Plans, actions, and sentences

Cem Bozşahin
www.metu.edu.tr/~bozsahin

Cognitive Science Department, Informatics Institute, METU/ODTÜ

November 10, 2017

joint work with Nick Findler, Mark Steedman, Deniz Zeyrek, Işın Demirşahin, Umut Özge, Tzu-Ching Kao, Ahmet Üstün, Çağrı Şakiroğulları, Nihan Soycan, Burcu Can

One observation

The relation between plans–actions, and discourse–sentences,

seems to be too close to be of coincidence.

Introduction

It has to do with thematic structure, or lack of it.

1. The closer we look at the relation between plans and scripts,
   - the closer it looks like the relation between discourse and sentences.
2. The closer we look at the relation between discourse and sentences,
   - the closer it looks like the relation between plans and scripts.
3. Can there be common mechanisms?

   for discourse and planning

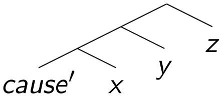
   for clauses and scripts

I have a plan!

1. Plans are functions to map states of the world to states of the world by a planner.
2. An AI classic: painting the ladder and painting the ceiling. Sacerdoti (1975)
   - plan : needspainting x $\rightarrow$ painted x
   - scripts:
     get-paint: getter, gotten, beneficiary
     get-ladder: getter, beneficiary
     paint-x: painter, x-ee, wet-x
     climb-x: climber, busy-x
3. Plan: no thematic structure (but goals and states)
4. Script: thematic structure (predicates rather than goals)states?

   Schank and Abelson (1977)

1. Syntax and thematic structure (from Bozşahin, 2017)
   1. The window broke. $init'(broken'\,w')$
   2. The stone broke the window. $\lambda z.cause'(init'(broken'\,w'))st'\,z$
   3. The man broke the window with a stone. $cause'(init'(broken'\,w'))st'\,man'$
2.
   1. The stone broke the window in a storm. (noncausative agent?)
   2. $\sharp$the storm broke the window with a stone.
   3. the storm broke the window.
3. Where is syntax?



1. The window broke. $S\backslash NP$: $\lambda x.init'(broken'x)$ $init'(broken'w')$
2. The stone broke the window.
   $(S\backslash NP)/NP$: $\lambda x\lambda y\lambda z.cause'(init'(broken'x)yz$
   $\lambda z.cause'(init'(broken'w'))st'\,z$ (causer?)
3. The man broke the window with a stone.
   $(S\backslash NP)/PP/NP$: $\lambda x\lambda y\lambda z.cause'(init'(broken'x)yz$
   $cause'(init'(broken'w'))st'\,man'$
4. Thematic difference corresponds with syntactic type.

1. Steedman and Bozşahin (2017) :
   All natural grammars can be radically lexicalized this way.
   1. maintain constituency while projecting thematic structure and logical form
   2. Projecting structure from the lexicon must rise above function application.

$$
\begin{array}{ccc}
\text{The stone} & \text{broke} & \text{the window.} \\
\overline{NP: st'} & \overline{(S\backslash NP)/NP: \lambda x \lambda y \lambda z.cause'(init'(broken'x)yz)} & \overline{NP: w'} \\
 & \multicolumn{2}{c}{\underline{\hspace{10em}}{>}} \\
 & \multicolumn{2}{c}{S\backslash NP: \lambda y \lambda z.cause'(init'(broken'w')yz)} \\
\multicolumn{3}{c}{\underline{\hspace{20em}}{<}} \\
\multicolumn{3}{c}{S: \lambda z.cause'(init'(broken'w')st'z)}
\end{array}
$$

$$\dfrac{\text{Mary}}{S/(S\backslash NP_{3s}) : \lambda p.p\,m'} \quad \dfrac{\text{likes}}{(S\backslash NP_{3s})/NP : \lambda x\lambda y.like'xy} \quad \text{and John hates} \quad \dfrac{\text{cats.}}{S\backslash(S/NP)}$$

$$\dfrac{S/(S\backslash NP_{3s}) : \lambda p.p\,m' \quad (S\backslash NP_{3s})/NP : \lambda x\lambda y.like'xy}{S/NP : \lambda x.like'x\,mary'}{>}\mathbf{B}$$

$$X/Y : f \quad Y/Z : g \;\Rightarrow\; X/Z : \lambda x.f(gx) \qquad (>\mathbf{B})$$

$$
\begin{array}{ccccc}
\textit{the} & \textit{man} & \textit{who} & \textit{Mary} & \textit{loved} \\
\hline
(S/(S\backslash NP))/N & N & (N\backslash N)/(S_{\mathrm{fin}}/NP) & S/(S\backslash NP_{\mathrm{3s}}) & (S_{\mathrm{fin,3s}}\backslash NP)/NP \\
: \lambda P\lambda Q.(the'x)and'(Px)(Qx) & : man' & \lambda P\lambda Q\lambda x.and'(Px)(Qx) & : \lambda P.P\,mary' & : \lambda x\lambda y.loved'xy
\end{array}
$$

$$\overline{\qquad S/(S\backslash NP_{\mathrm{3s}}) \quad (S_{\mathrm{fin,3s}}\backslash NP)/NP \qquad}{>}\mathbf{B}$$

$$S_{\mathrm{fin}}/NP\colon \lambda y.loved'y\ mary'$$

$$\overline{\qquad\qquad\qquad}{>}$$

$$N\backslash N : \lambda Q\lambda x.and'(loved'x\ mary')(Qx)$$

$$\overline{\qquad\qquad\qquad}{<}$$

$$N\colon \lambda x.and'(loved'x\ mary')(man'x)$$

$$\overline{\qquad\qquad\qquad}{>}$$

$$S/(S\backslash NP)\colon \lambda Q.(the'x)and'(and'(loved'x\ mary')(man'x))(Qx)$$

This book I think Mary knows John claims the library owned

.. makes a fine read if you can find it.

Other rules of projection

$$
\begin{array}{llll}
\textit{(the books) which I have} & \textit{read} & \textit{without} & \textit{understanding} \\
\cline{2-4}
 & VP/NP & (VP\backslash VP)/C_{\text{ing}} & C_{\text{ing}}/NP \\
\cline{3-4}
 & & \multicolumn{2}{c}{(VP\backslash VP)/NP \quad \mathbf{B}} \\
\cline{2-4}
 & \multicolumn{3}{c}{VP/NP \quad \mathbf{S}}
\end{array}
$$

$$Y/Z : g \quad (X\backslash Y)/Z : f \quad \Rightarrow \quad X/Z : \lambda x.f\,x(g\,x) \qquad (<\mathbf{S}_{\times})$$

1. Combinatory Categorial Grammar
   Steedman (2012); Bozşahin (2012); Steedman and Bozşahin (2017)
   1. Conjecture: These combinators are all that is needed to project constituency and thematic structure from lexicon onto surface structure.
   2. Child CCG grammars are learnable from expression-meaning pairs (Abend et al., 2017).

What about discourse?

1. Do we see same kind of projection of thematic structure?
2. Zikánová et al. (2015):22
   " The fact that the discourse project in Prague is based on the previous annotation of underlying syntax reflects the basic assumption, that, from the cognitive viewpoint, the semantics within a sentence is the same as the semantics of discourse relations."
3. This is not corroborated empirically.
4. It is a methodological assumption.
5. The distinction seems similar to distinction in Plans and Scripts.

Tree structure in discourse: syntactic?

Summarizing from Demirşahin (2015)

1. Hobbs (1985): coherence relations build successive trees connected at peripheries.
2. RST (Mann and Thompson, 1988): single tree with predefined rhetorical relations.
   1. Rhetorical relations are adjacent.
   2. They can be asymmetric (nucleus vs satellite), or symmetric (among nuclei).
3. DLTAG/PDTB (Webber, 2004; Prasad et al., 2014): discourse connectives drive discourse. They all take two arguments.
4. Asher (1993): Presupposition and entailment among abstract objects drive discourse.
5. Halliday and Hasan (1976) Non-structural ties (reference, substitution, lex cohesion etc) hold discourse together.

PDTB investigation of discourse trees

1. Lee et al. (2006, 2008): instances of apparent tree violations (partially overlapping arguments, crossing) can be explained away by anaphora and attribution.
2. Aktaș et al. (2010); Zeyrek et al. (2017) concur from Turkish written discourse.
3. Demirșahin (2015) from spoken dialogs.

Example from Zeyrek et al. (2017):

Ceza, Telekom'un iki farklı internet alt yapısı pazarında tekel konumunu kötüye kullandığı **için** ve uydu istasyonu işletmeciliği pazarında artık tekel hakkı kalmadığı halde rakiplerinin faaliyetlerini zorlaştırdığı için verildi.

The penalty was given **because** Telekom abused its monopoly status in the two different internet infrastructure markets and because it caused difficulties with its rivals' activities although it did not have a monopoly status in the satellite management market anymore.

1. Zeyrek et al. (2017): Ceza verildi wraps in the inner material. It is still function application. Ceza moves away from rheme to theme.
2. Demirşahin et al. (2013) All violations of tree structure are apparent: all are function application.

This book	I	think	Mary	knows	John	claims	the library	owned
$NP/(S/NP)$	$S/(S\backslash NP)$	$(S\backslash NP)/S'$	$S'/(S\backslash NP)$	$(S\backslash NP)/S'$	$S'/(S\backslash NP)$	$(S\backslash NP)/S'$	$S'/(S\backslash NP)$	$(S\backslash NP)/NP$

$$\underline{\quad}{>}\mathbf{B}\quad S/S'$$
$$\underline{\quad}{>}\mathbf{B}\quad S/(S\backslash NP)$$
$$\underline{\quad}{>}\mathbf{B}\quad S/S'$$
$$\underline{\quad}{>}\mathbf{B}\quad S/(S\backslash NP)$$
$$\underline{\quad}{>}\mathbf{B}\quad S/S'$$
$$\underline{\quad}{>}\mathbf{B}\quad S/(S\backslash NP)$$
$$\underline{\quad}{>}\mathbf{B}\quad S/NP$$
$$\underline{\quad}{>}\quad NP$$

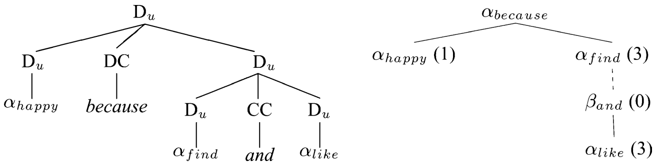
1. Forbes-Riley et al. 2006:68

   "It is only in an approach to discourse which has a syntax that is distinct from its semantics [..] that the concept of a syntax-semantic interface is relevant. For example, rhetorical structure theory [..] does not have a semantics that is distinct from its syntax."
2. They take discourse connectives to be predicates.
3. with same mechanism of compositionality as clause-level semantics.

1. Associating propositional labels (l) and variables (s) with the semantics of discourse units.
2. These predicates are always binary.
3. Syntactic predicates are not.
4. It seems that they are not because we cannot fix for all of them how many arguments would make them propositions.
5. Discourse trees do not embed like clause-level trees.

Example DLTAG/PDTB analysis (ibid.)

(20) a. (What are two reasons for Mary being happy?)
b. *Mary is happy* because [*she found a job*] *and* [*she likes it*].



l_1: because'(l_3,l_4), l_2: and'(l_4,l_5), l_3: happy'(m), l_4: find'(m,j), l_5: like'(m,j)

Figure 16 D-LTAG derived and derivation trees and semantic representation for (20).

$$
\begin{array}{ccccc}
\textit{the} & \textit{man} & \textit{who} & \textit{Mary} & \textit{loved} \\
\hline
(S/(S\backslash NP))/N & N & (N\backslash N)/(S_{\mathrm{fin}}/NP) & S/(S\backslash NP_{3s}) & (S_{\mathrm{fin,3s}}\backslash NP)/NP \\
: \lambda P \lambda Q.(the' x) and' (Px)(Qx) & : man' & \lambda P \lambda Q \lambda x.and'(Px)(Qx) & : \lambda P.P mary' & : \lambda x \lambda y.loved' xy
\end{array}
$$

$$
\xrightarrow{\quad\quad\quad\quad\quad\quad}{>}\mathbf{B}
$$

$$
S_{\mathrm{fin}}/NP: \lambda y.loved' y\ mary'
$$

$$
\xrightarrow{\quad\quad\quad\quad\quad\quad}{>}
$$

$$
N\backslash N : \lambda Q \lambda x.and'(loved' x\ mary')(Qx)
$$

$$
\xrightarrow{\quad\quad\quad\quad\quad\quad}{<}
$$

$$
N: \lambda x.and'(loved' x\ mary')(man' x)
$$

$$
\xrightarrow{\quad\quad\quad\quad\quad\quad}{>}
$$

$$
S/(S\backslash NP): \lambda Q.(the' x) and'(and'(loved' x\ mary')(man' x))(Qx)
$$

Discursive 'because' and 'and' have no thematic structure.

This is not what we get in the semantics of clause predicates.

$$\begin{array}{lccc}
\textit{(the books) which I have} & \textit{read} & \textit{without} & \textit{understanding} \\
\cline{2-4}
 & VP/NP & (VP\backslash VP)/C_{\mathrm{ing}} & C_{\mathrm{ing}}/NP \\
\cline{3-4}
 & & \multicolumn{2}{c}{(VP\backslash VP)/NP \quad \mathbf{B}} \\
\cline{2-4}
 & \multicolumn{3}{c}{VP/NP \quad \mathbf{S}}
\end{array}$$

Interim summary and suggestion

1. Discourse predicates are always binary.
2. Syntactic predicates are not.
3. It seems that they are not because we cannot fix for all of them how many arguments would make them propositions.
    1. Plans are like discourse predicates. They are always binary functions.

planner $\times$ soa $\rightarrow$ planner $\times$ soa

    2. Scripts are not. They need to know who does what to whom.
    3. Discourse trees do not embed like clause-level trees.
    4. Neither do plans. Scripts do.

How can we show that it is scripts that embed, not plans

1. Planning in AI/Cogsci is formalisms galore (see Ghallab et al. 2004, 2016).

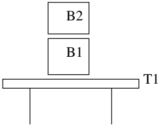
   One classic: Miller et al. (1960)
2. CCG has been used for planning before:
   1. for explanation (Steedman, 2002; Geib, 2015)
   2. modeling (Geib et al., 2006; Geib and Goldman, 2009; McMichael et al., 2006)
3. What we need is not a mechanism distinct from linguistic CCG.

Conjecture: meaningful sequences

1. $B\backslash A$ means A may enable B.
2. B/A means A may lead to B.
3. Two basic plan categories: States (S) and Terms (T)
4. When these modals are lexically specified, they begin to bear thematic roles.
5. For that we need a script to realize a plan.

1. STRIPS (Fikes and Nilsson, 1971) is the grandfather of all planners.
2. Mainly because it handled the Frame Problem with procedural semantics.

John
B1
pick-up



$$\text{pick-up} \quad := \quad (S\backslash T)\backslash T : \lambda x \lambda y \lambda z.pickup' xy$$
$$\texttt{pre:} \; inhand(y, nil), clear(x), block(x), on(x, z)$$
$$\texttt{add:} \; inhand(y, x), clear(z)$$
$$\texttt{del:} \; inhand(y, nil), on(x, z)$$

1. Agents, patients are inferred from LF of the script, not from plan structure (conditions of the world).
2. Plan states and script structure are distinct: e.g. no *(Pick-up\Obj)\Agt*

forward-backward reasoning may be optional

$$\dfrac{\text{John}}{\begin{array}{l} T: john' \\ \texttt{pre:-} \\ \texttt{add:-} \\ \texttt{del:-} \end{array}} \quad \dfrac{\text{B2}}{\begin{array}{l} T: b2' \\ \texttt{pre:-} \\ \texttt{add:-} \\ \texttt{del:-} \end{array}} \quad \dfrac{\text{pick-up}}{\begin{array}{ll} \multicolumn{2}{c}{(S\backslash T)\backslash T: \lambda x\lambda y\lambda z.pickup'xy} \\ \texttt{pre:} & inhand(y, nil), clear(x), block(x), on(x, z) \\ \texttt{add:} & inhand(y, x), clear(z) \\ \texttt{del:} & inhand(y, nil), on(x, z) \end{array}}$$

$$\xrightarrow{\text{apply}}$$

$$\begin{array}{ll} \multicolumn{2}{c}{S\backslash T: \lambda y\lambda z.pickup'b2'y} \\ \texttt{pre:} & inhand(y, nil), clear(b2'), block(b2'), on(b2', z) \\ \texttt{add:} & inhand(y, b2'), clear(z) \\ \texttt{del:} & inhand(y, nil), on(b2', z) \end{array}$$

$$\xrightarrow{\text{apply}}$$

$$\begin{array}{ll} \multicolumn{2}{c}{S: \lambda z.pickup'b2'john'} \\ \texttt{pre:} & inhand(john', nil), clear(b2'), block(b2'), on(b2', z) \\ \texttt{add:} & inhand(john', b2'), clear(z) \\ \texttt{del:} & inhand(john', nil), on(b2', z) \end{array}$$

backward chaining may be required

pick-up	B2	John
$(S/T)/T$: $\lambda x \lambda y \lambda z.pickup' x y$	T: $b2'$	T: $john'$
`pre:` $inhand(y, nil), clear(x), block(x), on(x, z)$	`pre:-`	`pre:-`
`add:` $inhand(y, x), clear(z)$	`add:-`	`add:-`
`del:` $inhand(y, nil), on(x, z)$	`del:-`	`del:-`

apply

S/T: $\lambda y \lambda z.pickup' b2' y$
`pre:` $inhand(y, nil), clear(b2'), block(b2'), on(b2', z)$
`add:` $inhand(y, b2'), clear(z)$
`del:` $inhand(y, nil), on(b2', z)$

apply

S: $\lambda z.pickup' b2' john'$
`pre:` $inhand(john', nil), clear(b2'), block(b2'), on(b2', z)$
`add:` $inhand(john', b2'), clear(z)$
`del:` $inhand(john', nil), on(b2', z)$

The inference mechanism depends on the sequence.

Planning can rise above applicative structures too

Suppose that John wants to fly after someone buys him a ticket.

John
T
$: john'$
pre: -
add: -
del: -

fly
$(S/S)\backslash T$
$: \lambda x \lambda s \lambda z.takeflight'\ s\ z\ x$
pre: $able(x, s), flight(z)$
add: $at(x, dest(z)), at(z, dest(z))$
del: $at(z, orig(z)), at(x, here)$

buy
$(S/T)/T$
$: \lambda x \lambda y \lambda z.buy'\ zxy$
pre: $payable(x), funds(F), have(y, F)$
add: $have(y, x), able(y, z)$
del: -

ticket
T
$: ticket'$
pre: -
add: -
del: -

John fly ———apply

S/S
$: \lambda s \lambda z.takeflight'\ s\ z\ john'$
pre: $able(john', s), flight(z)$
add: $at(john', dest(z)), at(z, dest(z))$
del: $at(z, orig(z)), at(john', here)$

buy ticket ———apply

S/T
$: \lambda y \lambda z.buy'\ z\ ticket'\ y$
pre: $payable(ticket'), funds(F), have(y, F)$
add: $have(y, ticket'), able(y, z)$
del: -

———compose

S/T
$: \lambda x \lambda y \lambda z.takeflight'(buy'\ z\ ticket'\ x)\ y\ john'$
pre: $able(john', buy'(z, ticket', x)), flight(y), payable(ticket'), funds(F), have(x, F)$
add: $at(john', dest(y)), at(y, dest(y)), have(x, ticket'), able(x, z)$
del: $at(y, orig(y)), at(john', here)$

John-fly-buy-ticket

1. Now consider the case in which the sequence does not give rise to an expectation that someone else will buy the ticket.
2. John appears once in the sequence, and does two acts.
3. This is not always an intrinsic property of the scripts involved. It may arise from the semantics of sequencing itself.

John: T : $john'$
pre:- add:- del:-

fly: $(S/S)\backslash T$: $\lambda x \lambda s \lambda z.takeflight'\ s\ z\ x$
pre: $able(x,s), flight(z)$
add: $at(x, dest(z)), at(z, dest(z))$
del: $at(z, orig(z)), at(x, here)$

buy: $(S\backslash T)/T$: $\lambda x \lambda y \lambda z.buy'\ zxy$
pre: $payable(x), funds(F), have(y,F)$
add: $have(y,x), able(y,z)$
del:-

ticket: T : $ticket'$
pre:- add:- del:-

apply (buy + ticket):

$S\backslash T$: $\lambda y \lambda z.buy'\ z\ ticket'\ y$
pre: $payable(ticket'), funds(F), have(y,F)$
add: $have(y, ticket'), able(y,z)$
del:-

substitute (fly + buy ticket):

$S\backslash T$: $\lambda x \lambda y \lambda z.takeflight'(buy'\ z\ ticket'\ x)\ y\ x$
pre: $able(x, buy'(z, ticket', x)), flight(y), payable(ticket'), funds(F), have(x,F)$
add: $at(x, dest(y)), at(y, dest(y)), have(x, ticket'), able(x,z)$
del: $at(y, orig(y)), at(x, here)$

apply (John + fly buy ticket):

S : $\lambda y \lambda z.takeflight'(buy'\ z\ ticket'\ john')\ y\ john'$
pre: $able(john', buy'(z, ticket', john')), flight(y), payable(ticket'), funds(F), have(john', F)$
add: $at(john', dest(y)), at(y, dest(y)), have(john', ticket'), able(john', z)$
del: $at(y, orig(y)), at(john', here)$

$$(X/Y)\backslash Z : f \quad Y\backslash Z : g \rightarrow X\backslash Z : \lambda x.fx(gx) \tag{S}$$

1. Scripts can be used in various orders, leading to different states.
2. Plans are states of the derivational process.
3. We face the same lexicalized knowledge acquisition problem for plans as we do for child language acquisition.

- Abend et al. (2017); Pasula et al. (2007)

Conclusion: some suggestions

We need to explain

1. why discourse can take abstract objects but not participants,
2. why discourse structures cannot embed like clauses,
3. why clause-level emdedding requires thematic structure,
4. how recipes can give rise to different planning states with the same recipe.

Sequencing itself seems to have its own semantics.

1. The closer we look at the relation between discourse and sentences,
   - the closer it looks like the relation between plans and scripts.

Projection of argument structure, thematic structure and plan structure

to complex behavior seems to be a simple and similar computational mechanism.

Motor and sequence control areas of the brain have been taken over by language to the point of co-existence. (Deacon, 1997, 2012).

It seems possible that language piggybacked on planning.

Thanks.

Abend, O., T. Kwiatkowski, N. Smith, S. Goldwater, and M. Steedman (2017). Bootstrapping language acquisition. *Cognition 164*, 116–143.

Aktaş, B., C. Bozsahin, and D. Zeyrek (2010). Discourse relation configurations in turkish and an annotation environment. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pp. 202–206. Association for Computational Linguistics.

Asher, N. (1993). *Referring to Abstract Objects in Discourse*, Volume 50 of *Studies in Linguistics and Philosophy*. Dordrecht: Kluwer Academic Publishers.

Bozşahin, C. (2012). *Combinatory Linguistics*. Berlin: De Gruyter Mouton.

Bozşahin, C. (2017). Verbs, heads, word orders and types. In S. Özsoy (Ed.), *Word Order in Turkish*. Springer. forthcoming.

Deacon, T. W. (1997). *The Symbolic Species: The co-evolution of language and the human brain*. The Penguin Press: London.

Deacon, T. W. (2012). *Incomplete nature: How mind emerged from matter*. Norton & Company.

Demirşahin, I. (2015). *Discourse Structure in Turkish*. Ph. D. thesis, Middle East Technical Univ. (ODTÜ), Ankara.

Demirşahin, I., A. Öztürel, C. Bozşahin, and D. Zeyrek (2013). Applicative structures and immediate discourse in the turkish discourse bank. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pp. 32–69. Association for Computational Linguistics.

Fikes, R. and N. Nilsson (1971). Strips: a new approach to the application of theorem proving to problem solving. *AI Journal 2*, 189–208.

Forbes-Riley, K., B. Webber, and A. Joshi (2006). Computing discourse semantics: The predicate-argument semantics of discourse connectives in d-ltag. *Journal of Semantics 23*(1), 55–106.

Geib, C. (2015). Lexicalized reasoning. In *Proceedings of the Third Annual Conference on Advances in Cognitive Systems ACS*, pp. 1–17.

Geib, C. and R. Goldman (2009). A probabilistic plan recognition algorithm based on plan tree grammars. *Artificial Intelligence 173*, 1101–1132.

Geib, C., K. Mourão, R. Petrick, N. Pugeault, M. Steedman, N. Krueger, and F. Wörgötter (2006). Object action complexes as an interface for planning and robot control. In *Proc. of Humanoids-2006 Workshop - Towards Cognitive Humanoid Robots*.

Ghallab, M., D. Nau, and P. Traverso (2004). *Automated planning: theory and practice*. San Francisco: Morgan Kaufmann.

Ghallab, M., D. Nau, and P. Traverso (2016). *Automated Planning and Acting*. Cambridge University Press.

Halliday, M. and R. Hasan (1976). *Cohesion in English*. Longman.

Hobbs, J. R. (1985). On the coherence and structure of discourse. Technical report, Report CSLI-85-37, Center for Study of Language and Information.

Lee, A., R. Prasad, A. Joshi, N. Dinesh, and B. Webber (2006). Complexity of dependencies in discourse: Are dependencies in discourse more complex than in syntax. In *Proceedings of the 5th International Workshop on Treebanks and Linguistic Theories, Prague, Czech Republic, December*.

Lee, A., R. Prasad, A. Joshi, and B. Webber (2008). Departures from tree structures in discourse: Shared arguments in the penn discourse treebank. In *Proceedings of the Constraints in Discourse III Workshop*.

Mann, W. C. and S. A. Thompson (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text 8*(3), 243–281.

McMichael, D., S. Williams, and G. Jarrad (2006). Functional combinatory categorial grammar. *Journal of Artificial Intelligence Research 25*, 1–42.

Miller, G., E. Galanter, and K. Pribram (1960). *Plans and the Structure of Behavior*. New York, NY: Henry Holt.

Pasula, H. M., L. S. Zettlemoyer, and L. P. Kaelbling (2007). Learning symbolic models of stochastic domains. *Journal of Artificial Intelligence Research 29*, 309–352.

Prasad, R., B. Webber, and A. Joshi (2014). Reflections on the Penn discourse treebank, comparable corpora, and complementary annotation. *Computational Linguistics*.

Sacerdoti, E. D. (1975). A structure for plans and behavior. Technical report, SRI.

Schank, R. C. and R. Abelson (1977). *Scripts, plans, goals and understanding*. New Jersey: Lawrence Erlbaum.

Steedman, M. (2002). Plans, affordances, and combinatory grammar. *Linguistics and Philosophy 25*, 723–753.

Steedman, M. (2012). *Taking Scope*. Cambridge, MA: MIT Press.

Steedman, M. and C. Bozşahin (2017). *Projecting from the lexicon*. So&So Press. forthcoming.

Webber, B. (2004). D-LTAG: Extending lexicalized TAG to discourse. *Cognitive Science 28*, 751–779.

Zeyrek, D., I. Demirşahin, and C. Bozşahin (2017). Turkish discourse bank: Connectives and their configurations. In K. Oflazer and M. Saraçlar (Eds.), *Turkish Natural Language Processing*. Springer. forthcoming.

Zikánová, Š., E. Hajičová, B. Hladká, P. Jínová, J. Mírovskỳ, A. Nedoluzhko, L. Poláková, K. Rysová, M. Rysová, and J. Václ (2015). *Discourse and coherence. From the sentence structure to relations in text*. Prague: UFAL.