Dialogue as Collaborative Tree Growth

This abstract proposes an application of Dynamic Syntax (DS Kempson et al 2001) to dialogue modelling, characterising parsing, generation, and the relation between them. As evidence for this approach, we show how it provides a natural basis for modelling (i) the parsing/generation of ellipsis, (ii) such dialogue properties as a speaker-feedback mechanism, speaker/hearer alignments of structure, and speaker/hearer role reversal in shared utterances (see Pickering and Garrod 2001). The account is in five stages.

1. Background Assumptions

DS is a model of NL understanding in which parsing is defined as the progressive projection of a decorated tree structure following the left-right sequence of words in the string. Logical forms are represented as decorated trees, whose topnode is decorated with a formula $Fo(\alpha)$ of type t, and whose dominated nodes are decorated with subterms of the formula α. The central concept is that of goal-directed tree growth, defined over partial trees, driven by the imposition of requirements, eg $?Ty(t)$ constituting the overall requirement to establish a logical form of type t, with $?Ty(e)$, $?Ty(e \rightarrow t)$, etc as subgoals. Tree nodes from the root downwards are created with imposed *requirements* which are subsequently *annotated* by formulae and other labels as provided by incoming lexical items and processes such as function application defined in tandem with type-deduction. (At each stage, there is one itemised node under development, indicated by a pointer, $\diamondsuit$.) ALL update processes are monotonic, progressively developing a tree structure meeting the requirements which are imposed on nodes as they are introduced.

In addition to the concept of requirement are other concepts of structural underspecification, such as the introduction of underspecified formulae (for anaphora), and underspecified tree relations (for left-dislocated expressions), each of which has to be subsequently updated. These various forms of underspecification interact in the process of progressive satisfaction of all imposed requirements through computational, lexical and pragmatic actions, each of which constitutes a monotonic step of tree growth. Wellformedness is defined in terms of the result of such actions: a sentence is wellformed if and only if at least one completed tree structure can be derived from a sequence of words, with no requirements outstanding.

2. Dynamic Syntax and Generation

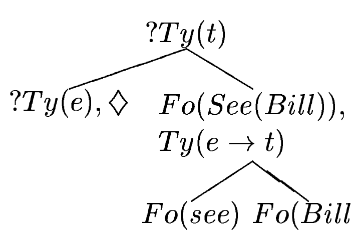
The second step is to articulate the relation between parsing and generation. We define a "tactical" generation system which defines pairs of a source tree and a progressively enriched partial tree in which the pointer decorates a node whose analogue in the source tree is being "checked", an action which is matched by the selection of some word from the lexicon and "writing" it at the right-hand edge of a sequence of already established words. Such checking action is licensed if and only if the word selected projects a compound parse action which can (in conjunction with a sequence of computational actions) yield an update of the parse tree from that defined over the words already linearised onto an update reflecting the annotations on the node being checked, all such subtrees being required to constitute a sequence of partial trees from start state to the source tree. For example, consider a simple tree structure such as projected from *A woman saw John.* We assume in generation that we have on the one hand a full tree as source representing interpretation of the string, and a parse tree made up of a single rootnode with pointer and requirement $?Ty(t)$, this the starting point for any parse sequence. Within the source tree, the generating system can then "check off" the determiner node, and hence "generate" the first word in the string, just because by a combination of node-introducing rules operating on the rootnode together with the lexical actions of the indefinite determiner a, the node annotated as $Fo(\epsilon, x, P), Ty(cn \rightarrow e)$ could be successfully so annotated as a daughter of a subject node in the emergent parse tree. The search, then, through the lexicon is to find this word.[1] With the first word generated, the generator attempts to check off the nominal node (because this is where the lexical action of parsing a determiner will leave the pointer), by scanning the lexicon to pick out the set of lexical actions that successfully project the required remaining nodes of this subtree (the lexical item *woman* is defined to project both a node annotated by a variable and a node annotated by the predicate $Woman$). Once the word is identified, the variable and $Ty(e \rightarrow cn)$ nodes are taken to be checked, thereby licensing the attachment of the word *woman* to the righthand edge of the sequence. The partial tree which reflects the result of the control parsing sequence, is duly updated to become a partial tree with a subject node decorated with $Fo(\epsilon, x, Woman(x))$. Having checked off the family of terminal nodes associated with the subject in the source tree in virtue of a successful sequence of steps in the parse tree annotating the subject node in that tree, the pointer returns to the rootnode of the parse tree, where a following computational step may introduce the predicate node. Accordingly, the generator attempts to check the content of its predicate node in the source tree. With verbs defined as being of the form $IF\ ?Ty(e \rightarrow t), THEN...$, the lexicon is scanned for a word which will lead to the introduction and decoration of a node with $Fo(see), Ty(e \rightarrow (e \rightarrow t)$. The word *saw* is duly selected, and written at the right edge of the sequence *A woman saw*, and that node is taken to be "checked". Given that the actions of the word *saw* (ignoring tense) leave the pointer in the parse tree at the object node, that node alone remains to be realised by some sequence of words. The pattern is quite general: generation involves the construction of paired trees, one a fully annotated tree which forms the input to the process during which its nodes are progressively checked, the second a tree which reflects the corresponding parse tree commensurate with establishing the node currently being checked. The provided sequence of actions is by no means unique: any sequence of actions conforming to the pattern of pairing source tree incrementally with emergent parse tree following the sequence of words is licensed: eg left-peripheral placement of any "dislocated" NP presuming on a parse-sequence projecting an initially unfixed node.

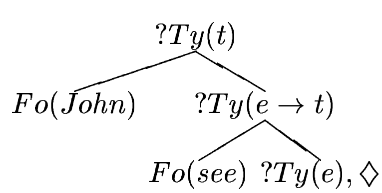
3. The Generation-Ellipsis Parallelism

The next step in the argument is to reduce the checking process to an independently defined process, the abstraction process used in ellipsis construal. For example, we take the parsing of the elliptical fragment in (1), which can be interpreted either as "Harry saw Bill" or as "John saw Harry":

(1) John saw Bill. Harry too.

For this fragment, a subtree needs to be constructed from the tree established from the first sentence as the basis for interpreting the fragment *Harry*. This subtree is (1a) or (1b):

(1a)


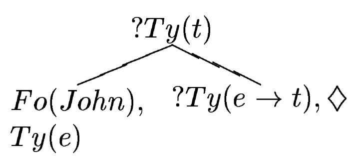
(1b)


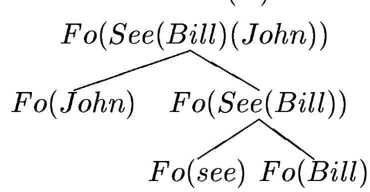
We need a process which takes some decorated tree as input, and replaces some annotated

[1] In this paper, we make the Fodorian assumption that words and concepts correspond one-to-one, trivialising this aspect of lexical search. All NPs project nodes of type e, adopting the epsilon calculus as the language of Fo values – see Meyer-Viol 1995. This paper has also nothing to say about phonological/phonetic levels of realisation.

node(s) of that tree with requirements, "abstracting" out some established value and in so doing providing a "newly incomplete" tree with which to establish some different completion (notice how such reversal along the process of tree growth at a terminal node automatically applies to all nodes dominating that terminal node). Accordingly we define a general process of tree abstraction back along any dimension of tree growth defined over pointed partial trees. With such a process as background, we define ellipsis construal as the parsing of the expression relative to a partial tree as context yielding a propositional formula as value. This process of abstraction and subsequent processing of the fragment yields as a result the parallelism of construal between antecedent structure and interpretation of the fragment characteristic of ellipsis. (See Dalrymple et al 1991, etc).[2]

The partial trees defined by this abstraction process need not be those constructed in the course of parsing [eg (1a) is not an intermediate structure in parsing an English string]. However, such trees DO, notably, correspond to intermediate steps of generation, IF we model the generation process of "checking" a node as a process of moving back up the tree from source tree to the original goal, representing the checking of nodes as an emptying, replacing the annotation by $?Ty(X)$, for suitable type - eg (1a) is the form of tree established, when a subject expression is the first constituent in a structure to be so "checked". Accordingly we define the checking process as licensing a replacement of values by some partial tree back along the process of tree growth eg replacing annotations with requirements. The steps of abstraction licensed in generation are constrained to be those for which there is a corresponding parse step from the partial tree associated with a previous generation step to that resulting from the sequence of actions associated with the word selected for linearisation. The effect of the constraint is that at each intermediate step, the pair of the progressively abstracted partial tree and the progressively enriched parse tree must unify to yield back the source tree. So the linearisation of the word *John* in uttering (1) and the abstraction from source tree to (1a) is licensed, because there is a well-defined parse step from a tree with single node $?Ty(t)$ to the tree (1a'), (1a) and (1a') unifying to provide all that is needed to establish the source tree (1'):

(1a') 

(1') 

We can now define the relation between generation and parsing while maintaining their distinctiveness (Levelt 1993). Parsing is the development of tree growth from start state ($?Ty(t)$) across intermediate pointed partial trees (T_i) to satisfaction of some completed goal (using Computational, Lexical and Pragmatic actions involving a string $w_1, ...w_n$), with all subgoals emptied:

$$\left[\begin{array}{lllll} START(?Ty(t)) & \longmapsto^{\{C,L,P\}} & T_i & \longmapsto^{\{C,L,P\}} & GOAL(Fo(\psi), Ty(t)) \\ \hline String(w_1) & & \langle w_1 \ldots w_i \rangle & & \langle w_1 \ldots w_n \rangle \end{array}\right]$$

Generation is the reverse shift from some completed tree as the source tree, with progressive emptying (tree abstraction) as each word is selected until a tree skeleton is reached, with all nodes decorated only with requirements. However, generation is not simply a sequence of parsing actions in reverse. Rather there is an interdependence between them – the success of abstraction

[2] See Kempson et al 1998 for arguments that this abstraction process is essentially structural.

relies on the parsability of the linearised output:

$$\left[\begin{array}{lccccc} \textit{Abstraction}: & Source(Fo(\phi), Ty(t)) & \longmapsto^{\{ABSTR\}} & T_i & \longmapsto^{\{ABSTR\}} & START(?Ty(t)) \\ \textit{Parsing}: & START(?Ty(t)) & \longmapsto^{\{C,L,P\}} & T_i{}' & \longmapsto^{\{C,L,P\}} & Source(Fo(\phi), Ty(t)) \\ \hline \textit{String}: & 0 & & \langle \ldots w_i \rangle & & \langle \ldots w_i \ldots w_n \rangle \end{array}\right]$$

So the difference between parsing and generation lies in the metalevel aspect to generation. Each step in generating a string with interpretation $Fo(\phi)$ represented as a tree structure T involves correlating some proposed abstracted tree (T_i) and some corresponding parse tree ($T_i{}'$) which must unify to yield back the source tree T with $Fo(\phi), Ty(t)$ at its topnode. The tightness of this parsing-generation correspondence is unique to DS, for in this system there is no intermediate system of constraints: the parsing architecture is the grammar formalism, syntactic structure is no more than the progressive construction of semantic representation, and the generation system is to be defined relative to this tree-growth process.

4. Dialogue and the generation of ellipsis.

This tightness of correspondence between generation and parsing is well-suited to an account of dialogue, satisfying the desiderata for successful dialogue modelling set out in Pickering and Garrod 2001. First the model by definition anticipates maximal alignment of syntactic/semantic/situational levels by eliminating levels other than that of semantic representation.[3] Secondly, the account yields an incremental self-monotoring device without separate articulation, as the checking of each generation step against the grammar formalism is a self-monitoring parsing step. We expect immediate production breakdown with parsing feedback delay (contra Levelt 1993: MacKay 1987). The account also leads us to expect ellipsis to play a large role in dialogue, with generation and construal of ellipsis able to rely on the rich concept of structural context relative to which ellipsis construal is defined.[4] By the proposed pattern of generation, utterance of elliptical fragments is licensed because by presuming on the definition of ellipsis construal as a process of abstraction, the utterance of a fragment may be all that is required to enable a hearer to construct a second propositional formula, given but a single step of tree abstraction and consequent compilation of a new tree:

(2) A: I bought a scooter for Bill. A skateboard too.

For example, in (2), relative to the context provided by the tree resulting from processing sentence *I bought a scooter for Bill*, all that is required to allow the interpretation of the fragment *a skateboard* is a process which takes out the decoration $Fo(\epsilon x, Scooter(x))$ from the tree provided by the first sentence, and replaces it within that tree by the requirement $?Ty(e)$ and pointer at the object node, consequently replacing the predicate and root nodes with the requirements $?Ty(e \to t)$ and $?Ty(t)$ respectively. But all that is then required in generation is to utter a fragment such as *a skateboard*, for with this utterance, a second propositional formula can be compiled by the hearer.

5. Collaborative production/interpretation

As a final testcase, we show how the account can model role-reversals in collaborative utterances:

---

[3]Despite no separate syntactic representational level, we anticipate alignments such as repetition of double object constructions:
(i) A: "I bought Ellie a scooter." B: "I bought her a skateboard."
Lexical specifications involve projection of concept plus a sequence of tree-update actions, repetition of a word ensuring identity of both concept and sequence of actions with the first occurrence. With verbs such as *buy*, which have discrete sequences of actions for the two processing strategies, repetition will ensure recovery not only of the same concept but of the same actions. Hence 'syntactic' as well as 'semantic' alignment.

[4]The account requires anaphora/ellipsis resolution at the level of tactical generation (contra Dale 1992).

(3) A: What shall I buy for ...
B: Bill?

In linearising the partial string, A's generation task starts from a source tree with a metavariable WH as the object argument whose value is to be requested and some constant denoting the person to be bought a present. A initiates the string with the *wh* expression on the grounds that B has a parse strategy for introducing an unfixed node, decorating it with the lexically projected WH metavariable, and merging it later with some appropriate fixed position. B, parsing, duly constructs such a partial tree, updating it by merging the unfixed node with the object node introduced in parsing the verb *buy*. This leaves one further node in the parse tree needing completion to meet the verb's requirement of an indirect object. At the juncture of uttering/parsing the indirect object expression, this partial tree is shared by A and B. With A signalling name loss, B provides it from independent knowledge, taking over the generation task, completing the pair of emptied tree (associated with the linearised string) and the completed tree (as constructed interpretation). A, now parsing, processes the name *Bill* to complete the partial parse tree, hence also inducing a completed pair of trees, one emptied, one with constructed interpretation. The construction of a tree in parsing some string, and the construction of the same tree in producing that string as a constraint on the generation steps licensed ensures the naturalness of the shift from one activity to the other.

Overall, the characterisation of language production as a metalevel activity making essential reference to the parse steps, both defined in terms of growth of semantic representation, provides the basis for a natural account of dialogue with no stipulation of actions other than those indepedently required in NL processing.

References

Dale, R. 1992. *Generating Referring Expressions.* MIT Press.

Dalrymple, M et al. 1991. Ellipsis and higher-order unification. *Linguistics and Philosophy 14*, 399-452.

Kempson, R. et al. 1998. VP Ellipsis: towards a dynamic, structural account. In Lappin and Benmamoun (eds.) *Fragments*, 227-290. OUP.

Kempson, R. et al. 2001. *Dynamic Syntax: The Flow of Language Understanding.* Blackwell.

Levelt, W. 1989. *Speaking: From Intention to Articulation.* MIT Press.

Mackay, D. 1987. *The Organization of Perception and actions: A theory for Language and Other cognitive Skills.* New York: Springer.

Meyer-Viol, W. 1995. *Instantial Logic.* Ph.D Utrecht.

Pickering, & Garrod, 2001. Toward a mechanistic psychology of dialogue. ms .