A Psychologically Plausible and Computationally Effective Approach to Learning Syntax

Stephen Watkinson and Suresh Manandhar,
Department of Computer Science,
University of York,
York YO10 5DD,
UK.

Abstract

Computational learning of natural language is often attempted without using the knowledge available from other research areas such as psychology and linguistics. This can lead to systems that solve problems that are neither theoretically or practically useful. In this paper we present a system CLL which aims to learn natural language syntax in a way that is both computationally effective and psychologically plausible. This theoretically plausible system can also perform the practically useful task of unsupervised learning of syntax. CLL has then been applied to a corpus of declarative sentences from the Penn Treebank (Marcus et al., 1993; Marcus et al., 1994) on which it has been shown to perform comparatively well with respect to much less psychologically plausible systems, which are significantly more supervised and are applied to somewhat simpler problems.

1 Introduction

Computational learning of natural language can be considered from two common perspectives. Firstly, there is the psychological perspective, which leads to the investigation of learning problems similar to those faced by people and the building of systems that seek to model human language learning faculties. Secondly, there is the computational perspective, which seeks to build systems that effectively solve language learning problems that are of practical importance.

In principle, there is significant overlap between these two perspectives. The most common language learning problems that we wish to solve computationally are frequently those that humans have to solve. For example when humans learn language, especially syntax, it seems to be in a mostly unsupervised setting i.e. there is no annotation of training examples. From a computational perspective, while there are some annotated resources available, in general we have very large amounts of unannotated text available from which we desire to be able to extract grammars, meaning etc. Given this overlap, it seems wise to investigate what we know of the human approach, as humans are good at solving these problems.

In this work we present a system for learning syntax that seeks to maintain both the psychological and computational perspectives. We also show that this is an effective way to build natural language learning systems. We represent the syntactic knowledge using the Categorial Grammar (CG) formalism, so in Section 2 we introduce CG. In Section 3 we aim to define the problem that is to be solved in a way that is psychologically plausible. This is followed in Section 4 by the description of CLL a computational effective solution to the problem, which we maintain is also reasonably psychologically plausible. Related work is discussed in Section 5. The results of experiments using CLL on examples from the Penn Treebank are presented in Section 6 and we draw some conclusions from this work in Section 7.

2 Categorial Grammar

Categorial Grammar (CG) (Steedman, 1993; Wood, 1993) provides a functional approach to lexicalised grammar, and so, can be thought of as defining a syntactic *calculus*. Below we describe the basic (AB) CG.

There is a set of *atomic* categories in CG, which are usually nouns (n), noun phrases (np) and sentences (s). It is then possible to build up *complex* categories using the two slash operators "/" and "\". If A and B are categories then A/B is a category and A\B is a category. With basic CG there are just two rules for combining categories: the forward (FA) and backward (BA) *functional application* rules.

$$\begin{array}{lcll} X/Y\ Y & \Rightarrow & X & \qquad\qquad (FA) \\ Y\ X\backslash Y & \Rightarrow & X & \qquad\qquad (BA) \end{array}$$

In Figure 1 the parse derivation for "John ate the apple" is presented, which shows examples of the types of categories that words can take and also how those categories are combined using the application rules.

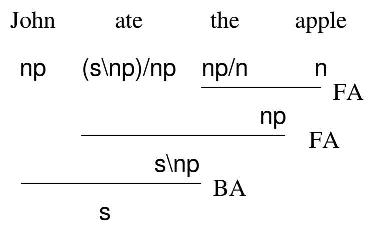

Figure 1: A Example Parse in Pure CG

Categorial grammar does not handle compound noun phrases very well, so we have added some simple combination rules that allow the possibility of joining adjacent nouns and noun phrases.

Perhaps the main advantage of using a lexicalised formalism such as CG for this task is that the learning of the grammar and the learning of the lexicon is one task. CG will also easily allow extensions such that new categories could be generated or that category schema could be used.

3 A Plausible Problem

The desire in this work, is to show that a computationally effective system, in this case CLL, can be built in such a way that both the problem it solves and the way it is implemented are psychologically plausible. We would also suggest that defining the problem in this way leads to a practically useful problem being attempted.

Initially we seek to define the problem in a psychologically plausible way. The aim is to induce a broad coverage grammar for English from a set of appropriate examples. Beyond this, the problem can to some extent be defined by the knowledge the learner already has; the information that is available in the environment and the knowledge which is to be learned. Psychology and psycholinguistics provide us with a significant amount of data from which we may derive a fairly good picture of how the problem is defined for humans. In particular, we will concentrate on a child's acquisition of their first language and how this relates to a computational model, as this seems to be the point at which human language acquisition is at its most efficient.

3.1 The Environment

With respect to the environment in which a child learns, we will concentrate on two questions.

1. What examples of language are children exposed to?

2. What kind of language teaching do children experience?

It is clear that children experience positive examples of syntax i.e. all the language utterances they hear, although these may be somewhat noisy (people make lots of mistakes). Children do not, however, experience negative examples, as people do not (at least in any consistent way) present ungrammatical examples and mark them as incorrect.

¿From a syntactic perspective, examples appear to have little discernible annotation. Pinker (Pinker, 1990) summarises what seems to be the only evidence that children receive structural information. It is suggested that structural information may be obtained by the infant from the exaggerated intonation which adults use when talking to children. While there may be a link, it is not clear what it is and it is certain that complete structures for sentences cannot be considered to be available, as there is not enough information in intonation alone.

Hence, we have defined a learning setting that is both positive examples only and unsupervised. However, there has been some suggestion that negative evidence may be available in the form of parental correction. This leads to issues of language teaching.

It is suggested that the language presented to children is in fact very detailed and structured. The *motherese hypothesis* or *child directed speech* (Harley, 1995; Pinker, 1990; Atkinson, 1996), proposes that, starting with very simple language, adults gradually increase the complexity of the language they use with children, such that they actually provide children with a structured set of language lessons. The theory is based upon research that shows that adults use a different style of speech with infants than with other adults (Snow and Ferguson, 1977).

However, Pinker (Pinker, 1990) provides arguments against the acceptance of the Motherese hypothesis. Firstly, although it may appear that the language is simplified, in fact the language used is syntactically complex – for example it contains a lot of questions. Secondly, there exist societies where children are not considered worth talking to until they can talk. Hence, there is no motherese and only adult-to-adult speech examples which infants hear and from which they have to acquire their language. These children do not learn language any slower than the children who are exposed to motherese. Atkinson (Atkinson, 1996) provides further arguments against the motherese hypothesis, suggesting that making the input simpler would make learning more difficult. For the simpler the input is, the less information is contained within it and so there is less information from which to learn.

An alternative suggestion for the provision of teaching is that negative evidence is actually available to the child in the form of feedback or correction from parents. This model was tested by Brown and Hanlon (Brown and Hanlon, 1979) by studying transcripts of parent-child conversations. They studied adults responses to childrens' grammatical and ungrammatical sentences and could find no correlation between children's grammatical sentences and parent's encouragement. They even found that parents do not understand children's well-formed questions much better than their ill-formed questions. Pinker (Pinker, 1990) reports that these results have been replicated. This can only lead to the conclusion that there is no significant negative evidence available to the infant attempting to learn syntax.

Hence, we have a learner that is unsupervised, positive only and does not have a teacher. In practice this means that we build a system that learns from an unannotated corpus of examples of a language (in this case we use unannotated examples from the Penn Treebank) and there is no oracle or teacher involved.

3.2 The Learner's Knowledge

A child can be considered to have two types of knowledge to bring to the problem. Firstly there may be some innate knowledge that is built into the human brain, which is used in determining the language learning process. Secondly, there is knowledge that the child has already acquired.

The issue of a child's innate knowledge has been the subject of a significant debate, which we do not have the space to do justice to here. Instead we will present the approach that we will take and the reasons for following it, while accepting that there will be those who will disagree.

The *poverty of stimulus* argument (Chomsky, 1980; Carroll, 1994) suggests that the environment simply does not provide enough information for a learner to be able to select between possible grammars. Hence, it seems that there needs to be some internal bias. Further evidence for this is the strong similarity between natural languages with respect to syntax, which has led Chomsky to hypothesise that all humans are born with a *Universal Grammar* (Chomsky, 1965; Chomsky, 1972; Chomsky, 1986) which determines the search space of possible grammars for languages. This is supported further by the *Language Bioprogram Hypothesis* (LBH) of Bickerton (Bickerton, 1984), who analysed creoles, the languages that develop in communities where different nationalities with different languages work alongside each other. Initially, in such contexts, a pigeon develops, which is a very limited language that combines elements of both languages found in the community. The pigeon has very limited syntactic structures. The next generation develops the pigeon into a full language – the creole. Bickerton (Bickerton, 1984) found that the creoles, developing from syntactically impoverished language examples as they do, actually contain syntactic structures not available to the learners from their pigeon environment. These structures show a strong similarity to the syntactic structures of other natural languages. Bickerton (Bickerton, 1984) states:

> "the most cogent explanation of this similarity is that it derives from the structure of a species-specific program for language, genetically coded and expressed, in ways still largely mysterious, in the structures and modes of operation of the human brain."

Practically, there are a variety of options for providing a suitable level of innate knowledge. By choosing a lexicalised grammar (see Section 2) we have allowed the system to have a few basic rules for word combination and a set of possible categories for words. Currently, the use of a complete set of possible lexical categories is perhaps too strong a bias to be psychologically plausible. In future we will look at either generating categories, or using category schemas, both of which might be more plausible.

The second type of knowledge available to the learner is that which has already been learned. We can, to some extent, determine this from developmental psychology. Before the stage of learning syntax children have already learned a wide variety of words with some notion of their meaning (Carroll, 1994). They then seem to be beginning to use single words to communicate more than just the meaning of the word (Rodgon, 1976; Carroll, 1994) and then they begin to acquire syntax.

In terms of a learning system this would suggest the availability of some initial lexical information like word groupings or some bootstrapping lexicon. Here we present results using a system that has a small initial lexicon that it is assumed that the child has learned. We are also investigating using word grouping information.

3.3 What is to be learned?

Given the knowledge that is available to the learner and the environment from which the learner receives examples of the language, the learner is left with the task of learning a complex, i.e. lexicalised, lexicon.

Using CG means that we are aiming to build a lexicon that contains the required CG category or categories for each word, which defines the syntactic role or roles of that word. In future, we may look at extending the grammar to include more detail, so that the syntactic roles of words are defined more accurately.

Interestingly, this leads us to a practically interesting problem. Given the amount of unannotated text available for a variety of different languages and for a variety of different domains, it would be very useful to have a system that could extract grammars from selections of such text.

4 A Computationally Effective Solution

The system we have developed is shown diagrammatically in Figure 2. In the following sections we explain the learning setting and the learning procedure respectively.

4.1 The Learning Setting

The input to the learning setting has five parts, which are discussed below.

The Corpus The corpus is a set of positive examples represented in Prolog as facts containing a list of words e.g.

```
ex([mary, loved, a, computer]).
```

The Lexicon The lexicon is initially empty, apart from a small set of closed-class words used to bootstrap the process, as this is what the learner induces. It is stored by the learner as a set of Prolog facts of the form:

```
lex(Word, Category, Frequency).
```

Where `Word` is a word, `Category` is a Prolog representation of the CG category assigned to that word and `Frequency` is the number of times this category has been assigned to this word up to the current point in the learning process, or in the case of the initial closed-class words a probability distribution is predefined..

The Rules The CG functional application rules and compound noun phrase rules (see Section 2) are supplied to the learner. Extra rules may be added in future for fuller grammatical coverage.

The Categories The learner has a complete set of the categories that can be assigned to a word in the lexicon.

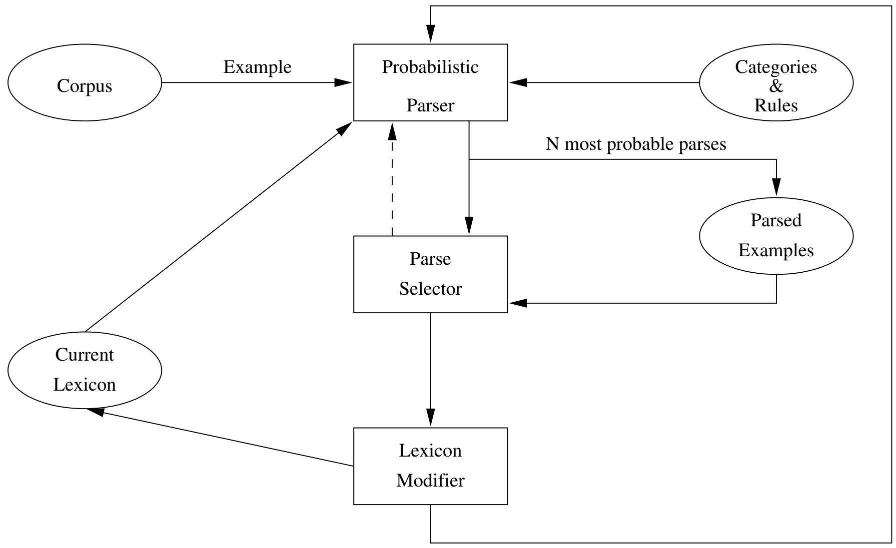

Figure 2: A Diagram of the Structure of the Learner

The Parser The system employs a n-best probabilistic chart parser, developed from a standard stochastic CKY algorithm taken from Collins (Collins, 1999). The probability of a word being assigned a category is based on the relative frequency, which is calculated from the current lexicon. Simple smoothing is used to allow for unseen lexical entries. The probabilities of the entries in the initial lexicon are predefined.

Each non-lexical edge in the chart has a probability calculated by multiplying the probabilities of the two edges that are combined to form it. Edges between two vertices are not added if there are n edges labelled with the same category and a higher probability, between the same two vertices (if one has a lower probability it is replaced). Also, for efficiency, edges are not added between vertices if there is an edge already in place with a much higher probability. The chart in Figure 3 shows examples of edges that would not be added. The top half of the chart shows one parse and the bottom half another. If n was set to 1 then the dashed edge spanning all the vertices would not be added, as it has a lower probability than the other s edge covering the same vertices. Similarly, the dashed edge between the first and third vertices would not be added, as the probability of the n is so much lower than the probability of the np.

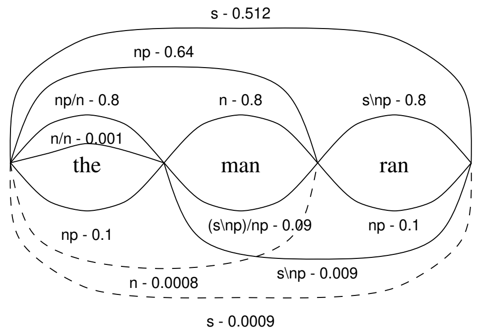

Figure 3: Example chart showing edge pruning

4.2 The Learning Procedure

Having described the various components with which the learner is provided, we now describe how they are used in the learning procedure.

Parsing the Examples Examples are taken from the corpus one at a time and parsed. Each example is stored with the group of parses generated for it, so they can be efficiently accessed in future. The parse that is selected (see below) as the current best parse is maintained at the head of this group. The head parse contributes information to the lexicon and annotates the corpus. The parses are also used extensively for the efficiency of the parse selection module, as will be described below. When the parser fails to find an analysis of an example, either because it is ungrammatical, or because of the incompleteness of the coverage of the grammar, the system skips to the next example.

The Parse Selector Each of the n-best parses is considered in turn to determine which can be used to make the most compressive lexicon (by a given measure), following the compression as learning approach of, for example, Li and Vitányi (Li and Vitányi, 1993) and Wolff (Wolff, 1987), who used it with respect to language learning. The current size measure for the lexicon is the sum of the sizes of the categories for each lexical entry. The size of a category is the number of atomic categories within it. It is not enough to look at what a parse would add to the lexicon. The effect on previous parses of the changes in lexicon frequencies must also be propagated by reparsing examples that may be affected.

This may appear an expensive way of determining which parse to select, but it enables the system to calculate the most compressive lexicon and up-to-date annotation for the corpus. We can also use previous parses to reduce some of the parsing workload.

Lexicon Modification The final stage takes the current lexicon and replaces it with the lexicon built with the selected parse.

The whole process is repeated until all the examples have been parsed. The final lexicon is left after the final example has been processed. The most probable annotation of the corpus is the set of top-most parses after the final parse selection.

5 Related Work

Wolff (Wolff, 1987) using a similar (if rather more empiricist) setting also uses syntactic analysis and compression to build grammars. However, this syntactic analysis would appear to be very expensive and the system has not been applied to large scale problems. The compression metric is applied with respect to the compression of the corpus, rather than the compression of syntactic information extracted from the corpus, as in CLL. It seems unlikely that this simple induction algorithm would generate linguistically plausible grammars when presented with complex naturally occurring data.

Joshi and Srinivas (Joshi and Srinivas, 1994) have developed a method called supertagging that similarly attaches complex syntactic tags (supertags) to words. The most effective learning model appears to have been a combination of symbolic and stochastic techniques, like the approach presented here. However, a full lexicon is supplied to the learner, so that the problem is reduced to one of disambiguating between the possible supertags. The learning appears to be supervised and occurs over parts-of-speech rather than over the actual words. However, some notion of label accuracy is supplied and this can be compared with the accuracy of our system.

Osborne and Briscoe (Osborne and Briscoe, 1997) present a fairly supervised system for learning unusual stochastic CGs (the atomic categories a far more varied than standard CG) again using part-of-speech strings rather than words. While the problem solved is much simpler, this system provides a suitable comparison for learning appropriate lexicons for parsing.

Neither Joshi and Srinivas (Joshi and Srinivas, 1994) nor Osborne and Briscoe (Osborne and Briscoe, 1997) can be considered psychologically plausible, but they are computationally effective and they do provide results for comparison.

Two other approaches to learning CGs are presented by Adriaans (Adriaans, 1992) and Solomon (Solomon, 1991). Adriaans, describes a purely symbolic method that uses the context of words to define their category. An oracle is required for the learner to test its hypotheses, thus providing negative evidence. This would seem to be awkward from a engineering view point i.e. how one could provide an oracle to achieve this, and implausible from a psychological point of view, as humans do not seem to receive such evidence (Pinker, 1990). Unfortunately, no results on natural language corpora seem to be available.

Solomon's approach (Solomon, 1991) uses unannotated corpora, to build lexicons for simple CG. He uses a simple corpora of sentences from children's books, with a slightly *ad hoc* and non-incremental, heuristic approach to developing categories for words. The results show that a wide range of categories can be learned, but the current algorithm, as the author admits, is probably too naive to scale up to working on full corpora. No results on the coverage of the CGs learned are provided.

6 Results

Early results on small simple corpora with a simpler version of the learner were presented in (Watkinson and Manandhar, 1999; Watkinson and Manandhar, 2000). Here, we present experiments performed using two complex corpora, C1 and C2, extracted from the Penn Treebank (Marcus et al., 1993; Marcus et al., 1994). These corpora did not contain sentences with null elements (i.e. movement). C1 contains 5000 sentences of 15 words or less. C2 contains 1000 sentences of 15 words or less. Lexicons were induced from C1 and then used with the parser to parse C2. Experiments were performed with a closed-class word initial lexicon of 348 entries (LIL) and a smaller closed-class word initial lexicon of 31 entries (SIL) to determine the bootstrapping effect of this initial lexicon.

The resulting lexicons are described in Table 1. These can be compared with a gold standard CG annotated corpus which has been built (Watkinson and Manandhar, 2001), which has a size of 15,136 lexical entries and an average ambiguity of 1.25 categories per word. This corpus is only loosely a gold standard, as it has been automatically constructed. However, it gives an indication of the effectiveness of the lexical labelling and is currently the best CG tagged resource available to us. The accuracy of the parsed examples both from the training and test corpora are also described in Table 1. Two measures are used to evaluate the parses: lexical accuracy, which is the percentage of correctly tagged words compared to the extracted gold standard corpus (Watkinson and Manandhar, 2001) and average crossing bracket rate (CBR) (Goodman, 1996).

In general the system performs better with the larger initial lexicon to bootstrap it. The size and ambiguity of the lexicon are close to that of the gold standard, indicating that the right level of compression has occurred. The best crossing bracket rate of 4.70 compares favourably with Osborne and Briscoe (Osborne and Briscoe, 1997) who give crossing bracket rates of around 3 for a variety of systems. Considering that they are solving a much simpler problem, our average crossing bracket rates seem reasonable.

The lexical accuracy value is fairly low. Joshi and Srinivas (Joshi and Srinivas, 1994) achieve a best of 77.26% accuracy. Two factors explain this. Firstly their system is simply disambiguating which tag to use in a context again using a corpus of tag sequences – a much simpler problem. Secondly, it would appear that the gold standard corpus they use is much more accurate than ours. Despite this, a system that assigned the tags randomly for our problem, would achieve an accuracy of 3.33%, so over 50% is a reasonable achievement.

7 Conclusions

There is further work to be completed in extending the system to allow it to deal with movement and thus the whole of the Penn Treebank. Further investigation of parameters of CLL should also be completed. Further work needs to be done in building an accurate gold standard corpus. There is also a possibility of performing experiments on sequences of parts-of-speech, as Joshi and Srinivas (Joshi and Srinivas, 1994) and Osborne and Briscoe (Osborne and Briscoe, 1997) did. This would reduce the effects of the sparse data problem.

However, we have presented a system that is psychologically plausible and whose results show that, given the complexity of the problem attempted, it is computationally effective. The results compare reasonably with systems attempting much simpler and psychologically less plausible problems.

References

Pieter Willem Adriaans. 1992. *Language Learning from a Categorial Perspective*. Ph.D. thesis, Universiteit van Amsterdam.

Martin Atkinson. 1996. Syntax and learnability. In Martin Atkinson, Stefano Bertolo, Robin Clark, Jonathan Kaye, and Ian Roberts, editors, *Learnability and Language Acquisition: a self contained Tutorial for Linguists*, pages 33 – 53. LAGB.

Derek Bickerton. 1984. The language bioprogram hypothesis. *The Behavioral and Brain Sciences*, 7:173 – 221.

Roger Brown and Camille Hanlon. 1979. Derivational complexity and order of acquistion in child speech. In John R. Hayes, editor, *Cognition and*

Initial Lexicon	Size	Average Ambiguity	Lexical Accuracy		Average CBR	
			C1	C2	C1	C2
SIL	12,706	1.21	44.76	47.53	5.43	4.70
LIL	13,851	1.24	49.54	51.89	5.61	4.86

Table 1: Summary of the Lexicons and Parses built by CLL

Development of Language, pages 11–53. John Wiley and Sons Inc.

David W. Carroll. 1994. *Psychology of Language*. Brooks/Cole Publishing Company, second edition edition.

Noam Chomsky. 1965. *Aspects of the Theory of Syntax*. The MIT Press.

Noam Chomsky. 1972. *Language and Mind*. Harcourt Brace Jovanovich.

Noam Chomsky. 1980. *Rules and Representations*. Basil Blackwell.

Noam Chomsky. 1986. *Knowledge of Language: Its Nature, Origin and Use*. Praeger.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

Joshua Goodman. 1996. Parsing algorithms and metrics. In *Proceedings of the 34th Annual Meeting of the ACL*, pages 35 – 64. Association for Computational Linguistics.

Trevor A. Harley. 1995. *The Psychology of Language: From Data to Theory*. Erlbaum (UK) Taylor & Francis.

Aravind K. Joshi and B. Srinivas. 1994. Disambiguation of super parts of speech (or supertags): Almost parsing. In *Proceedings of the 15th Conference on Computational Linguistics (COLING'94)*, pages 154–160.

M. Li and P.M.B. Vitányi. 1993. Theories of learning. In *Proceedings of the International Conference of Young Computer Scientists*.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn Treebank: Annotating predicate argument structure. In *The ARPA Human Language Technology Workshop*.

Miles Osborne and Ted Briscoe. 1997. Learning stochastic categorial grammars. In *Computational Natural Language Learning Workshop CoNLL'97*, pages 80–87.

Steven Pinker. 1990. Language acquisition. In Daniel N. Oshershon and Howard Lasnik, editors, *An Invitation to Cognitive Science: Language*, volume 1, pages 199–241. The MIT Press.

Maris Monitz Rodgon. 1976. *Single-word usage, cognitive development, and the beginnings of combinatorial speech: A study of ten English- speaking children*. Cambridge University Press.

Catherine E. Snow and Charles A. Ferguson, editors. 1977. *Talking to children: Language input and acquistion*. Cambridge University Press.

W. Daniel Solomon. 1991. Learning a grammar. Technical Report UMCS-AI-91-2-1, Department of Computer Science, Artificial Intelligence Group, University of Manchester.

Mark Steedman. 1993. Categorial grammar. *Lingua*, 90:221 – 258.

Stephen Watkinson and Suresh Manandhar. 1999. Unsupervised lexical learning with categorial grammars. In Andrew Kehler and Andreas Stolcke, editors, *Proceedings of the Workshop on Unsupervised Learning in Natural Language Processing*, pages 59–66.

Stephen Watkinson and Suresh Manandhar. 2000. Unsupervised lexical learning with categorial grammars using the LLL corpus. In James Cussens and Sašo Džeroski, editors, *Learning Language in Logic*, volume 1925 of *Lecture Notes in Artificial Intelligence*. Springer.

Stephen Watkinson and Suresh Manandhar. 2001. Translating treebank annotation for evaluation. In *Proceedings of the Workshop on Evaluation Methodologies for Language and Dialogue Systems, ACL/EACL 2001*. To Appear.

J.G. Wolff. 1987. Cognitive development as optimisation. In L. Bolc, editor, *Computational Models of Learning*. Springer Verlag.

Mary McGee Wood. 1993. *Categorial Grammars*. Linguistic Theory Guides. Routledge. General Editor Richard Hudson.