any further evolutionary work other than a little specialization of the vocal tract.
Appendix: Practical Applications of CCG

• It was widely expected in the '80s that the degree of derivational ambiguity CCG allows would make it completely impractical for parsing.

• However, any grammar that covers these data has the same problem.

• The universal recognition in the '90s of the need for statistical modeling in NLP was a great leveler.

• With such models, CCG can be parsed as fast and as accurately as anything else—

• —with the advantage of a surface compositional semantics including discontinuity and “non-projectivity”.
Practical Applications of CCG

• Many applications exploit the “surface compositional” semantics of CCG—for example:
  – Hockenmaier (2003); Clark and Curran (2004); Çakıcı and Steedman (2009); Lewis and Steedman (2014a) provide publicly available efficient parsers trained on WSJ.
  – Birch et al. (2007); Hassan et al. (2009); Mehay and Brew (2012) use CCG for statistical machine translation
  – Prevost (1995); White (2006) apply it to sentence realization
  – Briscoe (2000); Kwiatkowski et al. (2012); Krishnamurthy and Mitchell (2012) apply it to semantic parsing and language acquisition
  – Bos and Markert (2005); Lewis and Steedman (2013a,b, 2014b) apply it to open-domain question answering and entailment
References


Berant, Jonathan, Dagan, Ido, Adler, Meni, and Goldberger, Jacob, 2012. “Efficient Tree-Based Approximation for Entailment Graph Learning.” In


Lewis, Mike and Steedman, Mark, 2014b. “Combining Formal and Distributional


Mehay, Denis and Brew, Chris, 2012. “CCG Syntactic Reordering Models for


