Unmediated Data-

Oriented Generation

Dave Cochran, M.A.

Thesis submitted in fulfilment of the requirements of the degree of Master of Science

in Developmental Linguistics

Supervisor: Simon Kirby

25th August 2006

Contents

Abstract

This thesis describes the development of a system of Data-Oriented Generation (DOG) wherein noun-phrases are produced as descriptions of simple visual stimuli. This is work towards a broader goal of developing a psycholinguistically realistic Data-Oriented theory of Sentence Generation. Technologically, this is timely because, after sixteen years of research into Data-Oriented Parsing (DOP; the formalism was first proposed by Scha (1990), first implemented by Bod (1992) and has been further developed , for example, by Bod, (1998, 2003, 2006b), Bod, Bonnema and Scha (1996), Bod and Kaplan (1998) Goodman (2003), Hoogweg (2000), no-one has yet produced a system for Data-Oriented Generation.

Rather than use a logic-like formalism to encode meaning, the model of generation proposed operates by directly coupling linguistic exemplars with exemplars in other modalities – vision, in the present case, though, it is hoped that the model could be extended into other meaning-providing modalities.

<u>*Acknowledgements*</u>

I would like to extend my warm thanks to all my friends, colleagues and teachers in Linguistics and English Language (and some in Philosophy and Informatics) at the University of Edinburgh, who together have conspired to create a the most nourishing, challenging and exciting intellectual environment I could possibly have hoped for.

Simon Kirby, my supervisor, without whose insightful questioning, the concepts behind UDOG would never have become clear enough to implement, and who (bravely) took the project on despite its complexity and my inexperience as a programmer.

Thanks to Nigel Goddard, Steve Renals, Ewan Klein and Philip Koehn of the School of Informatics, who taught me to programme over the course of this year. Prior to their teaching, my only programming experience had been two separate incidents of copying a "Hello World" programme out of a book and having it not work.

Thanks to Rens Bod, Jim Hurford, Sarah Fisher, Hannah Cornish, Jeff Ketland, Antonella Sorace, Simon Levy, and countless others who I would have mentioned if I didn't right now have half an hour to get this thing printed, bound and handed in, for bright ideas, wit, wisdom and genial company over the last two years.

Finally, but by no means least, thanks to my wife Morrigan and daughter Coire for supporting me and inspiring me in so many ways, and for suffering through my long absences as I toiled away in the Microlab working on my dissertation.

Y'guys all rule. What can I say?

Chapter 1

Introduction[1]

This thesis describes the development of a system of Data-Oriented Generation (DOG) wherein noun-phrases are produced as descriptions of simple visual stimuli. This is work towards a broader goal of developing a psycholinguistically realistic Data-Oriented theory of Sentence Generation. Technologically, this is timely because, after sixteen years of research into Data-Oriented Parsing (DOP; the formalism was first proposed by Scha (1990), first implemented by Bod (1992) and has been further developed , for example, by Bod, (1998, 2003, 2006b), Bod, Bonnema and Scha (1996), Bod and Kaplan (1998) Goodman (2003), Hoogweg (2000), no-one has yet produced a system for Data-Oriented Generation. However, the emphasis of the proposed research is cognitive rather than technological, as most DOP research to date has been. The particular model under scrutiny, Unmediated Data-Oriented Generation (UDOG), is inspired by the question; "Can one have a perceptually grounded linguistic semantics without a 'Language of Thought' (LOT) to intervene between perception and utterance?" Or, put more broadly, "What is the input to sentence generation?" As such, the system proposed (First mooted in Cochran 2005) operates by means of direct connections between concrete exemplars of past experience in visual and linguistic modalities.

Typically, while psycholinguists studying language production have assumed the existence of pre-linguistic messages providing the input for language production, philosophers have had grave misgivings about this. Typical of this latter trend is

---

[1] Much of the material in this introduction is adapted from Cochran (2004), a paper submitted for the MSc course, Language Production. Please note that although this paper was awarded a mark, I dropped out of the degree programme towards which this mark would have counted, prior to completion; therefore, this material has not counted, and will not ever count, towards the assessment of any degree except in its present form as part of the present submission.

Simon Blackburn's argument (1984, p40-67) that the LOT Hypothesis is a "dog-legged theory", insofar as our ability to understand and perform operations with pre-linguistic messages stands in want of explanation in precisely the same way as our ability to understand expressions in language, thus inviting an infinite regress. The argument at the heart of all this is Wittgenstein's (1969) example of understanding the meaning of the word "red", in which it is asserted that, if one supposes the mind to contain an image of "red" to provide the word "red" with meaning, it is functionally equivalent to having an image of "red" *outside* the mind to correlate word to meaning, say, a labelled card with a red patch painted on it.

> When I hear the order "fetch me [a red flower from that meadow]." I draw my finger across the chart from the word "red" to a certain square, and I go and look for a flower which has the same colour as the square. … [But] consider the order "imagine a red patch". You are not tempted in this case to think that before obeying you must have imagined a red patch to serve you as a pattern for the red patch which you were ordered to imagine.
>
> Wittgenstein 1969, p3.

In contrast, psycholinguists have, without much discussion, tended to favour the assumption of pre-linguistic messages as a way of abstracting their desired object of investigation, the complex of systems by which we select the words, inflections, syntactic structures, phonemes and suchlike with which our desired meaning is to be expressed away from a matter which is murkier, more daunting, and less accessible to experimental research, that of how, in the first place, we decide which meanings we want to express. Levelt (1989), for instance, divides his "blueprint for the speaker" (p.9) first of all into the "Conceptualizer" and the "Formulator". However, it is quickly apparent that while the Formulator is a fairly clearly defined set of linked subsystems for handling different layers of the surface structure of linguistic expressions, the Conceptualizer, by Levelt's own admission, is a sort of heterogeneous "not-the-liver" category[2] set up to do everything the Formulator doesn't.

---

[2] I am indebted to Bedford (1997) for this singularly useful expression. It refers to a particular kind of fallacious category in cognitive science – one by which the discovery of a genuine category or "organ" within cognition is supplemented by the putative discovery of a second category comprising everything that the first doesn't.

The sum total of these mental activities will be called *conceptualising*, and the subserving processing system will be called the *Conceptualizer* (in full awareness that this is a reification in need of further explanation – we are, of course, dealing with a highly open-ended system involving quite heterogeneous aspects of the speaker as an acting person)

<div align="right">p.9, author's emphasis.</div>

One finds in Levelt's further exposition of the Conceptualizer (p70-106) that the output of "messages" which it feeds into the Formulator must meet certain criteria - they must be "propositional" (in a broad sense) (p.72-96), they must have perspective (i.e. carry information about topicality, news value, etc) (p96-100) and mood (eg. interrogative, declarative, etc) (p100-103), and be marked for whatever supplementary information the grammar of the language in question demands (eg. information about evidentiality is optional in English, but mandatory in Karaja (Maia, 2000)). Levelt laments the absence of a "message grammar", or any immediate prospect of one (p70). At this point, the "message" seems so much like language itself, that one or the other must surely be redundant.

The strength of "Language of Thought" based approaches to meaning, and to question of the "input to language production" has been largely a consequence of the fact that any more holistic alternative seemed simply impossible to model, and therefore impossible to consider scientifically. What follows is a pilot for a model that, I hope, may give a holistic alternative a chance to be properly scientifically evaluated.

Chapter 2

Background

2.1 Data-Oriented Parsing

The fundamental underlying framework for the present study is that of Data-Oriented Parsing, a model of Statistical Natural Language Processing first proposed by Scha (1990), and first implemented by Bod (1992).

This summary adapts material from Cochran (2005) and Chen, Cochran, Hanafusa, Laskowski, Ludke, and Ntarila[3] (2005). The simplest manifestation of STSG is DOP1, as described in Bod (1998 p12-23 and 40-50), though more sophisticated versions exist. The parser uses a large parsed corpus[4] divided into a training corpus and a smaller corpus against which the parser is tested. The parser breaks every tree in the training corpus down into all its possible subtrees, according to the wellformedness rules below.

1) Every subtree must be of at least depth 1.
2) Every connection must have a node on either end
3) Sister relationships must be preserved

---

[3] Note that the latter citation is of work elsewhere assessed for the present qualification.
[4] Such as the the Penn Treebanks (in English, Chinese, Arabic, etc) or the "Developing a Morphologically and Syntactically Annotated Treebank Corpus For Turkish" Project sponsored by the METU Informatics Institute & Sabanci University
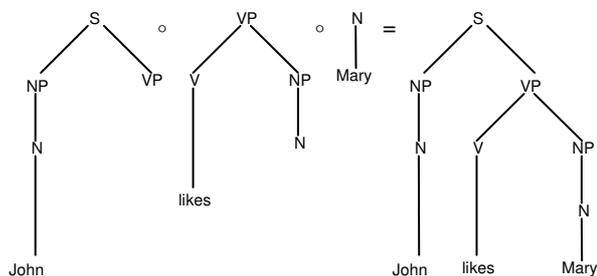
Figure 2.1: A derivation of "John likes Mary". "∘" is the operator for the tree-substitution operation.

The parser is given test corpus strings and builds up new parse-trees for these using the fragments available to it from the training corpus, starting with a fragment with an S-node at the top, and then, for each nonterminal leaf-node, working rightwards, substituting in additional subtrees, the topmost node of which must carry the same label as the node to be substituted. (see figure 2.1).

In DOP research it is necessary to distinguish between *parses* and *derivations*. A parse is the tree structure expressed over a string; a derivation is the particular sequence of subtree substitutions by which it was constructed. When parsing with probabilistic context-free grammars (PCFG's, see Manning and Schütze (1999, pp.381-405); note that a PCFG is equivalent to a DOP grammar in which subtree depth has been restricted to 1), there is a one-to-one mapping between parses and derivations, because all non-terminal nodes (nodes which have daughters in the completed parse and do not contain concrete representations of utterable content – words, morphemes, etc) *must* be substitution sites. In DOP, subtrees can be of any depth, and so in any given derivation, any subset of the non-terminal nodes could have been substitution sites, while the remainder will not have been. As such, if a parse contains N many non-terminal nodes, it will have 2^N many derivations.

For each subtree substitution t, its probability $P(t)$ is calculated as its total frequency of occurrence $|t|$ in the training corpus over the summed corpus frequency of subtrees with the same root node; [5]

---

[5] Note that although, beside the node-label on the substitution site, the input to be parsed is also a constraint on the selection of subtrees for substitutions, these constraints are not factored in to the calculation of probabilities.

$$P(t) = \frac{|t|}{\sum_{\{t':r(t')=r(t)\}} |t'|}$$
(2.1)

…where $r(t)$ and $r(t')$ are the node-labels on the root-nodes of subtrees t and t'.

The probability of a derivation is the product of the probabilities of its subtrees (note that $\circ$ is the notation for the substitution operation; thus $t_1 \circ ... \circ t_n$ is the sequence of substitutions, which together comprise the derivation);

$$P(t_1 \circ ... \circ t_n) = \prod_i P(t_i)$$
(2.2)

And the probability of a parse T is the sum of the probabilities of its possible derivations D;

$$P(T) = \sum_{\{D:\,D\text{ derives }T\}} P(D)$$
(2.3)

The output of the parser is, in theory, the most probable parse. In practice, there are issues of computational complexity that prevent this from being calculated directly; but these will be addressed in §4.2 below.

Bod (ibid p.54) reports accuracies of 85% on the ATIS[6] corpus for DOP1.

2.2 Data Oriented Semantics

Although the present work reports the first model of Data-Oriented Generation, it is not the first attempt to incorporate representations of meaning into a Data-Oriented model; van den Berg, Bod and Scha (1994) and Bod Bonemma and Scha (1996) report two models of Data-Oriented Semantics in which trees from the Penn Treebank were extended with predicate logic-like annotations on the non-terminal nodes. To give two instances of how this may be done, in the toy corpus illustrated in figure 2.2.a, expressions like ∃x(MANx & WHISTLESx) are located at the root node, and broken down with lambda-abstractions as you work down to the

---

[6] Air Transport Information System – part of the Penn Treebank.

terminal nodes (van den Berg, Bod and Scha 1994, cited in Bod Bonemma and Scha 1996), whereas in figure 2.2.b, the expressions above the immediate parents of the terminal nodes are replaced with more abstract substitution-schemas (Bod, Bonnema and Scha 1996).
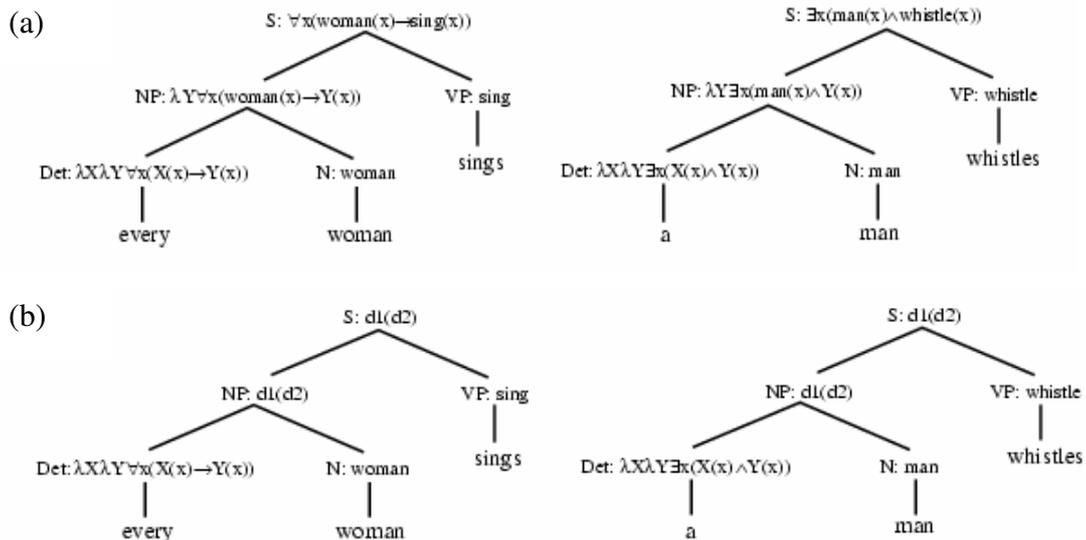

Figure 2.2: Two toy corpora from; Bod, Bonnema and Scha (1996). The authors note that constraining DOP to process semantic annotations of the type shown in figure 2.2(b) actually improves the parser's accuracy for syntax, and its overall processing speed.

Logic-based representations of meaning are a relatively cheap way of representing semantics in an NLP programme; specifically, they do not in any way require models of the diverse cognitive modalities in which the meanings are grounded to be build into the model of language. The novel models that I will describe in the next chapter, by contract, operate by directly coupling linguistic exemplars with exemplars in other modalities – vision, in the present case, though, as will be seen, it is hoped that the model could be extended into other meaning-providing modalities. It is furthermore hoped that models incorporating such direct couplings will prove much more powerful, not only in understanding linguistic meaning, but more generally the interactions between language and the other subsystems of human cognition.

2.3 SHRDLU

One of the earliest successes in Artificial Intelligence and Natural Language Understanding was Winograd's (1972) SHRDLU programme. SHRDLU was a dialogue system, capable of interacting with a human user (via teletype) with relation to an extremely limited simulated microworld, consisting of blocks of different colours and shapes which it was capable of manipulating at the request of the user, and answering queries about them. The BLOCKS World which SHRDLU inhabited had its own basic physics (for instance, other blocks can be stacked on a cube, but not on a pyramid), and SHRDLU had an understanding of those physics which was capable of influencing its dialogue. SHRDLU was a collection of interacting programmes in which explicit procedural knowledge of the BLOCKS world, and the syntax and semantics of its 200 word vocabulary, was hand-coded. SHRDLU's proved to be extremely impressive for its time; it could conduct perform inferences about blocks, learn vocabulary defined in terms of previous vocabulary, and conduct sensible, natural sounding dialogue with reference to its micro-world. Ultimately, however, the hand-coding of explicit knowledge proved to be a dead-end approach, as it proved to be highly brittle, domain specific and not scaleable. Dreyfus's (1997) classic criticism of SHRDLU (cited in Clark 1991, pp. 25-27) follows argues against the hypothesis that microworld based SHRDLU-style AI could ever "scale up" from microworlds to anything comparable to the world as experienced by humans; Dreyfus draws on an example from and MIT internal memo circulated by Minsky and Papert (1970), in which they consider the domains of knowledge from which a child must draw to be able to understand the following sentence in conversational context;

> Janet: That isn't a very good ball you have. Give it to me and I'll give you my lollipop.

> …We conjecture that, eventually, the required micro-theories can be made reasonably compact and easily stated … once we have found an adequate set of structural primitives for them.
>
> (Minsky and Papert 1970, p. 48 & p.50, cited in Dreyfus 1997, p.147)

Dreyfus comments that this assumption of Minsky and Papert's is ultimately untenable, because there in fact is no micro-theory for such a conversation separable from the rest of human experience and meaning. This, Dreyfus (ibid) casts as a general failure of the "micro-worlds paradigm";

> …it … is likewise misleading to call a set of facts and procedures concerning blocks a *world* when what is really at stake is the understanding of what a world is. A set of interrelated facts may constitute a *universe*, a domain, a group, etc., but it does not constitute a *world*, for a world is an organised body of objects, purposes, skills and practices in terms of which human activities have meaning … one cannot equate … a program that deals with a "tiny bit of the world" with one that deals with a "mini-world".
>
> (pp.150-1, author's emphasis)

Dreyfus's criticism sticks because of the ways in which our diverse cognitive competencies *saturate* one another, what Clark (1991, p.25) calls the "thickness" of our concepts, creating an explosively complex manifold of dependencies between different knowledges and modalities of knowledge that simply prohibits their direct modelling as explicit declarative knowledge; for a knowledge-representing formalism to stand as a "theory of content" that can be applied to *worlds* in Dreyfus's sense, it must be *scaleable* and *robust*. The "atomism" underpinning Winograd, Minsky and Papert's approaches fails that test.

However, in many subfields of AI research, statistical, experience-based approaches, have, by comparison, proven to be highly robust and scaleable; for example, in Natural Language Processing (Manning and Schütze 1999), Vision Science (Kersten 2000) and Robotics (Thrun 2005); A fact which gives considerable weight to Brooks' (Brooks 1997) suggestion that explicit knowledge representations are simply the wrong sort of abstraction to be working with when trying to model cognition. It is my hope that the method of directly coupling exemplars across modalities of cognition provides a direction for an integrated, multimodal approach to probabilistic AI and cognitive modelling.

2.4 L_0

Another precedent worthy of note is the L_0 Miniature Language Acquisition task proposed in Feldman, Lakoff, Stolcke and Weber (1990). The task was intended to integrate three domains of Cognitive Science research; vision, judgement and language; to design an algorithm which;

- Could be trained using a corpus of simple images, consisting of 3-4 simple geometric figures (circles, triangles, squares, etc) paired with one or more true statements about these scenes in some arbitrary natural language
- Which after training could be presented with novel images and judge whether novel statements about these images were true or not.
- Which would be robust to being tested on many different languages, including non-Indo-European languages.

The present project is of a similar kind differs from L_0 in a few aspects; firstly, the system is trained only on English. This is merely a consequence of limitations of time, and should be remedied in due course. Likewise the fact that in the present work, the images used are only one dimensional, is merely a consequence of limited time; a proposal for a more sophisticated Data-Oriented Picture Parser, capable of handling 2D inputs, can be found in Cochran (2006a). However, my own interest lies more in the direction of modelling language production, rather than truth-judgements, and UDOG work will probably continue in that direction. This is not least because I hope to be able to integrate UDOG into Iterated Learning Simulations (Kirby 1999, Hurford 2000, Briscoe 2002); a model of social transmission of (linguistic) knowledge wherein generations of agents are taught a toy linguistic task, with the first generation being trained on a random language, and each subsequent generation being trained on the productions of the previous – notably, via a bottleneck, whereby no generation is trained on the whole language, but must generalise from their training input in order to be able to handle novel stimuli. These models have proved to be of

considerable value in explaining various language universals in terms of the dynamics of cultural learning.

Feldman, Lakoff, Bailey, Narayanan, Regier and Stolcke (1996) describe a candidate system for the "touchstone" of the L_0 task, which combines language learning using a Probabilistic Context Free Grammar with visual learning using Artificial Neural Networks. Notably, their system is highly modular, employing heterogeneous architectures for different subtasks. In contrast, the UDOG approach is designed around a single, integrated system and a shared Data-Oriented architecture, which it is hoped will be able to scale up to integrating further cognitive modalities.

2.5 The Ubiquity of Trees

One notable feature of DOP-research is that the paradigm is not limited to language; Bod presents successful DOP models for music, trained on the Essen Folksong Collection (2002, 2005) and for equational reasoning in physics, using a corpus collected from undergraduate physics students at the University of Amsterdam (2004, 2005). Coleman and Pierrehumbert (1997) successfully use a DOP-like model to predict English-speakers' judgements of the phonotactic well-formedness of nonsense words. Cochran (2005) suggested that DOP might also be applied to motor memory. Tu, Chen, Yuille and Zhu (2005) propose that algorithms for parsing language into tree-structures may be adapted to vision, and indeed describe a Markov-Chain-based algorithm which does just that. In considering the role of trees-structures in cognition, it is helpful to set aside the visual "tree" metaphor, and consider them purely as data-structures; specifically, "trees" constitute nested mappings of higher to lower level patterns of information. Given the diversity of cognitive modalities in which the merit tree-structural analyses has been shown, Bod's (2005) characterisation of tree-structures as ubiquitous to cognition seems to be a sensible working hypothesis for an integrative vision of Cognitive Science, but it can't be the complete picture. The "nested mappings" referred to above seem to work well for mapping *intra*-modal relations in cognition, but alone they offer no mechanism for understanding the "saturation" of real human cognition with connections and correlation across modalities, to which Dreyfus (1997) alludes (see §2.3 above); beyond being a hypothesis with regard to language production and linguistic meaning, the model of

cross-modal connections between nodes in tree-structures described in the next chapter is the first intimation of a broader hypothesis regarding the informational basis of the saturatedness of human perception and cognition.

Chapter 3

The Model

As noted in the introduction above, over the course of 16 years of research into Data-Oriented approaches language, no-one has yet published a model for Data-Oriented Generation; what follows in the current chapter is a description of two pilot-models intended as a first attempt at making good that deficit.

3.1 UDOG

The "Unmediated" in Unmediated Data-Oriented Generation signifies the absence of a logic-like code or "Language of Thought" to pass messages between subsystems of cognition; in the case of the task at hand, language and vision. Rather, what is proposed is a set of direct crossmodal couplings between particular exemplars in the signifier-providing system (language) and signified-providing system (vision). This is illustrated in an extremely simplified form in figure 3.1
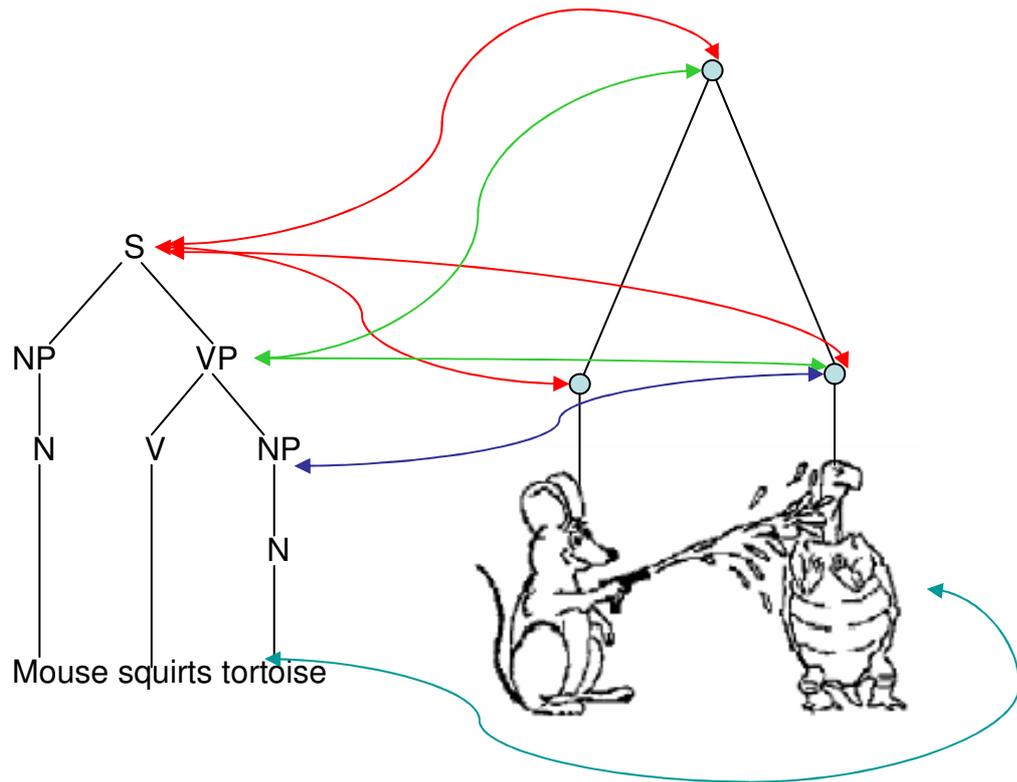
Figure 3.1: paired parse trees over visual and verbal content, with crossmodal connections. Note that some crossmodals are omitted for clarity. The crossmodals are coloured only for clarity; the colours do not signify anything.

Note that this illustration is considerably simplified on a number of counts, however; the picture-parse tree for a naturalistic two-dimensional image such as the one shown would be much more complex, and for clarity only a subset of the cross-modal links for that pair of trees has been shown. Furthermore, in human cognition, the perception of an image like the one shown would not be describable by a surface tree alone – rather the perception of the image would be modified by a perception that the static image in fact represents a fragment of a time-sequence, that the patterns of black pixels on while represent intentional agents, that the squirting is an intentional act, and that certain details of the arrangement of pixels in the mouse and tortoise's faces informs us of their affective states.

Two terms of art in the preceding description stand out as being in want of further definition; "picture parse" and "crossmodal". It is worth dwelling on these for a moment before proceeding, as they are crucial to the following model.

In order to understand what is meant by a picture parse, I refer the reader back to my comments in §2.5, on "the ubiquity of trees", to the effect that the nested information-structures represented by trees provide a highly general data-structure in cognition, whereby lower-level segmentations of cognitive content may be mapped onto higher-level ones; in vision, this comprises the relation of lowest-level feature recognition (Hubel and Weisel, 1963, 1965) to the identification of objects and assemblies of objects. This nested segmentation of the visual field has proven important in the field of computational vision science, where it has been found to be indispensable in the field of automated visual analysis; Tu, Chen, Yuille and Zhu (2005), for example, present an algorithm which uses Markov Chains to produce analyses if images "into their constituent visual patterns … in a spirit similar to parsing speech and natural language" (p.113). Von der Heydt (2004) reviews a considerable body of evidence for extra-striate areas of the visual cortex performing "intermediate processing" operations, which he characterises as "image parsing … which appears as a mediator between local feature-representations … and the processes of attentional selection and object recognition" (p.1139). However, the level of sophistication in modelling vision indicated in figure 3.1 above is simply beyond the scope of the present project (though, as noted above in §2.4, see Cochran 2006a for a proposal for a Data-Oriented Picture Parser). For the purposes of the present model, a rather simpler visual analysis is required; limited to a maximum of three layers of nested structure at most are used (as illustrated in figure 3.2 below); between the bottom layer of primitive objects (lines, dashes and dots) and the top layer, corresponding to the whole image, one mediating layer wherein primitive objects of the same type may be grouped into clusters of two or three *may* be found. A tree-structure, therefore, seems natural and cognitively plausible to join up these layers of nested structure. As a further simplification, the primitive objects in the image will be arranged one-dimensionally, so that it can be parsed with a standard DOP1 parser.

Broadly speaking, crossmodals are connections between individual nodes in the tree-representations of previously experienced cognitive structures from grounded in different cognitive modalities. In the case of the current model, the only modalities we are concerned with are vision and language, but in theory the integrated action of any grouping of cognitive modalities may be represented and mediated by crossmodals, at least provided the information in both modalities is organised under

tree-structures. In a sense, they perform a dual function; on the one hand, they mediate crude associations between pairs of trees associated in memory (in the case of the current models, between the parses of images and their descriptions); that is to say, they allow for a record to be kept of what trees were created at the same time, allowing cognitive systems (in the case of the present model, language production) to exploit statistical patterns and regularities regarding in the co-occurrence of pattern across modalities; patterns like "at times when verbal trees containing the morpheme 'cat' are processed or produced, it is more likely than it is at other times that the patterns associated with the presence of cats will be found in the visual field". This of course, only requires associations between whole trees, rather than particular nodes; thus the other function of crossmodals is to constrain more exactly which subtrees may permissibly be linked in a bimodal subtree (the term "bimodal subtree" means, a pair of unimodal subtrees joined by crossmodals; the term "unimodal subtree" is used to distinguish an ordinary well-formed DOP subtree from a bimodal subtree); the exact details of which combinations are permissible – what counts as a well-formed bimodal subtree – differs between the two models tested (see §§3.2 and 3.3 below for the details), so the function of crossmodals differs between the two models. In constructing the templates for the training corpora for the models, three factors determined the decisions as to which nodes in associated trees should be connected by crossmodals;

1) If terminal content under verbal node n^w (meaning, the set of words/morphemes represented by those proper descendent[7] nodes of n^w that have no daughter nodes) can be used to *refer* to the terminal content under visual node n^v (the set of primitive visual objects represented by those descendent nodes of n^v that have no daughter nodes), then n^w and n^v will be connected with a crossmodal. The teal crossmodals shown in figure 3.2 are of this type.

2) If verbal terminal node tn^w (a terminal node here being a node with no daughters, representing a word/morpheme or primitive visual object) can be used to refer to visual terminal node tn^v, then tn^w and tn^v will be connected with a crossmodal. The blue crossmodals shown in figure 3.2 are of this type.

---

[7] Note that the term "descendent" is to be distinguished from "proper descendent"; that is to say, the set of descendants of node n includes n, whereas the set of proper descendants does not.

3) If a verbal terminal node tn^w can refer to the part-whole relationship between visual node tn^v and its proper descendent $tn^{v'}$, then tn^w will be connected to tn^v and $tn^{v'}$ with a crossmodal. The orange crossmodals shown in figure 3.2 are of this type.
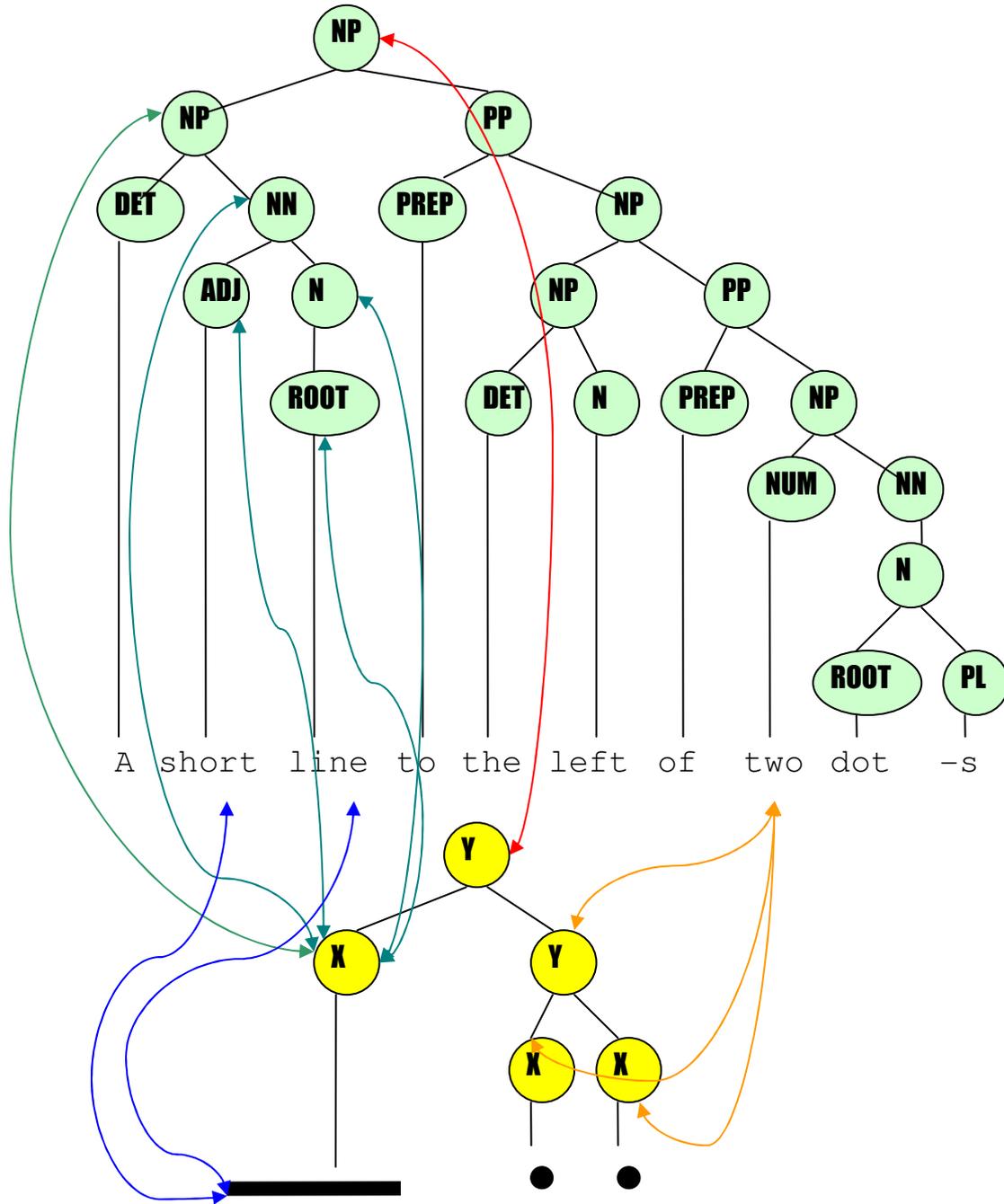

Figure 3.2: Paired image and description trees, with crossmodal connections; note that some crossmodals have been omitted for clarity. Nodes and crossmodals have been coloured for clarity only[8]

---

[8] Some further comments on the parses are warranted here; (next page)

In rough summary, the algorithm stores a corpus of such image/description pairings; the images employed are no more than one-dimensional arrangements of lines, dashes and dots. These are paired with, and crossmodally linked at multiple nodes to, noun phrases describing the images. The bimodal tree-pair shown in figure 3.2 is an actual example of an entry from the training corpus used in the model. When presented with visual stimuli, the algorithm, beginning with a bare Y- node and a bare NP-node, generates novel tree-pairs by substituting the visual and verbal parts of paired subtrees extracted from the training corpus for random non-terminal leaf-nodes in the incomplete derivation, until one or both of the trees has no non-terminal leaf-nodes. For any one stimulus, multiple derivations will be completed, which are used as a Monte-Carlo sample; a Monte-Carlo sample being a random sample drawn from an unknown probability distribution, used to estimate the probabilities of the distribution; in this case, it is used to approximate the most probable output.

More formally;

1) On the presentation of a novel visual stimulus, image *i*, the algorithm generates a bare Y node and a bare NP to act as the first non-terminal leaf nodes in the derivation (note that Y and NP are the labels on the root-nodes of all image and verbal trees respectively in the training corpus.

2) A non-terminal leaf-node from the image of the derivation, and one or more from the verbal side, are selected to be substitution sites. The criteria for node-selection differ in the two different versions of the algorithm that were implemented, and so these will be detailed below in their respective sections. To clarify, a non-terminal leaf node is one which carries a node-label, such as *NP*, *PREP*, etc, (non-terminal), but has no daughters at the current stage of the derivation, and so can serve as a substitution site.

---

1) The parse of "to the left of" in the verbal tree is somewhat non-standard, and requires comment; this parse was chosen over the more typical interpretation, that "to the left of" and "to the right of" are in fact complex prepositions, because I wanted avoid treating any element as idiomatic; I wanted to ensure that the whole tree was treated as compositional and decomposable, thereby giving the simulations a harder job in the test stage. (Chapter 5).

2) "line" and "dot" are immediate daughters of *ROOT* nodes, whereas "left" is not. This is because "left" does not take a plural, whereas "line" and "dot" do.

3) The node-label *NN* refers to a nominal group.

4) The nodes of the visual tree are labelled *X* and *Y* to distinguish those nodes that are the immediate mothers of terminal nodes (*X*) from those which are not (*Y*)

3) A random equiprobable[9] unimodal visual subtree is extracted from the training corpus, subject to certain constraints to be enumerated below. To be exact, the probability of any token (tv^{token}) of visual subtree tv which meets the set of constraints C^{visual} is given by the equation;

$$P(tv^{token}) = \frac{1}{\sum\limits_{\{tv':tv' \ meets \ C^{visual}\}} |tv'|}$$

(5.1)

Where t' is any subtree (type), and $|t'|$ is the number of tokens of that type. The constraint-set C^{visual} is simply intended to exclude any substitutions which, without seeing any of the rest of the corpus, could be shown to utterly preclude the generation of a tree with an arrangement of leaf nodes exactly corresponding to the arrangement of elements in the stimulus. Specifically, C comprises the following constraints, in relation to a stimulus s, where s is that sub-part of image i which occurs between the elements of the image corresponding to the closest terminal nodes or peripheries[10] to the left and right of the substitution site (see figure 3.3):

a) The root-node of t, t(root) must be labelled with the same label as the selected substitution site.

b) The frontier of t, t(frontier) – that is to say, the leaf-nodes, both terminal and non-terminal, of t, must contain no terminal nodes not corresponding to any element in the stimulus.

c) t(frontier) must contain no terminal nodes corresponding to elements in the stimulus in any order not found in the stimulus.

d) t(frontier) must contain no subtree containing more leaf-nodes between any two terminals or peripheries than there are elements in the stimulus between the corresponding positions in the stimulus.
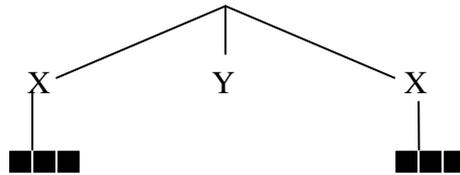
---

[9] Note that it is tokens of corpus subtrees that are equiprobable, not types; if a particular type is found twice in the corpus, it will be twice as probable as one that is only represented once.
[10] Periphery is here to be taken to mean the black space to the left or right of the leftmost or rightmost nodes in the tree respectively

e) *t(frontier)* must contain no subtree containing zero leaf-nodes between any two terminals or peripheries if the number of elements in the corresponding space in the stimulus is greater than zero.

If *Image (i) =* ■■■   ■■■■■   ■   ■   ■■■

And *Partial derivation (T) =*

X   Y   X
■■■   ■■■

then *Stimulus (s) =*   ■■■■■   ■   ■   **and therefore**

t =   Y
   X   X   X   X
■■■■■   ■   ■

t' =   Y
   X   X   X   X
■■■■■   ■   ■

t" =   Y
   X   X
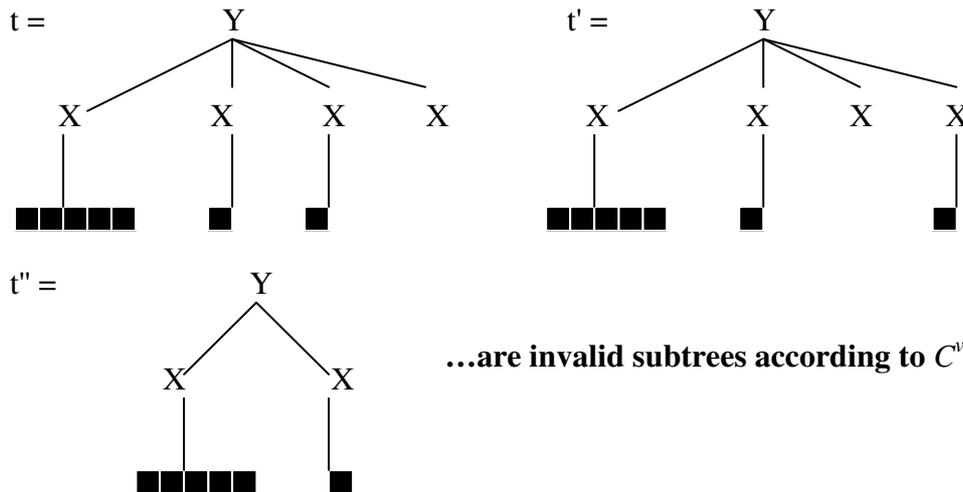■■■■■   ■

…are invalid subtrees according to C^{visual}

Figure 3.3: Given image *i* and partial derivation *T*, the stimulus by which the generation of the next subtree is constrained will equal *s*; in the case shown here, *t* and *t'* violate *C(d)*, while *t"* violates *C(e)*.

The procedure of on-the-fly subtree generation, or "Subtree Roulette", allows for the use of all subtrees without their needing to be stored and represented individually – rather one simply stores the corpus as whole trees; this serves as an alternative to Goodman's (2003) PCFG-reduction method. Using this method, derivations are spatially and temporally linear in relation to tree- and corpus-size. There is not space here to fully detail the technicalities of this method, but in short, each tree-node is annotated with the number of subtrees it heads (given by

equation 3.2), and each tree is annotated with the number of subtrees it contains for each node-label (i.e., if a tree contains two NP-nodes, one with 32 subtrees, the other with 4, the total will be listed as 36). Then, if, for example, a subtree headed PP is needed for a substitution, all trees containing PP-nodes are assigned a sub-part of a range of integers from 0-N_{PP}, where N_{PP} is the total number of PP-headed subtrees in the corpus, and each tree's part of the range is proportional to the number of PP-headed subtrees in that range. A random number between 0 and N_{PP} is rolled, and the tree assigned the part of the range within which the random number fell is chosen; if the chosen tree contains two or more PP-nodes, this "roulette wheel" procedure is repeated to decide between them. Finally, a top-down breadth-first traversal of the tree is made, omitting the root-node of the subtree, at each node i casting a random number between zero and t_i, where t_i is the number of subtrees headed by node i, and removing all descendants of i if the roll come up as zero. In this way, all subtrees headed with PP have a probability of 1 over N_{PP}.

4) This visual tree is combined with one random equiprobable unimodal verbal subtree tw (or, depending on the version of the algorithm, a set of verbal subtrees $tw_1 \ldots tw_n$) to form a bimodal subtree. The generation of verbal subtrees is subject to the following constraints, C^{verbal}.

   a) tw must be taken from the same corpus tree-pair as the visual tree already selected.

   b) tw must be rooted in nodes bearing the same labels as those selected to be substitution sites in step 2

   c) The resulting bimodal subtree must also meet the the criteria for the well-formedness of bimodal subtrees. These criteria, like the criteria for node-selection, differ in the two versions of the algorithm, and will be detailed below in their respective sections.

The probability of any verbal unimodal subtree (token) tw^{token} or tw_i^{token} is given by the equation below;

$$P(tw^{token}) = \frac{1}{\sum_{\{tw':tw'\ meets\ C^{verbal}\}} |tw'|}$$ (3.2)

5) The bimodal tree is substituted for the nodes at the selected substitution sites, and the elements in the image corresponding to terminal nodes on the image side of the subtree are marked as having been accounted for by those terminal nodes; this is to facilitate the finding of the stimulus *s* for the next substitution, as specified in step 3.

6) Some nodes in the visual trees exist in slave-master relationships to their immediate sister-nodes. Slave nodes cannot be selected as substitution sites, but if a master node is a substitution site, then a copy of the subtree substituted will also be substituted at each slave, and each non-terminal leaf-node of the copy will aso be enslaved to the corresponding node in the original. Similarly, when image subtrees are extracted from the corpus, the material underneath any slave-node must be identical to the material underneath their master. This is necessary to allow groups of repeated elements to be recognised as such, like the pair of dots in the image-tree in figure 3.2; this is essential for the algorithm to be able to subitise such groups and reliably describe them in numerical terms. For a fuller explanation of the job done by this subsystem, see §4.3.3.

7) If no bimodal subtree can be found, the algorithm "backtracks" by undoing the last bimodal subtree substitution; that is to say, all descendant nodes of the last successful substitution site will be removed from the tree. Arbitrary constants can be set to limit the number of such reversals before the preceding substitution must also be undone, or, globally, before a derivation be abandoned outright as a dead-end. In all test runs conducted for the present study, these constants were set at 100 and 100,000 respectively.

8) Steps 2 to 7 are repeated until either one or both of the trees is complete (i.e., has no non-terminal leaf-nodes), or, as specified in step 7, the derivation is abandoned. If the derivation is successful, the resulting verbal tree is stored (whether complete or not) as part of the Monte-Carlo sample.

9) Steps 1 to 8 are repeated until *N* many trees were accumulated in the Monte Carlo sample. In all test runs conducted for the present study, *N* was set at 500.

10) Because the verbal strings contain more elements (words/morphemes) than the visual images employed, the trees by which they are parsed contain far more nodes, and therefore more potential substitution sites. Therefore, the great majority of derivations will result in incomplete verbal trees. For this reason, it is not the most frequent verbal output that is selected from the Monte Carlo sample,

but the graph-theoretic unification of the largest unifiable subset of the sample. Because of the high prevalence of incomplete verbal trees in the output from derivations, instead of simply polling the Monte-Carlo set for the most frequent output, an algorithm, the details of which not relevant here, was used to find the largest unifiable subset of the trees in the sample. Two trees are taken to be unifiable if there is at least one possible (not necessarily complete) tree of which both trees are co-racinous[11] legal subtrees according to the unimodal wellformedness criteria of DOP1. The unification of the two trees, then, is the smallest tree that meets this description, if any tree can. Two unifiable trees and their unification are shown in figure 3.3. The system's output, then, is the unification of the largest unifiable subset of the sample.
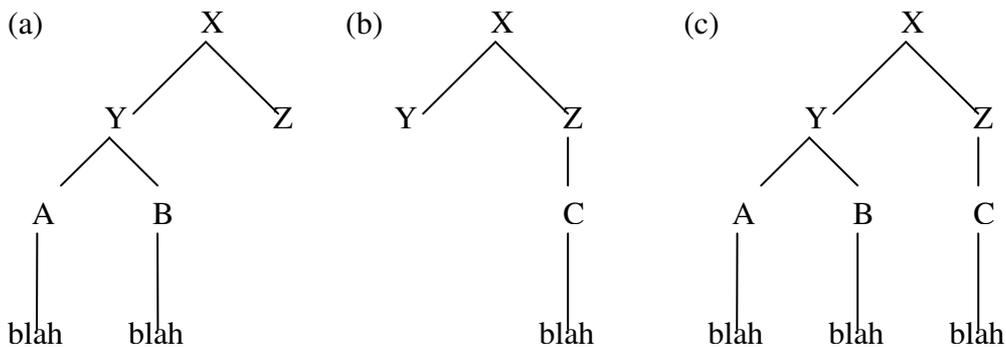

Figure 3.4; two unifiable trees (a, b) and their unification, (c); the working assumption here is that, although the trees output by UDOG are incomplete, they will tend, if the algorithm is working, to be fragments of correct outputs; therefore unifying them allows complete (or at least, closer to complete) trees to be made. If fragments of correct trees are indeed the most frequent output, the largest unifiable set should unify into a correct complete output.

As noted above, two versions of the criteria for substitution-site selection and bimodal subtree wellformedness were tested; one in which the statistical regularities of the training data alone were trusted to do the job of solving the "binding problem", and another in which the problem was addressed directly by restrictions placed on the selection and wellformedness criteria. The former, I shall refer to as "Naïve UDOG", and the latter as "Binding UDOG". The problem of binding may be stated as simply being the problem of ensuring that subtrees are placed in such a manner as to ensure that their proper places according to their semantic relationship; for instance, if full

---

[11] Subtrees t^1 and t^2 are co-racious iff $root(t^1) \equiv root(t^2)$.

sentences were being generated, part of this problem would be to ensure that the subject and object NP's are placed in the subject and object positions respectively.

3.2 Naïve UDOG

In the naïve version of UDOG, exactly one substitution site are picked at random from the non-terminal leaf-nodes of each tree. A bimodal subtree is well-formed iff;

1) Both of the component unimodal subtrees are well-formed by the normal standards of DOP1.
2) Both unimodal subtrees should originate from the same tree-pair.
3) The verbal subtree should contain only nodes which either;
   a) Have no crossmodal connections at all, or
   b) Have crossmodal connections, at least one of which is to a node in the visual subtree.

It is worth seeing how this plays out in practice with a toy example. Consider the follow example of a corpus tree-pair *P*:
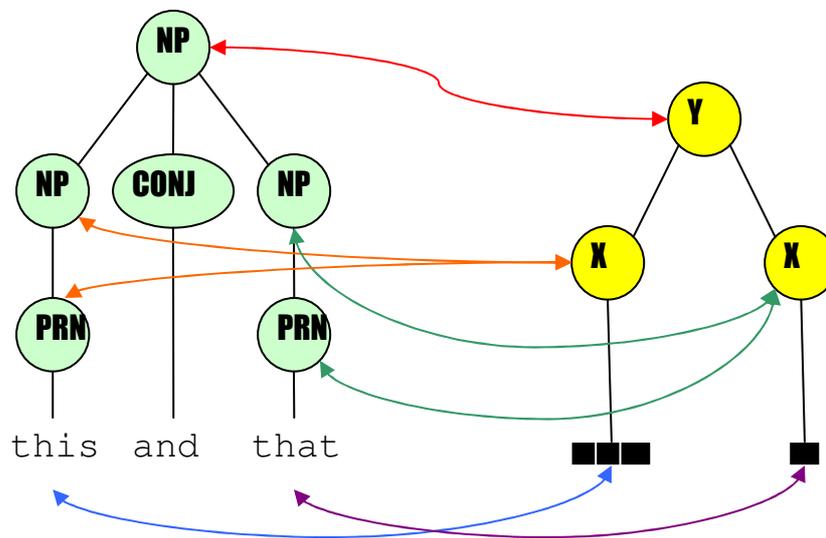

Figure 3.5; a toy tree-pair; here, all crossmodals are shown

Let us suppose that the following unimodal subtree *tv* is taken from the visual tree;
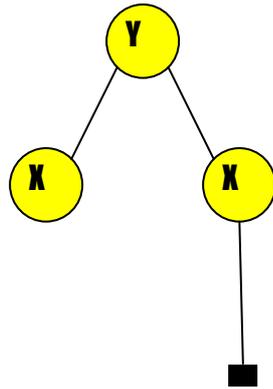
Figure 3.6; a subtree (*tv*) of the visual tree in fig. 3.5

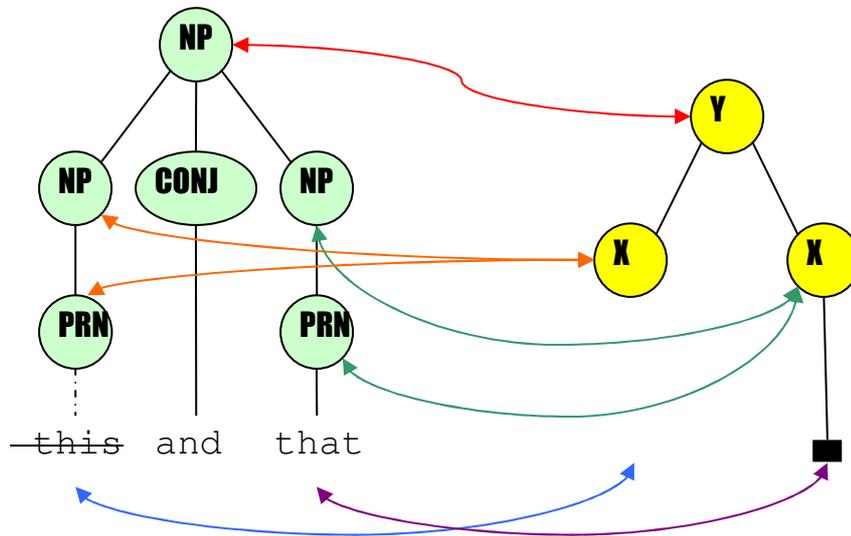This would leave one node of the verbal tree in violation of (3);

Figure 3.7; One node in the verbal tree cannot occur in a verbal subtree in a valid bimodal subtree with *tv*

Any subtree *tw* of the remainder of the verbal tree may validly be combined with *tv* to form a bimodal subtree; below are some examples;
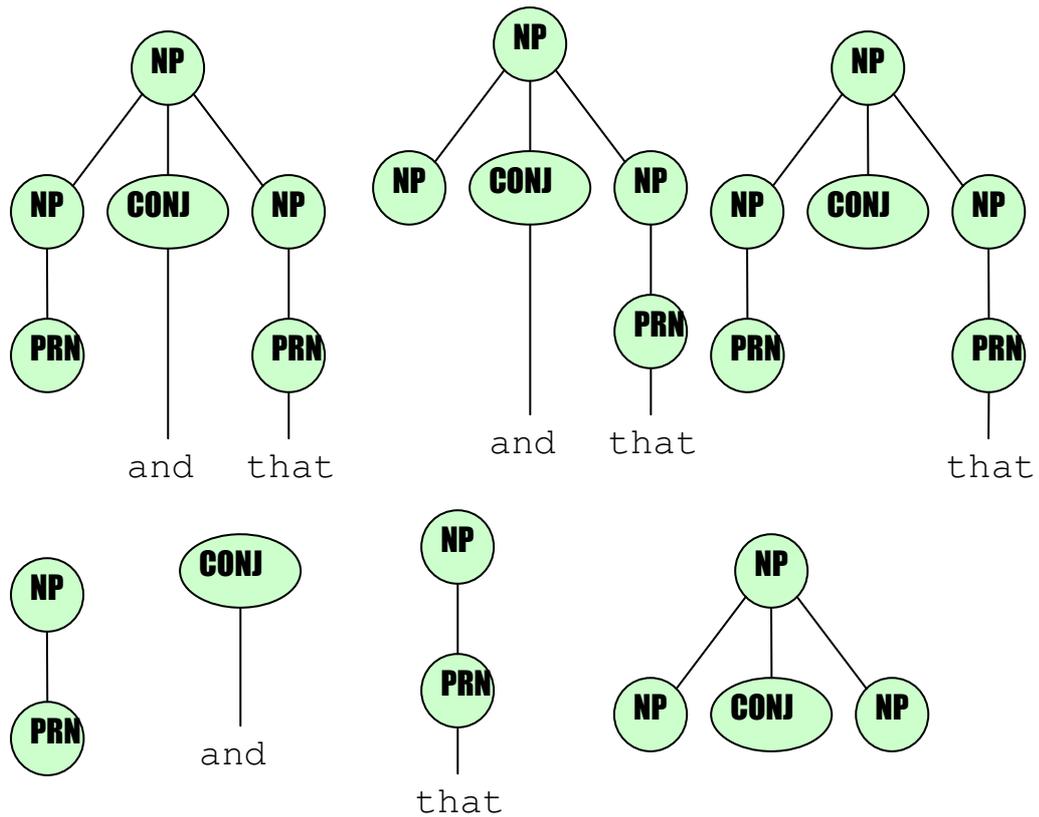
Figure 3.8; A non-exhaustive set of verbal trees *tw* that may validly be combined with *tv*

3.3 Binding UDOG

In Binding-UDOG, exactly one substitution site on the image tree is chosen at random, but then if this substitution site is crossmodally connected to any of the potential substitution sites on the verbal tree, it can in theory substitute subtrees at *all* of these sites, because a Binding-UDOG bimodal subtree can have one *or more* verbal subtrees. The wellformedness criteria are as follows;

1) All of the component unimodal subtrees are well-formed by the normal standards of DOP1.
2) All unimodal subtrees should originate from the same tree-pair *P*.
3) Each verbal subtree should contain only nodes which either;
    a) Have no crossmodal connections at all, or

b) Have crossmodal connections, at least one of which is to a node in the visual subtree.

4) The root node of each verbal subtree should be crossmodally connected to the root node of the visual subtree.

5) No root node of a verbal subtree can be an ancestor of descendant of the root node of any other; that is to say, if nodes n_1 and n_2 are in an ancestor-descendant relationship *in the verbal tree W of the originating corpus tree-pair*, they cannot both be selected to be the head-nodes of subtrees in the same bimodal subtree.

6) The set of verbal subtrees in a well-formed bimodal subtree cannot be a proper subset of the set of verbal subtrees in any other well-formed subtree.

7) For each node-label L represented x many times in the set of possible substitution sites, there should be no more than x many verbal subtrees in the bimodal subtree with root-nodes labelled L.

The algorithm exhaustively checks all possible subsets of the set of nodes in the verbal tree connected to the root node of the visual subtree for validity, according to the standards of (5), (6) and (7). A subset is chosen at random, at a probability modelled by the equation 5.3 below;

$$P(S) = \frac{\sum\limits_{node_i \in S} subtrees_{node_i}}{\sum\limits_{S_i \in V} \sum\limits_{node_j \in S_i} subtrees_{node_j}} \tag{5.3}$$

-Where S and S_i are sets of nodes, V is the set of valid sets of nodes according to criteria (5), (6) and (7) above, and $subtrees_{node_x}$ is the total number of subtrees rooted in $node_x$. The total number of subtrees of any node $node$ can be found using equation 5.4;

$$subtrees_{node} = \prod_{\{node_i : node \equiv mother(node_i)\}} (subtrees_{node_i} + 1) \tag{5.4}$$

For each node *n* in the chosen set, a subtree *t* for which *n* = *root(t)* is chosen at a probability modelled by equation 5.5 below;

$$P(t) = \frac{1}{subtrees_{t(root)}}$$ (5.5)

If it is either not possible find a substitution site, or a bimodal subtree, that meets the above criteria, the system backs off to the criteria of naïve UDOG.

The purpose of the additional conditions is to alter the character of substitutions in binding UDOG, from individual substitutions of subtrees for non-terminal leaf-nodes, to the substitution of a whole complex of crossmodally root-connected subtrees for a complex of crossmodally connected non-terminal leaf-nodes, thereby preserving ordering relations across substitutions.

This whole process is rather complicated, so it is worth drawing out with examples. Let us first consider substitution-site selection; consider the figure 3.9 below as an example of a partial derivation *D*:
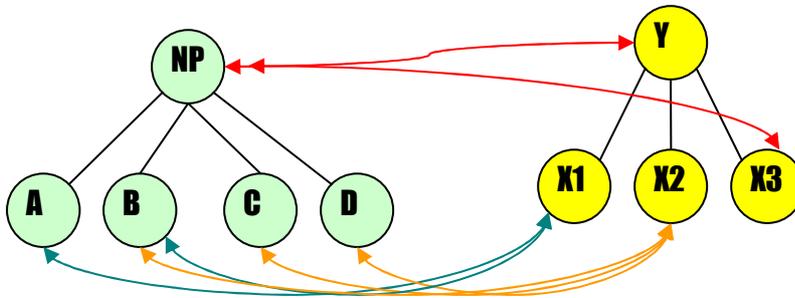
Figure 3.9; Partial derivation; all crossmodals shown

The parser selects randomly from X1, X2 and X3. Which of the non-terminal leaf nodes of the partial verbal tree D^{verbal} are available as substitution sites depends on which non-terminal leaf-node of D^{visual} is selected; if it is X1, A and B are available as substitution sites; if X2, it is B, C and D; if it is X3, X3 is not connected crossmodally to any *available* node, so the algorithm backs off to the rules and constraints of Naïve UDOG and selects exactly one of A, B, C or D at random. Now,

let us suppose that it selects X2, and selects the subtree headed by the X2' node of visual tree P^{visual} of paired tree P shown in figure 3.10 below:[12]

blah–de–blah blah blah
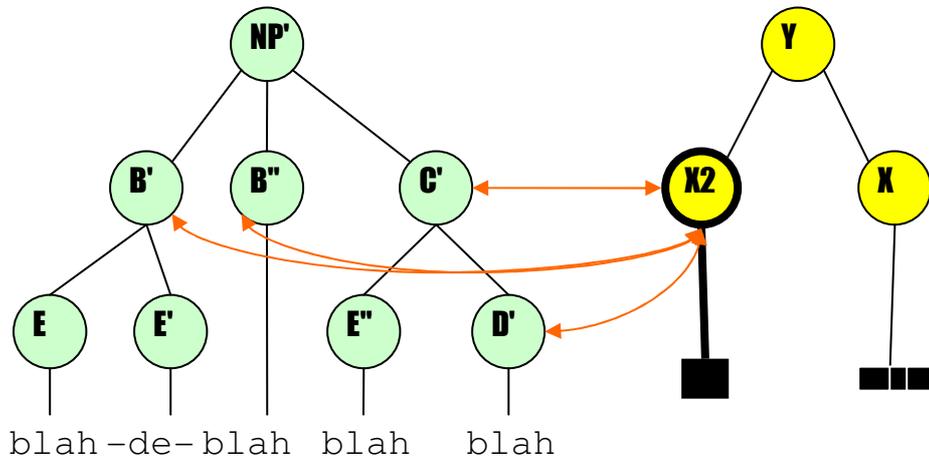
Figure 3.7; Paired trees *P*: for simplicity we will say that the set of crossmodals shown is exhaustive. Nodes and connection with emphasis represent the selected visual subtree *tv*.

Recall that the available substitution sites are labelled B, C and D. This means that all the nodes crossmodally connected to X2 can potentially head verbal subtrees in bimodal subtree *tb*. However, not all of them are mutually compatible. B' and B" cannot both be used, because there is only one substitution site labelled B waiting to be filled. C' and D' cannot both be used because they are in a ancestor-descendent relationship; we have four possible sets to be selected from; {B', C'}, {B', D'}, {B", C'} and {B", D'}. Now let us work out the numbers of subtrees:

$subtrees_E =$	$(0 + 1) =$	1
$subtrees_{E'} =$	$(0 + 1) =$	1
$subtrees_{E''} =$	$(0 + 1) =$	1
$subtrees_{D'} =$	$(0 + 1) =$	1
$subtrees_{B'} =$	$(subtrees_E + 1)(subtrees_{E'} + 1) =$	4
$subtrees_{B''} =$	$(0 + 1) =$	1
$subtrees_{C'} =$	$(subtrees_{E''} + 1)(subtrees_{D'} + 1) =$	4
$subtrees_{NP'} =$	$(subtrees_{B'} + 1)\ (subtrees_{B''} + 1)(subtrees_{C'} + 1) =$	50

---

[12] Note that, as the mother of a terminal node, X2' only has one possible subtree. Also note that the primes used here do not denote a difference of node-label, and are only used to distinguish same-labelled nodes in *D* and *P*.

Therefore;

$subtrees_{\{B', C'\}} =$	$(4 + 4) =$	8
$subtrees_{\{B', D'\}} =$	$(4 + 1) =$	5
$subtrees_{\{B'', C'\}} =$	$(1 + 4) =$	5
$subtrees_{\{B'', D'\}} =$	$(1 + 1) =$	2
	TOTAL =	20

Table 3.1; totals of subtrees in valid sets.

Thus;

$P(\{B', C'\}) =$	$\dfrac{8}{20} =$	0.4
$P(\{B', D'\}) =$	$\dfrac{5}{20} =$	0.25
$P(\{B'', C'\}) =$	$\dfrac{5}{20} =$	0.25
$P(\{B'', D'\}) =$	$\dfrac{2}{20} =$	0.1

Table 3.1; probabilities of subtrees in valid sets.

Let us suppose that {B', D'} is selected. D' only has one possible subtree, tw_1:
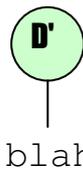
Figure 3.11: tw_1

B' has 4 possible subtrees;
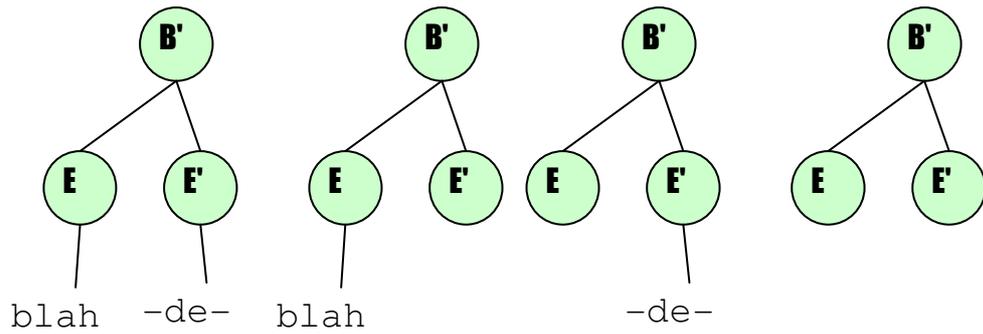
Figure 3.12: subtrees headed with B'.

Now let us suppose that the third of these is selected as tw_2, at a probability of 0.25 (all four subtrees are have the same probability) This gives us the following bimodal subtree tb;

Figure 3.13; Bimodal subtree tb.

Finally, tb is substituted into P ($P \circ tb = P'$)[13]:

_____

[13] Recall that $\circ$ is the substitution operator

Figure 3.14; Bimodal substitution operation.

Chapter 4

Understanding

the Models

Before going any further, some comment should be given regarding the rationale of the models; in particular, it is necessary, in the design of any computational model of a natural system, to distinguish scrupulously between those features that are intended to approximate the supposed real features of the syste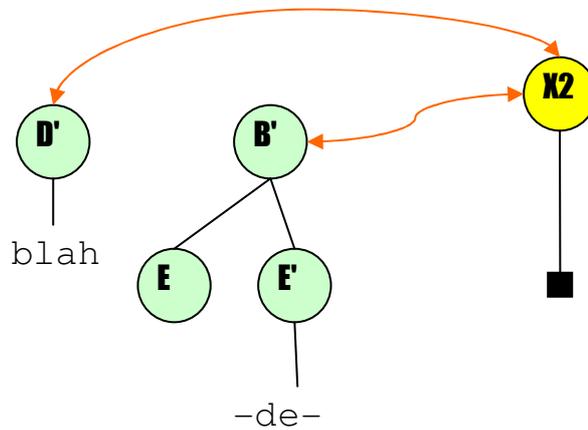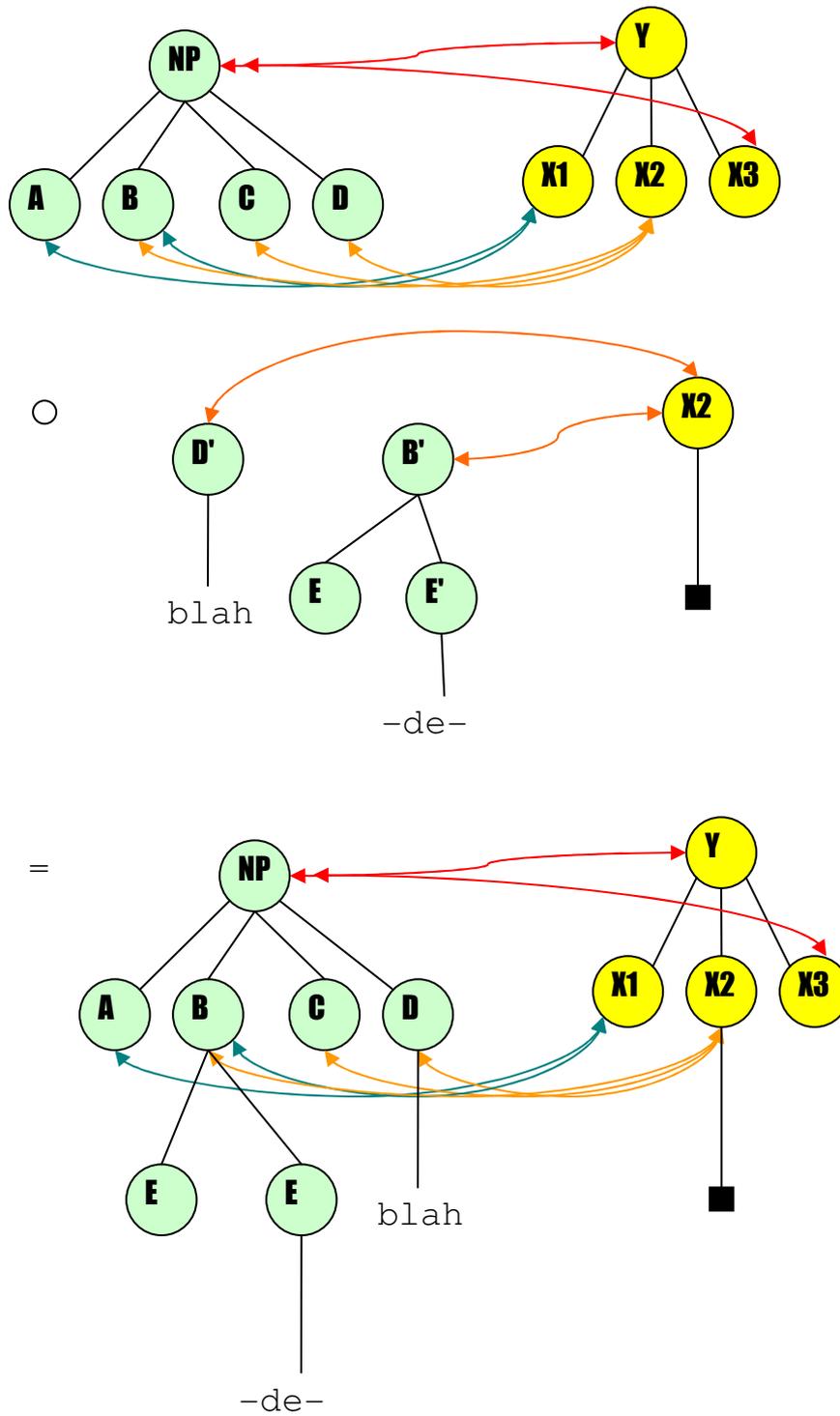m modelled, and those that merely constitute technological fixes of technological problems. This is a thorny problem, not only because it is necessary to open up thorny issues in the epistemology of modelling and simulation, but also because some of the specific issues involved cut across the spectrum of Data-Oriented research. I will first introduce in outline some of the relevant general issues of epistemology and method in modelling, before outlining how that has cashed out in practice, in Data-Oriented research generally, and in the current work in particular.

4.1 Approximation and representation in simulation; the useful fiction of the substrate neutrality of algorithms

With any novel development in scientific technique, new methodological and epistemological questions and challenges are sure to follow; and no new development in the past fifty years has had so been so pervasive across all disciplines of science as the development of computational simulation. Rohrlich (1991, cited in Hartmann

1996) characterises simulations (specifically discussing those used in physics) as "a qualitatively new and different methodology … that … lies somewhere intermediate between traditional theoretical physical science and its empirical methods of experimentation and observation" As such, a new literature in the Philosophy of Science has arisen to evaluate the methodological limitations and epistemic potentials of computational simulation. Hartman (1996) draws out the relationship between simulations and dynamic theoretical/mathematical models; under his understanding, such models comprise the mathematical characterisation of "a set of assumptions about some system" (p.4, citing Redhead 1980), and;

> "a simulation results when the equations of the underlying dynamic model are solved. This model is designed to imitate the time-evolution of a real system. To put it another way, a simulation imitates one process by another process. In this definition, the term 'process' refers solely to some object or system whose state changes in time."
>
> (ibid. p.5)

The crucial tacit assumption here is that of the *substrate neutrality of algorithms*. In the ideal case, the model should describe the algorithm instantiated by both the simulation, where the particular implementation, in some particular programming language, on some particular computer, serves as the substrate, and the natural phenomenon, in some other substrate. However, it is important to realise that such substrate-neutrality is in fact a mathematical fiction; just as geometry defines lines and points as one- and zero-dimensional objects with zero area and volume, though no such objects exist in the real world, the Second Law of Thermodynamics requires that no algorithmic process in nature can continue indefinitely, without its causal processes eventually being interrupted or perturbed by the causal processes of its substrate[14]. It is simply that if, say, the behaviour of an algorithm running on a computer is changed by a bug in the implementation, or a block of memory being destabilised by the machine overheating, one says "*that's* not the algorithm." An algorithm is an axiomatically defined mathematical entity which, by *fiat*, allows us to

---

[14] This of course should not be taken as a denial of the great utility of algorithmic thinking, any more than it should be taken as a denial of the obvious usefulness of geometry.

parse the totality of a natural (or computational) phenomenon into "algorithm" and "substrate".

A good example here is the debate between Dennett (1995) and Gould (Gould and Lewontin 1979, Gould 1997) regarding adaptationism, and in particular Gould and Lewontin's (1979) hypothesis that non-adaptations, such as historical accident, the recycling of obsolete systems towards new functions, and especially "spandrels", structural by-products of adaptive changes, capable of being subsequently co-opted to adaptive functions.(Gould and Lewontin 1979). Dennett, against this, claims that "either spandrels are not ubiquitous after all, or they are the *normal basis* for adaptations, and hence no abridgement at all of pervasive adaptation" (Dennett 1995, p.268). What this disagreement boils down to, I would contend, is a difference in the way the parties involved parse the phenomenon into algorithm and substrate; Dennett chooses to isolate only adaptation by transmission of mutation with selection as "algorithm", and the residue is "substrate". Gould regards non-adaptations as sufficiently important to the final state as to warrant their co-option from "substrate" to "algorithm". In this light, given that in either case the algorithm remains a mathematical fiction, the disagreement must be recast as one about what constitutes the optimal scientific *strategy*, rather than the correct scientific *finding*.

The relationship between the simulation and the natural phenomenon becomes further attenuated by the fact that it is not always feasible to implement the algorithm given by the theoretical model exactly as stated. Constraints of computational tractability, for instance, can steer a simulation away from the idealised form of its model. Krohs (2006) gives a nice example of Field and Noyes' (1974) model of the Belousov-Zhabotinsky (BZ) chemical oscillator. Krohs notes that while the BZ reaction itself takes place in continuous time, the equations of Field and Noyes' model can only be (tractably) solved in discrete time-steps, generating an error which can be reduced by increasing the number of time steps, up to the limit case where the number of steps is infinite and the error is zero. However, the floating point arithmetic employed by computers creates a rounding error with every time step, so that beyond a certain point the reduction in discretization error caused by an increase in temporal resolution is outweighed by the increase in rounding error. Similarly, the feasible implementation of Data-Oriented models falls somewhat short of the idealised "textbook" form of DOP, as will be shown below.

4.2 Approximation and idealisation in Data-Oriented research

Now, recall my thumbnail sketch of DOP1 in §2.1. This "textbook DOP" is in fact an idealisation from any real computational implementation of the algorithm. There are three principal areas in which real DOP implementations must exercise technological fixes in order to *approximate* the "ideal" version; any DOP implementation must have some way of dealing with the space complexity of representing the set of all well-formed subtrees, which is as we shall see below exponential in relation to string length, and the time complexity of the task of finding the most probable parse, which is exponential in relation to corpus size.

Also, more worryingly, despite being capable of representing dependencies of indefinite distance, provided they are represented in the training corpus, DOP in its *ideal* form is also only an approximation to human linguistic performance, since DOP grammars are limited to Context-Free-ness, and cannot reliably represent the (weak) Context-Dependence of natural language. As we shall see below, there are both theoretically significant and (merely) technological fixes to this problem

4.2.1 Storing subtrees

To see that the complexity of storing all subtrees is spatially exponential in relation to string-length, consider the subset of subtrees, for any given tree, rooted in the root node of the overall tree, in which there are either no non-terminal leaf nodes (i.e, the subtree and the tree are identical), or the only non-terminal leaf nodes are those which, in the whole tree, are the immediate mothers of terminal nodes. Each unit of the string, then, corresponds to a two-valued parameter of the subtree; it may either be deleted, leaving its mother-node as a non-terminal leaf-node, or it may be retained. These parameters are orthogonal to each other, so if there are N many units in the string, there will be 2^N possible subtrees of the specified type. Since this is a proper subset of the total set of legal subtrees, it follows that the size of the complete set of legal subtrees will also be exponential in relation to string-length.

The first DOP implementations solved this problem simply by either limiting string-length (Bod, pers com), or taking a random sample of the possible subtrees, thereby

approximating the ideal form of the algorithm, with some success. Goodman (2003) removed the need to store all subtrees by developing a method known as "PCFG-reduction" (probabilistic context-free grammar), in which one generates a PCFG in which (binarized) DOP subtrees are constructed "on the fly" during derivation. Each CFG rule is expanded into eight PCFG rules, which differ according to whether each of the nodes (mother, left daughter, right daughter) is accessible to the substitutions entailed in constructing a subtree or in combining subtrees. The probabilities of the rules are modulated to ensure that the aggregated probability of a reconstructed subtree is the same as the probability of the same subtree in DOP. I will not expand here on the technical details, as I use my own alternative (albeit similar) solution to this problem in implementing UDOG.

4.2.2 Finding the most probable parse

The problem of finding the Most Probable Parse (MPP) presents an even more serious challenge. To quote Bod (1998, p43);

> A sentence may have exponentially many parse trees and any such tree may have exponentially many derivations Therefore, in order to find the most probable parse of a sentence, it is not efficient to compare the probabilities of the parses by exhaustively unpacking the chart Even for determining the probability of one parse, it is not efficient to add the probabilities of all derivations of that parse.

Sima'an (1996) has proven that no polynomial-time algorithm can deterministically find the most probable parse. Numerous approaches have been employed to approximate the MPP. Often, this is achieved by using heuristic to exclude classes of subtrees (Sima'an 1999, Way 1999). However, Bod (2003b) has demonstrated that any limitation on the range of allowable trees results in a loss of accuracy; thus the method of choice remains the simplest; Monte-Carlo sampling, wherein an arbitrarily large sample of random parses is taken, and the most frequent is selected as the system's "best guess" at the most probable parse, with a chance of error that can be made arbitrarily small (Bod 1998, p.45).

4.2.3 Context-Dependence

The problem of consistently dealing with context dependencies is of a somewhat different character than the problems above, in that context-freeness is "built in" to even the idealised form of DOP; any DOP subtree *t* may be (with information loss) be rewritten as context-free rewrite rules, of the form *root(t)* → *frontier(t)*, as demonstrated in figure 4.1;
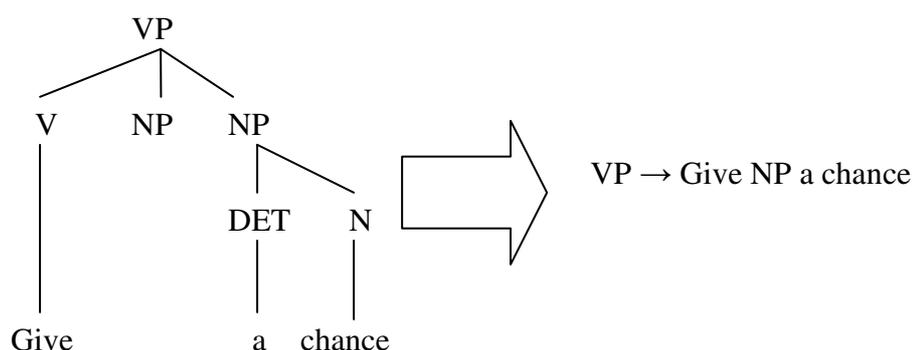

Figure 4.1: A DOP subtree rewritten as a context-free rewrite rule

Because DOP uses all corpus subtrees of any depth, any long-distance dependency can be modelled *if* it is present in the training corpus. However, it cannot generalise over such dependencies Several solutions in which the ideal form of DOP is changed in theoretically significant ways; using either Lexical-Functional Grammar annotations (Bod and Kaplan 1998) or Tree-Insertion (Hoogweg 2000); these methods are not our concern here. What I wish to note here is that another way around the problem exists, which I count as a "technological fix" rather than a theoretical development; that is, the problem may be ignored. Human language is only weakly Context-Dependent, and only a small number of natural language utterances in any given corpus will contain long-distance dependencies. Moreover, if since DOP *can* model long-distance dependencies that are present in its training data, if the subtrees containing such dependencies are in Zipfian (Zipf 1949) distribution across the training and test corpora, most such dependencies encountered in the test data will be present in the training data also. A recent trend in DOP research has been towards models that accord increasingly greater importance to *simplicity*, which is to say, shortness of derivation (i.e. using fewest subtrees) rather than likelihood. Bod (2003a)

describes SL-DOP (Simplicity-Likelihood), in which the simplest parse is selected from the *n* likeliest; for the optimal value of *n*, at $n = 12$, SL-DOP attained a state-of-the-art F-score[15] of 90.7%. This has since been bettered by Bod (2005), using DOP⁺, which simply selects the simplest parse and uses likelihood only as a tiebreaker when there is more than one simplest parse, with an F-score of 91.1%.. I attribute the success of these simplicity-oriented DOPs to the following two causes;

1) Bod (2003b) notes that although very large subtrees tend to be rare and therefore have low probability, they nevertheless are very important to approximating the MPP, because they account for more of the completed tree, thereby reducing the number of substitutions in derivations, thus requiring fewer subtree-probabilities to be multiplied together. Therefore simplicity and likelihood will often yield the same result.

2) Larger subtrees can capture longer-distance dependencies. By favouring larger subtrees SL-DOP and DOP⁺ *capitalise better upon the long-distance dependencies present in the training data.*

4.2.4 Model-DOP, Simulation-DOP, Real-DOP?

With the preceding remarks in mind, what are we to make of our hypothesis that DOP of some sort if broadly the correct hypothesis regarding how language is processed by human brains? What is the relationship between DOP in its ideal, "textbook" form (Model-DOP, to borrow Kroh's distinction), its implementation *in silico* (Simulation-DOP) and its hypothesised implementation *in vivo* (Real-DOP)? Unfortunately, the empirical work (here meaning laboratory experimentation on human subjects, rather than computational simulation) to test the validity of the claim that the human brain indeed implements DOP-like processes, let alone investigate what the implementational details of those processes is, has yet to be done. Indeed, most investigators, as in §1 noted above, are more interested in DOP for its potential as a technology, rather than as a model of human cognition. It is therefore unclear exactly

---

[15] The F-score is the standard measure of accuracy in parsing and other Natural Language Processing tasks; it is the harmonic mean of precission (percentage of elements in the output correct) and recall (percentage of elements in the correct parse found in output); for the equation for calculating the harmonic mean, see Chapter 6, footnote 18.

what epistemic (as opposed to merely pedagogic) work Model-DOP actually *does*. Recall for a moment the moral of the discussion of the Dennett/Gould debate in §4.1; that rather than being a type of process that one may find in the world, an algorithm is a mathematical/axiomatic specification whereby processes in the world may be parsed into "substrate" and "algorithm". Here is it important to recognise that science is principally a *strategic* activity; by which I mean that epistemic and methodological decision-making in science is governed principally by the desire to be able to do more science.[16] The idealisations made in algorithm-thinking allow us to generalise explanations over diverse phenomena, and it would be desirable if we were able to claim that Model-DOP is the core of the algorithmic specification that allows us to generalise explanations of the behaviour of Simulation-DOP to explanations of human linguistic behaviour; however, at best, this can only be part of the story. Because, as shown above, the parsing of the processes is a strategic move rather than an ontological commitment, different, even perhaps non-complementary, algorithmic specifications afford different generalisations that may themselves may prove to be complementary. We are therefore justified in asking whether alternative algorithm/substrate parses may afford novel generalisations; and in particular, whether some of the "technological fixes" present in Simulation-DOP are in fact also implemented in some form in Real-DOP. Bod (1998, p.49) notes, in a section on "Cognitive aspects of Monte Carlo disambiguation", "It is unlikely that people disambiguate sentences by sampling derivations, keeping track of the error probability of the most frequently resulting parse." However, in response to this, there are certain features of standard Model-DOP that we may wish to demote from algorithm to implementation; that is, the tacit assumption, with regard to the processing architecture, that exemplars are stored passively in memory and the recombination of subtrees into derivations and aggregation of derivations yielding the same trees takes place serially under a central processing system using a "worktable" of working memory. But an alternative story may be told wherein the neural representations of exemplars and the connections between them are themselves active computational units, and the brain computes multiple derivations in parallel as activation cascades

---

[16] Fully backing up this position would require a dissertation longer that the present one. Some of the claims are argued for in Cochran 2006b. Note that I this is not meant as any sort of cynical insinuation against scientists' commitment to truth; rather, I hold that even the working definitions by which "what it is for a scientific claim to be true" is determined are themselves the product of strategic considerations.

through the interconnected neural representations of trees; in this case, the "seriality" feature of Simulation-DOP would be demoted to a implementational detail, whereas Monte-Carlo sampling would lose its status as a technological "fix" for approximating MPP, and become co-opted to the algorithm proper. What I am proposing here is not the construction of a new Simulation-DOP[17], but rather a revised of *Model*-DOP whereby the relation if Simulation-DOP to Real-DOP may be reconsidered. Similarly, if such a parallelised DOP were to comprise a network in which the tree-nodes and connections were mirrored in the physical network, but supplemented with supernodes corresponding to node-labels (see figure 4.2 for a toy example of such a hypothetical model), Goodman's (2003) PCFG-reduction method could be taken to model the probability of an actvation at node *n* being passed locally, to *mother(n)* or *daughters(n)*, or via the "node-label" supernode to which *n* is connected, to some other node in some other
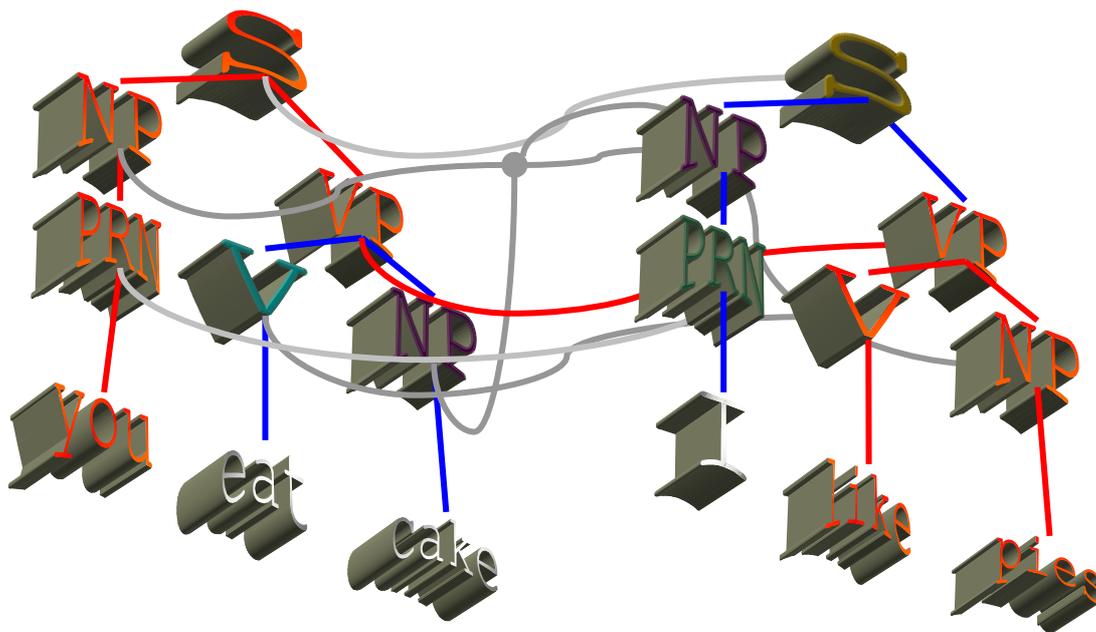
Figure 4.2; An idealised toy example of a parallelised active-memory DOP parser; on receiving an activation, each node computes the probability of passing activation on to its mother, its daughters, or, via a supernode, to any other tree-node with the same node-label. In the figure, nodes and connections are shaded red to illustrate the passing of activation on receipt of the stimulus "You like pies". With a larger corpus, such a model could generate multiple derivations in parallel

exemplar connected to the same supernode. In this way elements of PCFG-reduction (and elements of the Subtree Roulette method outlined in chapter 3; after all, the

---

[17] Which is not to say that I don't have active-memory parallel derivation DOP simulations earmarked as a potential future project.

exemplars would, in this picture, be stored as whole trees) may be promoted from being implementational fixes to a computational problem, to being features of the algorithm – of *Model*-DOP. It is with this in mind that I now wish to turn to the question of how Simulation-UDOG may be parsed into algorithm and implementation.

4.3 Reconsidering the present models

Three aspects of the UDOG systems outlined in Chapter 3 are particularly in need of comment here; the postprocessing of the partial trees outputted by the generation process, wherein the largest unifiable subset of the output is determined and the unification of those trees given as the final output; the uses made of the crossmodal connections between the paired trees; and the master-slave system used in the visual parses..

4.3.1 Unification

The unification system must quite straightforwardly be admitted to be a technological fix for a shortcoming of both Naïve and Binding UDOG; it was my hope that in redefining the wellformedness rules for bimodal subtrees, so that a valid bimodal subtree may draw more than one unimodal verbal subtree from the training corpus tree-pair from which it is sourced, binding UDOG would mostly produce whole outputs in both modalities; this proved not to be the case, and the unification system was developed simply as a stand-in for the development of further versions of UDOG which I hope will eventually produce complete outputs, and therefore no longer need to unify outputs. The Unification process is itself cognitively implausible, not least because it is in fact a variation on the "maximum clique" problem in graph theory, which is known to be NP-complete (Karp 1972, Zuckerman 1993, Wikipedia contributors 2006). The maximal clique problem, given a graph in which only a subset of the possible pairs of vertices are connected by edges, is the problem of finding the largest subgraph in which all vertices are connected by edges to all other vertices in the subgraph. If the output trees in the Monte Carlo set are each treated as one vertex, and edges are drawn between all only those pairs of output trees that are unifiable, it should be clear that this subtask of finding the maximum unifiable set is

equivalent to the Maximum Clique problem, and therefore the whole task of finding the largest unifiable subset is also NP-complete. A brute force algorithm was used to solve the problem, but the result was that the size of the Monte Carlo sample was severely restricted. Given the rapidity with which language users can compose novel sentences, it seems implausible that they should have to solve an NP-complete problem every time they try to do so.

4.3.2 Crossmodals

The role of the crossmodals is somewhat less clear-cut. It is open to two principal interpretations; the first is that they are merely needed as an heuristic tool for excluding unlikely subtrees from the verbal derivation, and that this is in fact only needed because of the smallness of the training corpus; given a much larger body of data, and a much larger preferably also a much larger Monte-Carlo sample, the statistical correlation of visual and verbal subtrees in the training data would be enough to do the required work, and crossmodals would be unnecessary. The second is that crossmodals are in fact essential to guarantee the binding of syntactic/semantic relations. The Naïve and Binding versions of UDOG may be said to represent the first and second interpretations respectively. In this case, I would contend that the strategically optimal interpretation of the Simulation-UDOGs – regarding whether crossmodals ought to be included in Model-UDOG at all – cannot be prejudged *a priori*. As will be seen in the next chapter, the tests performed on Naïve and Binding UDOG were designed to help point towards a resolution of this ambiguity.

4.3.3 Master and slave nodes

As noted in §3.1, some nodes in the visual parses exist in master-slave relationships to their immediate sisters; a the set of subtrees under a slave-node will always be *exactly* the same as the set of subtrees under its master-node, as illustrated in figure 4.4. The purpose of this is to ensure that a group of identical objects in the visual input - say, three adjacent dots - will be preferentially parsed so that their lowest common ancestor-node will have three immediate daughters, two of which are slaves of the other; the point being that such a set of nodes could *only* be used to parse a group of three identical objects, and will be crossmodally connected to verbal subtrees

containing the syntactic apparatus necessary for saying "three *N* -s"; the sort of bimodal subtree in question is illustrated in figure 4.3 below;
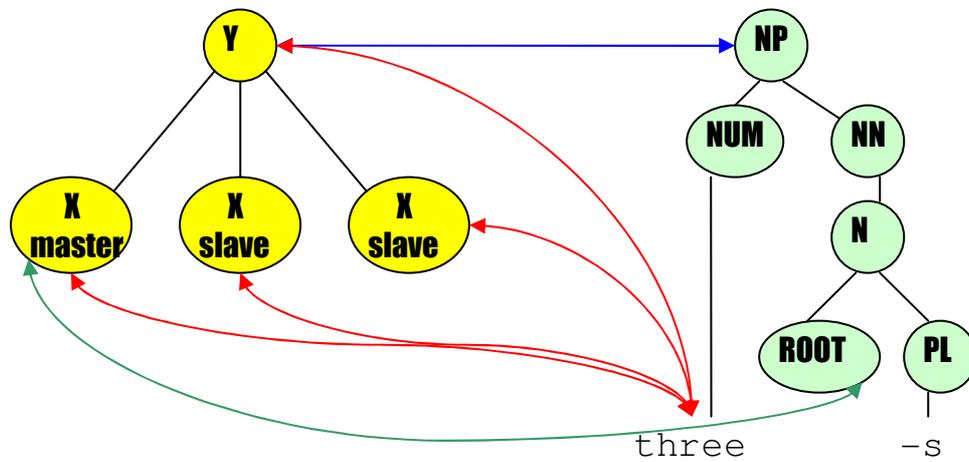
Figure 4.3; a bimodal subtree used in parsing and describing groups of three identical objects
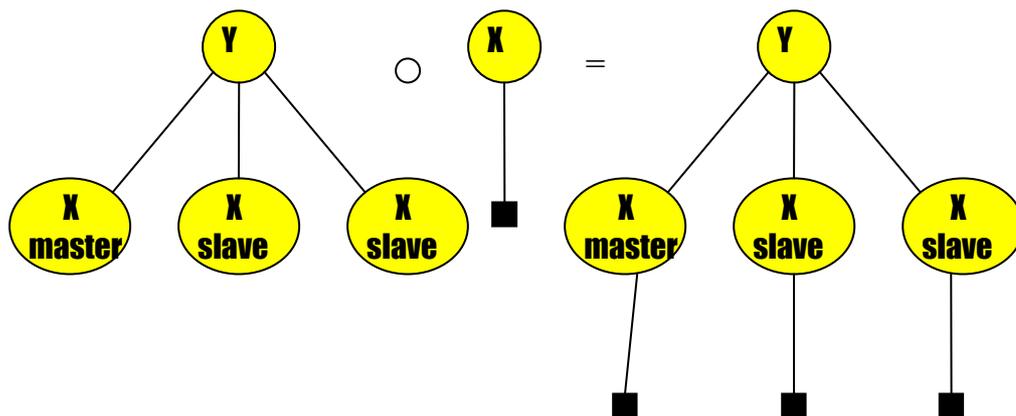
Figure 4.4; substitution at a master node.

Is this to be taken as part of Model-UDOG or not? The position here is slightly more complicated than in the previous cases, What must be made clear here is that the decision to use the visual modality as the source of meanings for the present models was purely pragmatic, and should not in any way be taken to imply a "picture theory" of meaning (Wittgenstein 1974); the particular form of the stimuli, of lines and dots, was chosen in order to allow for very simple, one-dimensional, concrete stimuli, and for the concepts required for their description to require no deeper analysis of the visual input than a surface parse. It was necessary to allow identical visual objects to

occur in (subitizable) groups, crossmodally linked to descriptions containing numeral terms, because otherwise it would not have been possible to generate a large enough set of diverse possible stimuli to train and test the systems on. However, this required a slight violation of the original premise, of sticking only to surface parses in the visual modality for meaning. However, the present models are to be seen as pilots for more sophisticated versions, capable of combining meanings from diverse cognitive modalities. The slave-master system was implemented to get the very little amount of numerical cognition required for present purposes, whilst avoiding the need to integrate into the model a full system of Data-Oriented Arithmetic/Numeration as a third cognitive modality, desirable though that may be in the long run. Probably for present considerations, it should be regarded as a technological fix – a piece of the implementation, rather than the algorithm proper. But it remains an unanswered question, if, at a later stage Data-Oriented Arithmetic/Numeration were to be modelled, how much would vision and language be implicated? Would something like the slave-master system have to be re-incorporated into surface visual processing as part of *that* model? That is as yet unknown.

Chapter 5

The Tests

Two sets of tests were conducted on both UDOG systems; a general test of their abilities to describe novel stimuli, and a "wug" test specifically geared to test their ability to generalize syntactic patterns over novel vocabulary items. The form of the tests is detailed below, and the results are given in the following chapter.

5.1 General

For the general test, a 120-item corpus was automatically generated using a java script named "CorpusMonkey"; the CorpusMonkey was loaded with four basic visual objects and their names, detailed in table 5.1 below.

Name	Form
"dot"	1 pixel
"dash"	Line of 3 pixels
"short line"	Line of 5 pixels
"long line"	Line of 10 pixels

Table 5.1; basic objects used in the general test.

The CorpusMonkey then generated all 120 possibilities for images consisting of either one group of one, two or three identical objects, paired with a description of the form "X", "two X-s" or "three X-s"; or two such groups, provided each group is comprised of different types of basic object, paired with descriptions of either the form "X to the left of Y", or "Y to the right of X". Which form of description was

employed was selected at random, with equal probabilities. For examples of real CorpusMonkey generated tree-pairs used in tests, see figure 3.2 above, or figures 5.1 and 5.2 below;
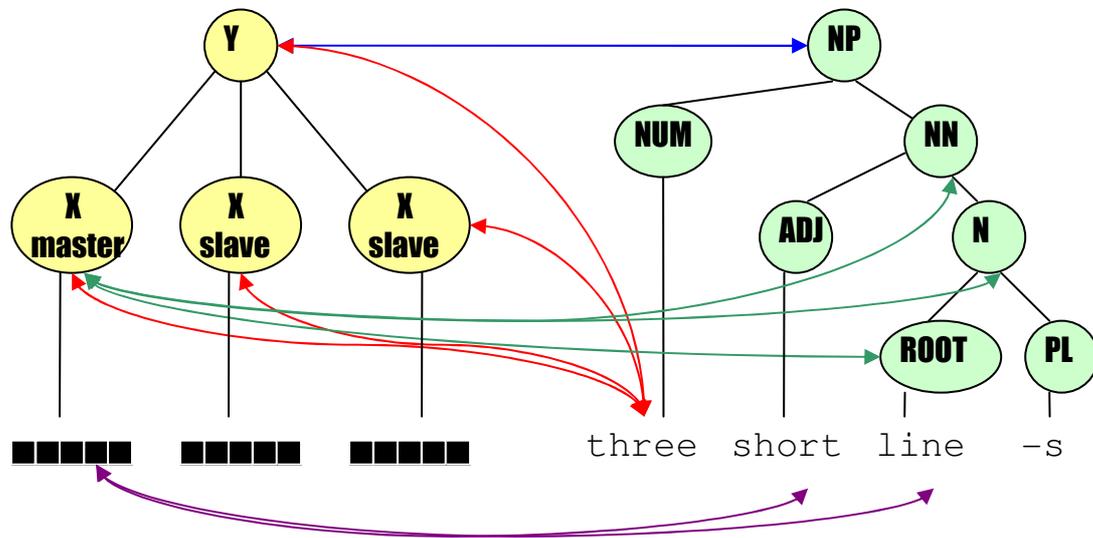
Figure 5.1 A CorpusMonkey generated tree-pair; all crossmodals are shown.
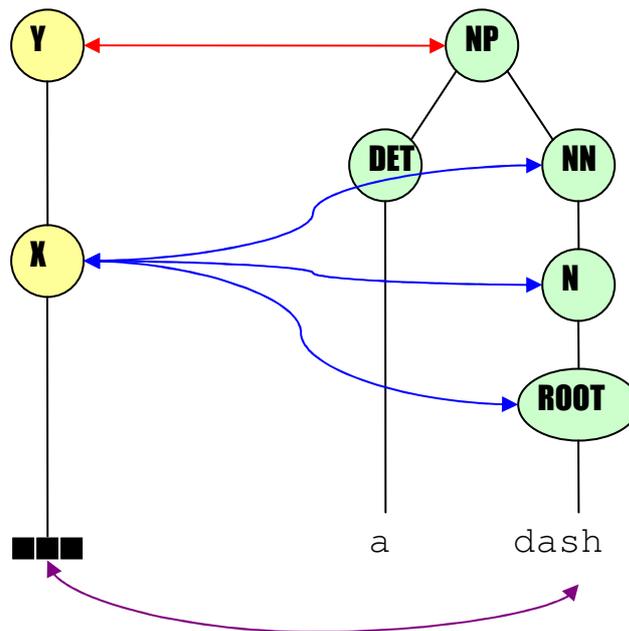
Figure 5.2 A very simple CorpusMonkey generated tree-pair; all crossmodals are shown.

Both images and descriptions were generated with annotations for tree-structure, node labels and crossmodals from templates programmed into the CorpusMonkey. These were divided into six random subdivisions, and the test was performed in six stages, in each stage using a different subdivision for test data (from which only the unanalysed, unannotated images were presented) and the remainder

were presented as training data. In the way, the models were tested on the entire dataset, all as unseen data. The purpose of this test was to assess the models' general performance on unseen data.

5.2 Wugs

A second batch of tests was run using an additional basic vocabulary item; the "wug", which was simply a seven-pixel line. Here, the training data comprised all six subsections of the main test data, plus twelve identical exemplars of the form shown in figure 5.3 below. The new word and object, "wug", was only present in the training data in these twelve exemplars.
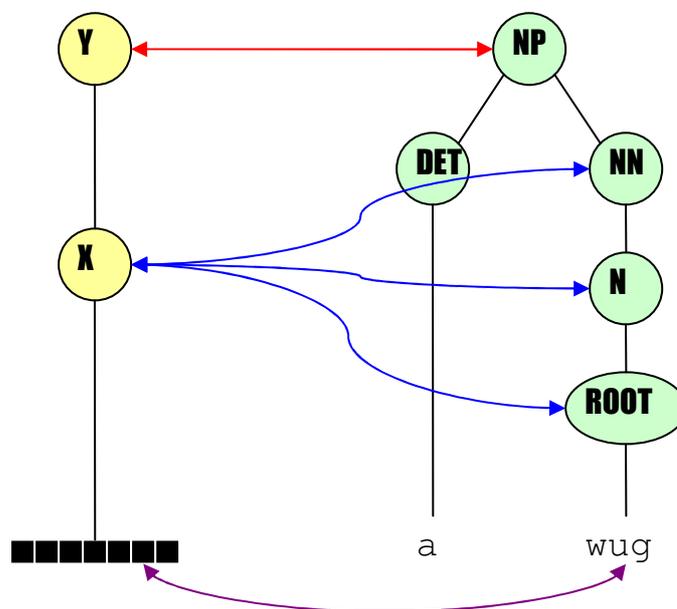
Figure 5.1; additional exemplar used in "wug" test. Note that the set of crossmodals shown here is complete.

The test data comprised 72 stimuli in which all the possible arrangements of groups of one, two or three wugs placed to the left or right of one, two or three dots, dashes, short lines or long lines. As with the general test, the test stimuli comprised only the unanalysed image, with no tree-structure or annotation. The purpose of the test was to see whether the models could generalise the syntactic patterns of the standard training data to the novel vocabulary item.

Chapter 6

Results

6.1 Measures

In both tests the output for each stimulus of both models was manually scored according to four measures. The measures used are tabulated in table 6.1. All measures were taken as percentages;

Measure	Description
Object (O)	Judged on the identification of the correct type or types of basic object are named. Because, in some outputs, the number of types named did not match with the number of types present in the stimulus, this was judged as an F1-score; that is to say, as the harmonic mean[18] of precision (the proportion of correct elements in the output) and recall (the proportion of elements in the input correctly named in the output). If an element was correctly named more than once, only the first instance was counted. Half points were awarded where; a. A dot, dash or wug preceded by "long", "short" or "ADJ". b. A long or short line was named with the correct adjective but "NN", "N" or "ADJ" instead of "line", or with "ADJ line", or "line" with no adjective.

---

[18] The harmonic mean or x and y is given by the equation $M = \dfrac{2xy}{x+y}$

Number (*N*)	Judged according to whether the named objects were correctly numbered. Again, this was expressed as an F1-score.
Relation (*R*)	If the image contained only group of same-type basic objects, or a single basic object, full marks on this measure were awarded for naming only one object type, and no "to the left of" or "to the right of" term, and zero marks are awarded otherwise. If two types of basic object are present in the stimulus, full marks are awarded if the only two types are named and the left/right relationship between them is correctly described. Partial marks are given if a correct relationship description is present but more than two tokens of object-group namings are present, because in this case the system has in such a case had extra chances to get it right.
Grammaticality (*G)*	A purely subjective measure of grammaticality and intelligibility.

Table 6.1; basic objects used in the general test.

An overall score was then given as the average of all four measures, with equal weight given to all. Where the Monte-Carlo sample was tied between more than one but fewer than twenty-one largest unifiable subsets, the average of all tied outputs was taken. Where more than twenty unified subsets were tied, a sample of twenty was taken and averaged.

To give the reader more of a feel for the measures taken, I will present two real examples of how scores were computed;

Example 1	Input = ■■■■■■■■■ ■ ■
Scores;	Output = three line to the dot of two long dot
Object	Input contains two object groups; a long line and dots. Long line is partially named, as "line", and so scores half marks. Dots, in "the dot", are correctly named. Third named group, "two long dots" is spurious. Precision is 1.5 out of 3 = 50%, recall is 1.5 out of 2 = 75%, harmonic mean = 60%
Number	Long line is erroneously numbered, dots group is erroneously numbered in second named group, "the dot", but correctly in

	third, "three long dots". Precision is 1 out of 3 = 33.33%, recall is 1 out of 2 = 50%, harmonic mean = 40%
Relation	No relation is named. 0%
Grammaticality	Phrase makes no sense. 0%
Overall	Average score = 25%

Table 6.2; scoring for a poor output; real example taken from general test output.

Example 2	Input =
Scores;	■■■■■■■■■■ ■■■■■■■■■■ ■■■■■ ■■■■■ ■■■■■ Output = three short lines to the right of two long line
Object	Short and long lines correctly identified, no spurious outputs 100% for both precision and recall, harmonic mean = 100%
Number	Short and long lines correctly numbered, no spurious outputs 100% for both precision and recall, harmonic mean = 100%
Relation	Relation correctly identified. 100%
Grammaticality	Phrase is perfectly intelligible and overall well-formed; 10% penalty for missed plural. 90%
Overall	Average score = 97.5%

Table 6.3; scoring for a good output; real example taken from general test output.

6.2 General

The results of the general test on both systems are summarised in table 6.2 below.

	Object	Number	Relation	Grammaticality	Overall
Naïve	54.48%	33.88%	22.70%	28.42%	36.62%
Binding	76.51%	71.70%	53.99%	57.60%	68.52%

Table 6.4; Performances of the naïve and binding UDOG systems on the general test.

Eyeballing the data, the overwhelming impression is that the binding system far outperforms the naïve system on all measures; overall, the binding performance is almost double the naïve, and on individual measures the binding system more than doubles the naïve score on all counts except Object, where it is still approximately

40% better. It is notable that Object is the only metric for the most part not dependent on word-ordering considerations. It is also of interest that, in comparing the three non-subjective scoring criteria (Object, Number and Relation), for both systems, the easiest, Object, was that which depended on the shortest-distance syntactic/semantic relations (between noun and adjective within an NN group, if any syntactic relation was present at all), and the hardest was that which depended on the longest-distance syntactic/semantic relationship, spanning the whole noun phrase. A 2x4 mixed-design ANOVA was conducted to test the significance of the differences in table 6.2.

	F	Significance. at p
System	67.71	<0.001
System * Scoring Criterion	11.31	<0.001
Scoring Criterion	93.79	<0.001

Table 6.5; 2x4 mixed design ANOVA

The differences between the two systems, four scoring criteria, and their interaction, were all found to be highly significant at $p < 0.001$. This finding was investigated in more detail, comparing the individual scoring criteria (within systems) using pairwise t-tests (table 6.3) and the systems performance on each scoring criterion individually using independent samples t-tests (table 6.4).

		t	Significance. at p
Naïve	Object – Number	5.92	<0.001
	Object – Relation	10.88	<0.001
	Object – Grammaticality	9.15	<0.001
	Number – Relation	6.31	<0.001
	Number - Grammaticality	5.10	<0.001
	Relation - Grammaticality	-2.62	0.01
Binding	Object – Number	-5.14	<0.001
	Object – Relation	6.51	<0.001
	Object – Grammaticality	5.45	<0.001
	Number – Relation	9.07	<0.001
binding	Number - Grammaticality	8.84	<0.001

(cont'd)	Relation - Grammaticality	-1.31	0.193

Table 6.6; Pairwise t-tests for significance of difference between types of measure

	t	Significance. at p
Object	7.143	<0.001
Number	11.124	<0.001
Relation	5.656	<0.001
Grammaticality	5.489	<0.001

Table 6.7; Independent samples t-tests, for significance of difference between systems

All differences between types of measure proved highly significant, at $p<0.001$, except for between Relation and Grammaticality, which remains significant at $p<0.05$ for Naïve UDOG, and does not attain significance for Binding UDOG. All these results were double checked using non-parametric tests (Friedman tests for the pairwise t-tests, a Kruskal-Wallis test for the independent samples t-test).

6.3 Wugs

The results of the Wug test on both systems are summarised in table 6.5 below

	Object	Number	Relation	Grammaticality	Overall
Naïve	50.65%	29.31%	8.01%	14.54%	25.62%
Binding	79.45%	94.62%	69.06%	66.00%	77.28%

Table 6.8; Results from the Wugs test

Eyeballing the data, the difference between the two systems seems to be even more marked, most notably in Relation, where Naïve UDOG performs at a fraction of its score on the general test, whereas Binding UDOG has actually improved. Indeed, the pattern is found across the board, that Naïve UDOG becomes less accurate faced with a vocabulary item for which it has no context, whereas Binding UDOG performs better than in the general test.

Theoretically speaking, what is of greatest interest here is effect of the "wug" condition on performance, as compared to the general test (or, here, the "no-wug" condition), in relation to the Relation score, since the binding of elements into correct semantic relations was quire explicitly what the binding system was formulated to do,

and to the Overall score. A 72-item random sample was taken at random from the general test dataset, so that 2x2 mixed ANOVAs could be performed, between "system" and "wugs/no-wugs", for the Relation score (table 6.6), and the Overall score (table 6.7).

	F	Significance. at p
System	237.861	<0.001
System * Wugs	6.422	0.12
Wugs	0.483	0.488

Table 6.9; 2x2 mixed design ANOVA on Overall scores

	F	Significance. at p
System	124.054	<0.001
System * Wugs	0.194	0.047
Wugs	93.79	0.66

Table 6.10; 2x2 mixed design ANOVA on Relation scores

No main effect, in either case, was found for Wugs; which is unsurprising given that the difference between wugs and no-wugs in the two systems pull in opposite directions. In both cases, significant interaction effects were found for System and Wugs, at $p < 0.05$, and highly significant results were found for System, at $p < 0.001$. The effect of the Wugs condition was investigated in greater detail using independent t-tests (table 6.8)

		t	Significance. at p
Relation	Naïve	3.446	0.01
	Binding	-2.217	0.028
Overall	Naïve	3.15	0.002
	Binding	168.395	0.027

Table 6.11; independent t-tests on the effect of the "wugs" condition on relation scores and overall scores for both models.

In all cases, the effect of the Wugs parameter is found to be significant at $p < 0.05$. It is no surprise that the naïve version suffered in the wug test; it relies wholly on the contexts given in exemplars to bind syntactic elements within semantic

relations, which it was expressly denied in the wug test. The surprising result is that that the improvement in performance in Binding-UDOG also proved significant. The implications of this and all the above results will be discussed in the next chapter.

Chapter 7

Implications

The simulations described in the preceding chapter amount to a limited test of a pilot for a larger programme of research. In cognitive science, for any genuine empirical conclusions to be drawn from a computational model, it will not suffice that the model score well on a quantitative test, or even successfully predict findings already present in the experimental literature; it must make *new* predictions, that must then be confirmed by *new* experiments. As yet, UDOG has not attained the level of sophistication necessary to make empirical predictions, and so it would be premature to draw out conclusions regarding human cognition from the present results. However, Binding-UDOG at least must be accounted a success, and a successful pilot study justifies and suggests directions for future research.
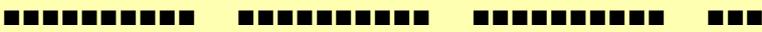
7.1 Generation

First of all, Binding-UDOG shows, for the first time, that the Data-Oriented approach can be applied to generation tasks, and that a Data-Oriented model can integrate more than one cognitive modality. However, the system is in its infancy, and it is of greater practical import to draw implications for future work out of the details of the successes and the shortcomings of both models.

Recall from §4.3 the question left open regarding the boundaries of the algorithm in relation to the implementation; the role of the crossmodals in the model. It was remarked that we may simply wish to regard the crossmodals as a technological fix for a problem of data-sparsity; that it may be that with a richer dataset, DOG would be able to solve the problem of syntactic binding (of ensuring that elements,

such as noun phrases, within a sentence or phrase are bound into the position corresponding to their semantic role) on the strength of the statistical regularities of the training data alone. On the other hand, it may be that we wish to treat them as a central feature of the algorithm proper – that DOG in fact cannot adequately model compositional language without the level of coupling between signifying and signified exemplars found in UDOG, wherein substitution operations are of complexes of crossmodally connected root-nodes at crossmodally connected complexes of leaf-nodes, rather than of single substitutions of roots and leaves. It should by now be clear from the vastly superior performance of Binding compared to Naïve UDOG that the latter is the correct answer. The Wug-Test, in particular, was especially designed to probe the two systems' ability to handle this problem; that Naïve-UDOG performed significantly worse, while Binding-UDOG in fact performed significantly better, serves for as clear an indication as could be hoped for that the overall difference in their performance is substantially accounted for by Naïve-UDOG's inability to handle binding. However, it remains to be seem whether a Naïve approach might be adequate for modelling interactions between *non*-linguistic cognitive modalities.

Although Binding-UDOG's scores are eminently satisfying for a first pass at a novel algorithm, there is much room for improvement, as a such, it would be useful to examine in detail they system's most common major error, shown in table 7.1 below:

	Input =
Scores;	■■■■■■■■■ ■■■■■■■■■ ■■■■■■■■■ ■■■
	Output = three lines to the dash of three lines
Object	"Dash" and correctly identified, "long lines" partially identified as "lines", plus one spurious output. 1.5 out of 3 = 50% for precision and 1.5 out of 2 = 75% for recall; harmonic mean = 60%
Number	Both non-spurious elements correctly numbered. 2 out of 3 = 66.67% for precision and 2 out of 2 = 100% for recall, harmonic mean = 80%
Relation	No relation identified. 0%

Grammaticality	Makes no sense. 0%
Overall	Average score = 35%

Table 7.1; An example of the most common error of Binding-UDOG; taken from General Test run.

This type of error, of the form "*Concrete-NP(X)*[19] to the *Concrete-NP(Y)* of *Concrete-NP(X)*", accounts for over 90% of those outputs made by Binding-UDOG on the General Test which scored less than 50% overall. It cannot be dismissed as an artefact of the unification process, whereby no derivations contain two instances of *Concrete-NP(X)*, but the unification process combines trees with the *Concrete-NP(X)* in either leftmost of rightmost position and an incomplete branch on the other side to make trees with *Concrete-NP(X)* on both sides; 11 derivations were found in the 500 that generated the above output with an NP containing "line" in both positions, as in the following example;

DET long line to *DET* *N* of a long line

It is worth noting that, given the form of trees generated by the CorpusMonkey script, in any bimodal tree where two groups of objects are in a left-of/right-of relation, as in figure 3.2, one of the visual tree nodes (directly above one of the groups of same-type objects or the other will be crossmodally linked to three of the five nodes in the verbal tree labelled "NP". Figure 7.1 shows the same tree-pair as figure 3.2, but with a different set of crossmodals shown.

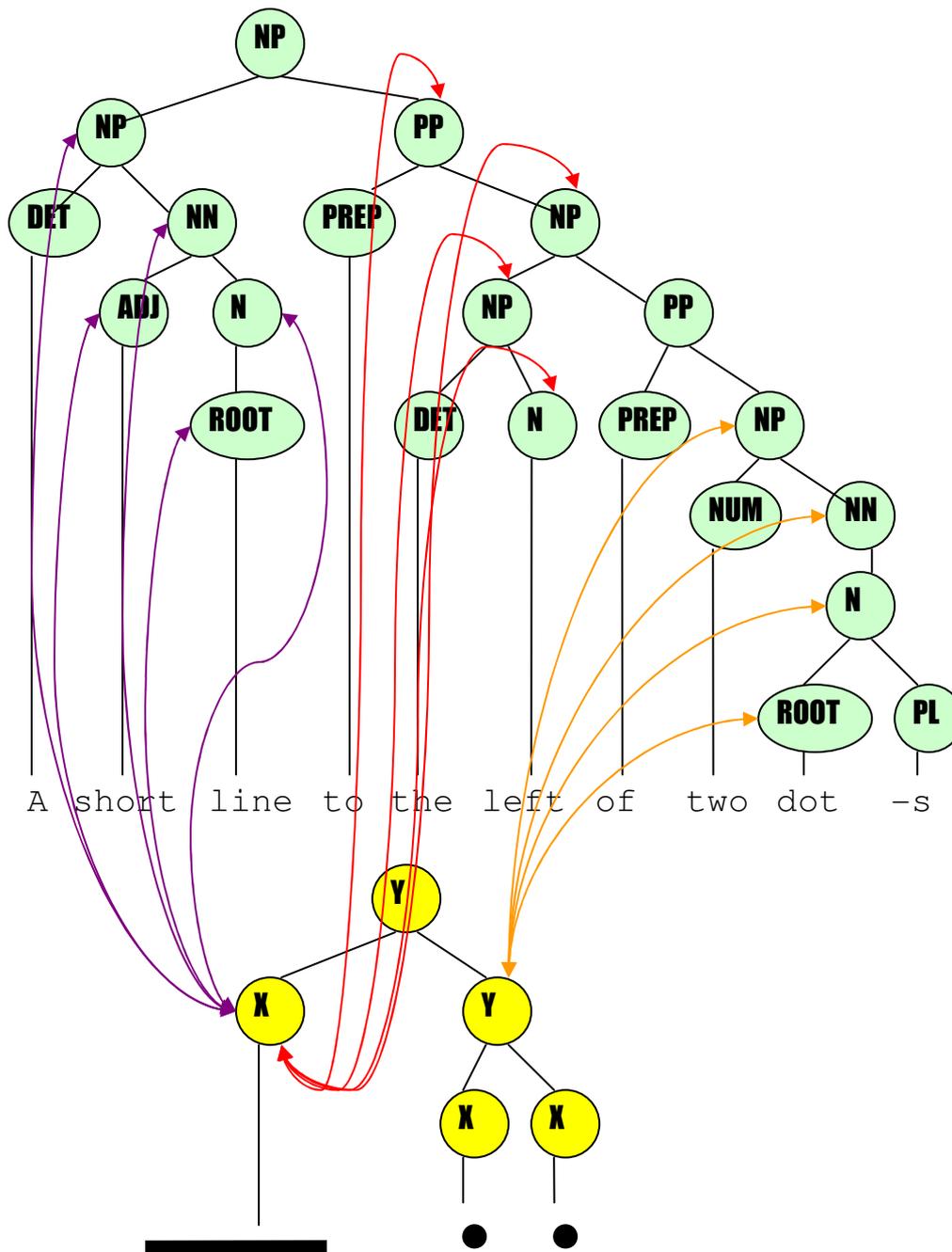[19] That is to say, an NP of the form DET/NUM (ADJ) dot/dash/line (PL).

Figure 7.1: Tree-Pair taken from the General Test Treebank, with an incomplete set of crossmodals shown.

Note that the NP nodes over "the left" and "the left of two dots" cannot both form head-nodes of verbal subtrees in a single bimodal subtree according to the wellformedness criteria stipulated for Binding UDOG in §3.3 (criterion 5, specifically), whereas either can co-occur with the NP node over "a short line" as head-nodes of verbal subtrees in a bimodal subtree, as in figure 7.1:
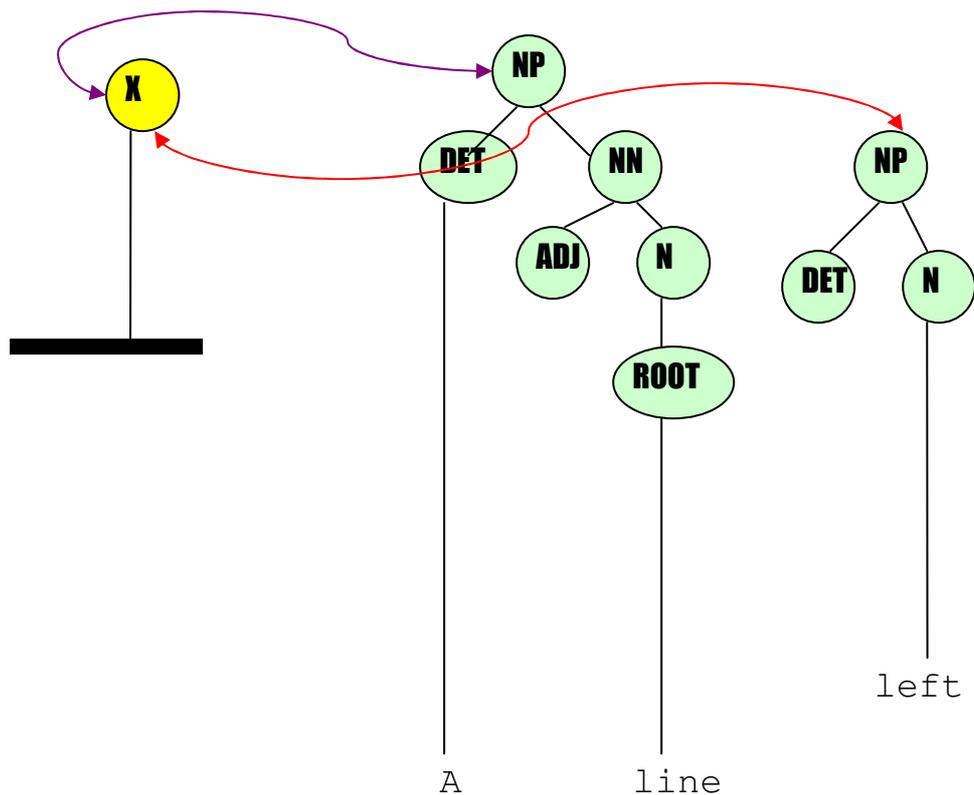
Figure 7.2: A legal bimodal subtree of Binding-UDOG of the tree-pair in figure 7.1, with two verbal subtrees headed with NP nodes.

Note that where Binding-UDOG produces a bimodal subtree like that in figure 7.2, it has no way to determine which of the two NP-labelled substitution sites should receive which verbal subtree, and must select randomly. One possible solution to this problem would be to attach labels to crossmodals, so that those that correspond to concrete reference relations (in figure 7.2, the purple crossmodal) and those that relate to part-whole relations (the red crossmodal) are distinguished, guaranteeing that a verbal subtree the root of which is joined to the root of the visual subtree by a concrete crossmodal can only be substituted at a substitution site joined to the visual substitution site by a crossmodal of the same kind.

7.2 Object-naming in the one-word stage

One outcome of the tests performed on the two UDOG systems was that Binding UDOG would actually perform better on the Wug Test than on the General Test; I had

instead expected that it would either show no significant effect, or that its performance would be decremented, either to the same degree as the Naïve system, indicating that crossmodals should be seen as a technological fix for sparse data, or to a significantly less degree, indicating that crossmodals should be seen as an essential feature of the algorithm. However, it is easy to figure out just how the Wug condition helped Binding UDOG along. One common type of error is illustrated in a real example taken from Binding-UDOG's performance on the General Test, in table 7.2 below;

	Input = ■■■■■ ■ ■ ■
Scores;	Output = a dot *PL* to the right *PREP* a short line to the line of two *NN*
Object	"Dot" and "short line" correctly identified, plus two spurious outputs 2 out of 4 = 50% for precision and 2 out of 2 = 100% for recall, harmonic mean = 66.67%
Number	One short correctly numbered, three dots not correctly numbered. 1 out of 4 = 25% for precision and 1 out of 2 = 50% for recall, harmonic mean = 33.33%
Relation	Relation correctly identified; score halved, however, because the output contains two potential loci for relation terms. 50%
Grammaticality	First half is more or less sensible and grammatical, second half is gibberish; 10% penalties plural marker ad preposition in first half left unrealised. 30%
Overall	Average score = 45%

Table 7.2; real example of common error from the output of Binding-UDOG on the General Test

What has happened here is that structure for the relation expression has been imported into the output from two separate sources; one coming with material contributing to the description of the single short line, the other coming with the what I presume to be an abortive attempt at describing two or the three dots; in both cases, the object-and-number describing material came bound up with relation-describing material, and these together caused a confused and ill-formed output. In the Wug Test, the description of the wug cannot come with such extraneous material, since the only

exemplars associating the word "wug" with images of wugs contain nothing more than a single wug, described as "a wug" (see figure 5.1).

This suggests an interesting hypothesis regarding First Language Acquisition, to be followed up if further work on UDOG proves successful. Binding UDOG benefits notably from having access to isolated examples or words paired with their referents. Bates, Bretherton and Snyder (1988) outline a "two-strand" theory of individual differences in First Language Acquisition, wherein two main learning strategies employed by infant language learners; "Strand two" is characterised by slow vocabulary growth and a tendency towards holophrases in which multi-word utterances are used as unanalysed wholes, but of greater interest here is "Strand one". Below is Bates *et al*'s (ibid.) full tabulation of the key features of "Strand one" semantic learning;

- High proportion of nounsin first 50 words
- Single words in early speech
- Imitates object names
- Greater variety within lexical categories
- Meaningful elements only
- High adjective use
- Context-flexible use of names
- Rapid vocabulary growth

Bates *et al*, ibid.

If some mechanism like Binding UDOG does indeed form the basis of human linguistic production, might it be that the comparatively rapid vocabulary learning of "Strand one" learners, and their ability to use names context-flexibly, owes to their creation of exemplars of a noun linked to its referent, isolated from context, just like the "wug" exemplars in the Wug Test in §§5.2 and 6.3, which are then available to the child as part of her exemplar-base. This suggests a direction for the empirical testing of the UDOG model against human subjects.

7.3 Crossmodals, network structure and access consciousness

One major direction for future UDOG research will be to expand the model to encompass multiple cognitive modalities, so that a single linguistic output can bind together meanings drawn from a diversity of modalities which approximates the "saturatedness" of real human language use in naturalistic conditions. I wish to, rather speculatively, draw out one possible consequence if such developments of the model were to prove successful.

Block (1995) distinguishes two understandings of consciousness; "Access-Consciousness" (A-Consciousness), characterised as the availability of cognitive content for report, reasoning and the control of behaviour, and "Phenomenal Consciousness" (P-Consciousness); the qualitative "what-it's-like" of experience, which may well be, as Chalmers (1996) postulates, beyond the reach of scientific investigation altogether, or else is, as Dennett (1991) holds, strictly reducible to A-Consciousness. Disavowing any consideration of the reducibility or otherwise of P-Consciousness, it may be that "saturated" UDOG models of the type suggested above may offer a the basis of a novel theory of A-Consciousness, and the role played therein by language, or rather, by particular exemplars in the language modality.

One of the most interesting developments in Graph Theory in the last decade, with applications in as diverse fields as Physics, Urban Planning, Genetics, Neuroscience and Sociology, is the theory of *Small World Networks* (Watts and Strogatz 1998). A Small World Network is a random graph in which the considerable majority of nodes have only local connections (which in network terms, means only having connections where, if x is connected to y and y is connected to z, there is a high probability that x will also be connected to z), but there exist a small number of "supernodes" that have very many of non-local connections. The consequence of this network structure is that it is possible to go from any node in the network to any other in a small number of moves. This of course is not the first time that an application of Small World Networks to cognition and consciousness has been thought of; see Roxin, Reicke and Solla (2004), for example. What I do wish to offer as novel is the suggestion of a network specifically of exemplars, connected intramodally at potential substitution sites, but also crossmodally. Doubtless crossmodal connections also exist

between non-linguistic exemplars (between vision and motor control for instance), but what is unique to human consciousness is the role played by linguistic exemplars as supernodes, giving us a more integrated form of A-Consciousness than any other species. What I am proposing here is not a psycholinguistics-style boxes-and-arrows diagram with the "language box" in the middle, but rather a decentralised network in which concrete exemplars across all modalities of cognition are joined up, mostly by local connections, but with a population of supernodes which join up exemplars from many modalities, and the majority of these supernodes happen to be linguistic exemplars.

Chapter 8

Conclusions

The achievement of the model itself is small, but what it has shown to be possible – generation and multimodal integration under a Data-Oriented framework, represent considerable advances for Data-Oriented approaches to Cognitive Science and Artificial Intelligence. On the webpage for the his new Cognitive Systems research group at the University of St. Andrews[20], Bod (2006a) proposes the goal of the new group to be "to develop one system that *unifies* different modalities" (author's emphasis); certainly the models of language, music and reasoning in Bod (2005) show that unimodal DOP models can be used to unify cognitive modalities under a single *formalism*; but the programme of multimodal Data-Oriented research that the present work warrants, *if successful*, offers a way to *integrate* different modalities within a single *model*.

---

[20] Which I will be joining in September 2006.

References

Bates, E., I.Bretherton, & L. Snyder (1988). *From first words to grammar: Individual differences and dissociable mechanisms*. New York: Cambridge University Press.

Bedford, F. (1997). "False categories in cognition: The Not-the-Liver fallacy. Cognition", 64, 231-248.

Blackburn, S. (1984). Spreading the Word: Groundings in the Philosophy of Language. Oxford: Oxford University Press.

Bod, R. (1992). "A Computational Model of Language Performance; Data-Oriented Parising". *Proceedings COLING-92*, Nantes, France

Bod, R. (1998), *Beyond Grammar; An Experience-Based Theory of Language*, Stanford, CA: Centre for the Study of Language and Information.

Bod, R (2002) "A Unified Model of Structural Organization in Language and Music" *Journal of Artificial Intelligence Research*, 17, 289-308.

Bod, R. (2003a), An Efficient Implementation of a New DOP Model Proceedings EACL'03, Budapest, Hungary

Bod, R. (2003b). "Do All Fragments Count?" *Natural Language Engineering*, 9:307-323.

Bod, R. (2004). "Exemplar-Based Explanation." Proceedings, *Computation and Philosophy* (ECAP'04), Pavia, Italy.

Bod, R. (2005). Towards Unifying Perception and Cognition: The Ubiquity of Trees. Prepublication.

Bod, R. (2006a). "Cognitive Systems Group: The DOP Approach to Language and Cognition", http://cogsys.dcs.st-and.ac.uk/ (Accessed 28th August 2006).

Bod, R. (2006b). "Unsupervised Parsing with U-DOP", paper given at CONLL 2006, New York, NY.

Bod, R., R. Bonnema and R. Scha, (1996). "A Data-Oriented Approach to Semantic Interpretation." *Proceedings Workshop on Corpus-Oriented Semantic Analysis,* ECAI-96, Budapest, Hungary.

Bod, R. and R. Kaplan, (1998). "A Probabilistic Corpus-Driven Model for Lexical-Functional Analysis", *Proceedings COLING-ACL '98*, Monteal, Canada.

Bod, R., R. Scha and K. Sima'an, (eds.) (2003). *Data-Oriented Parsing*. Stanford, CA: Centre for the Study of Language and Information.

Briscoe, E. (2002) "Grammatical Acquisition and Linguistic Selection", in Ted Briscoe, editor, *Linguistic Evolution through Language Acquisition: Formal and Computational Models*. Cambridge: Cambridge University Press.

Brooks, R. (1997). "Intelligence without representation", in Haugeland (1997).

Chalmers, D. (1996). *The Conscious Mind: In Search of a Fundamental Theory*. New York: Oxford University Press.

Chen, H., D. Cochran, I. Hanafusa, C. Laskowski, K. Ludke, and M. Ntarila, (2005) "Statistical Learning". Presentation for MSc module *Psychology of Language Learning*, University of Edinburgh, 22nd November 2005.

Clark, A. (1991), *Microcognition*. Cambridge: The MIT Press.

Cochran, D. (2005), "Using Stochastic Tree-Substitution Grammar in Iterative Learning Simulations as a way of approaching issues in Diachronic Syntax", paper given at the College of Arts and Social Sciences Postgraduate Conference at the University of Aberdeen on 23rd June 2005.

Cochran, D., (2006a), "Data-Oriented Picture Parsing", Unpublished paper available at; http://www.ling.ed.ac.uk/~s0454279/

Cochran, D. (2006b), "Non-Reductive Neutral Monism", Work In Progress Seminar given in the Department of Philosophy, University of Edinburgh, 24th May 2006.

Coleman, J. and J. Pierrehumbert (1997) "Stochastic Phonological Grammars and Acceptability", *3rd Meeting of the ACL Special Interest Group in Computational Phonology: Proceedings of the Workshop*, 12 July 1997. Association for Computational Linguistics, Somerset NJ. 49-56.

Dennett, D. (1991). *Consciousness Explained*, Boston: Little & Company.

Dennett D. (1995). Darwin's *Dangerous Idea: Evolution and the Meaning of Life*. Penguin Books, London.

Dreyfus, H. (1997), "From Micro-Worlds to Knowledge Representation: AI at an Impasse", in Haugeland (1997).

Feldman, J, G. Lakoff, D. Bailey, S. Narayanan, T. Regier and A. Stolcke (1996) "L_0 – The first five years of an automated language acquisition project." *Artificial Intelligence Review*, 10:114-125.

Feldman, J, G. Lakoff, D. A. Stolcke and S. Weber (1990). "Miniature Language Acquisition: A touchstone for cognitive science". In *Proceedings of the 12th Annual Conference of the Cognitive Science Society*, pp.686-93, MIT: Cambridge, Mass.

Field, R. &. R. Noyes, (1974). "Oscillations in chemical systems. IV. Limit cycle behavior in a model of a chemical reaction". *Journal of Chemical Physics*, 60:1877-1884.

Goodman, J., (2003). "Efficient parsing of DOP with PCFG-reductions". In Bod, Scha and Sima'an 2003, pp. 125-146.

Gould, S (1997). "Evolution: the Pleasures of Pluralism". *New York Review of Books*, June 26, 1997.

Gould, S..and R. Lewontin, (1979). "The spandrels of San Marco and the panglossian paradigm: a critique of the adaptationist programme". *Proceedings of the Royal Society of London, Series B*. Vol. 205, pp. 581-598.

Hartmann, S. (1996). "The World as a Process: Simulation in the Natural and Social Sciences." In R. Hegselmann, U. Muller, and K. Troitzsch, eds., *Modelling and Simulation in the Social Sciences from the Philosophy of Science Point of View*, 77–100. Dordrecht: Kluwer Academic.

Haugeland, J. (ed.) (1997), *Mind Design II: Philosophy, Psychology and Artificial Intelligence*. Cambridge: The MIT Press.

Hoogweg, L., (2000), *Extending DOP1 with the insertion operation*. Master's thesis, University of Amsterdam, Amsterdam.

Hubel, D. and T. Wiesel, (1963), "Receptive fields of cells in striate cortex of very young, visually inexperienced kittens." *Journal of Neurophysiology*, 26:994-1002..

Hubel, D. and T. Wiesel, (1965) "Receptive fields and functional architecture in two nonstriate visual areas of the cat." *Journal of Neurophysiology*, 28:229-289.

Hurford, J. (2000). "Social transmission favours linguistic generalization", in Chris Knight, James R. Hurford and Michael Studdert-Kennedy, editors, *The Evolutionary Emergence of Language: Social Function and the Origins of Linguistic Form*, pages 324-52. Cambridge: Cambridge University Press.

Karp, R. (1972). "Reducibility Among Combinatorial Problems". *Proceedings of a Symposium on the Complexity of Computer Computations*.

Kersten, D. (2000). "High-level vision as statistical inference". In Gazzaniga , S. (Ed.), *The New Cognitive Neurosciences*, Cambridge, The MIT Press.

Kirby, S. (1999). Learning, Bottlenecks and Infinity: a working model of the evolution of syntactic communication. In K. Dautenhahn and C. Nehaniv, editors, *Proceedings of the AISB'99 Symposium on Imitation in Animals and Artifacts*. Cambridge, Mass. & London : MIT Press.

Krohs, U. (2006). "A priori measurable worlds". Paper presented at *Models and Simulations*, CNRS, Paris, 12th-13th June 2006.

Levelt, W (1989). *Speaking: From Intention to Articulation*. Cambridge, Massachusetts: MIT Press.

Maia, M (2000). Evidentiality processes in Karaja. Published online at http://www.museunacional.ufrj.br/linguistica/membros/maia/publicacoes.htm

Manning, C. and H. Schütze, (1999). *Foundations of Statistical Natural Language Processing*. Cambridge, The MIT Press.

Minsky, M. and S. Papert (1970). Draft of a proposal to ARPA on artificial intelligence at MIT, 1970-71.

Redhead, M. (1980). "Models in Physics". *British Journal for the Philosophy of Science*, 31:145-163.

Rohrlich, F., (1991) "Computer Simulation in the Physical Sciences". In A. Fine, M. Forbes and L. Wessels (eds.), *PSA 1990*, Vol. 2, 507-518, East Lansing.

Roxin A., H. Riecke, S. Solla (2004) "Self-sustained activity in a small-world network of excitable neurons". *Physical Review Letters* 92:198101.

Scha, R. (1990). "Taaltheorie en Taaltechnologie: Competence en Performance", in Q. de Kort and G. Leerdam (eds.), *Computertoepassingen in de Neerlandistiek*, Almere: Landelijke Vereniging van Neerlandici (LVVN-jaarboek).

Sima'an, K. (1996). "Computational Complexity of Probabilistic Disambiguation by means of Tree Grammars", in *Proceedings COLING-96*, Copenhagen, Denmark.

Sima'an, K. (1999). *Learning Efficient Disambiguation*. ILLC Dissertation Series 1999-02, Utrecht University/University of Amsterdam, The Netherlands.

Tu, Z., X. Chen, A Yuille and S-C Zhu, (2005) "Image Parsing: Unifying Segmentation, Detection and Recognition". *International Journal of Computer Vision*. 63:113-140.

Van den Berg, M., R. Bod, and R. Scha, (1994). "A Corpus-Based Approach to Semantic Interpretation", *Proceedings Ninth Amsterdam Colloquium*, Amsterdam, the Netherlands.

Von der Heydt R (2003) Image parsing mechanisms of the visual cortex. In: *The Visual Neurosciences* (J. Werner and L. Chalupa, eds), pp 1139-1150. Cambridge, Mass.: MIT press.

Watts, D. and S. Strogatz (1998). "Collective dynamics of 'small-world' networks". *Nature* 393:440-442.

Way, A (1999). "A Hybrid Architecture for Robust MT using LFG-DOP". *Journal of Experimental and Theoretical Artificial Intelligence* 11:459-473.

Wikipedia contributors (2006). "Clique problem," *Wikipedia, The Free Encyclopedia*, http://en.wikipedia.org/w/index.php?title=Clique_problem&oldid=70700523 (accessed August 23, 2006).

Wittgenstein, L., (1969). Preliminary studies for the Philosophical Investigations, generally known as the Blue and Brown Books, 2nd edition. Oxford; Basil Blackwell.

Wittgenstein, L., (1974). *Tractatus Logico-Philosophicus*. London: Routledge & Kegan Paul.

Zipf, G. (1949). "Human Behaviour and the Principle of Least-Effort". Cambridge Mass.: Addison-Wesley.

Zuckerman. D. (1993) "NP-Complete Problems have a version that is hard to Approximate". *Proceedings of the Eighth Annual Conference on Structure in Complexity Theory*.