A Memory-Based Alternative for Connectionist Shift-Reduce Parsing
Report ILK-0012, available from `http://ilk.kub.nl/`

Jorn Veenstra & Walter Daelemans
[veenstra,walter]@kub.nl
Induction of Linguistic Knowledge (ILK)
Computational Linguistics, Tilburg University
PO box 90153, 5000 LE Tilburg, the Netherlands

Abstract

In this paper we present memory-based learning (MBL) as a psychologically plausible model of natural language processing. MBL shows a state-of-the-art performance on real-world language tasks, such as word pronunciation, word sense disambiguation and parsing. We compare MBL to a neural network model, SARDSRN (Mayberry & Miikkulainen, 1999) on a parsing task, and we show that MBL has a better generalisation accuracy on the test set. MBL has a solid foundation in the psychological literature within theories on human categorisation and reasoning. We argue that MBL is an alternative psychologically plausible model for human parsing, next to neural network models.

Introduction

In this paper we present memory-based learning (MBL) as a psychologically plausible model of human parsing. MBL is a class of machine learning algorithms that classify instances in a similarity based way. An MBL system contains two components: In the learning component, instances (also known as cases or exemplars, that is: a bundle of feature-value pairs associated with their correct class) are stored in an instance base (hence the name memory- or instance-based learning). In the performance component, a new instance is classified by retrieving the most similar instances from this instance base and by extrapolating the class of this new instance from the classes of the retrieved instances.

Contrary to most other machine learning techniques that are applied to human parsing tasks (e.g. maximum entropy (Rathnaparkhi 1997), transformation-based error-driven learning (Brill 1993)), MBL has a solid psychological foundation. It is inspired by exemplar- or similarity-based categorisation (e.g. Hahn & Chater (1997), Tversky (1977), Nosofsky (1988)), Case-based reasoning (Stanfill & Waltz, 1986), and analogical modeling (Skousen (1989)). Arguments in favour of memory-based parsing can also be found in the literature on child language acquisition, see for example the recent paper by Tomasello (2000).

Another class of machine learning techniques that are claimed to offer psychologically plausible models of human parsing are neural networks or connectionist models, which are inspired by the workings of neurons in the brain. Elman's (1993) simple recurrent networks (SRNs) are widely used for connectionist modeling of human parsing. SRNs are backpropagation networks with an extra context layer, which are particularly suited to predicting the next element in a sequence, e.g. predicting the next word in a sentence. In the SPEC system, Miikkulainen (1996) combined an SRN with an auto-associative RAAM network (Pollack, 1990) as a compressed distributed representation of the stack of a parser. This way he could use SRNs to construct a shift-reduce (SR) parser (Tomita, 1986). However, since the memory of a standard SRN is rather short, an often heard argument against SRNs is that they cannot handle long distance dependencies adequately. Mayberry & Miikkulainen (1999) (henceforth MM-99) present a connectionist parser (SARDSRN) that extends the SPEC system to enhance its memory capacities; SARDSRN combines the SPEC system with a self organising map (SOM (Kohonen 1990). SARDSRN gains a signicant improvement in performance on a corpus with long distance verb/argument agreement and embedded structures.

SR parsing breaks the parsing process down into relatively simple parse actions, viz. `shift`, `reduce` and `fin` (see section *Shift reduce parsing* for more details). SR parsing thus allows to rephrase the parsing process in terms of classification. For the memory-based shift reduce (MBSR) parser we build a large instance base of parse states with their correct parse actions. When a new sentence is to be parsed the MBSR parser construes a first parse state from this new sentence, classifies this state as one of the three parse actions and construes a new parse state based on this classification, the last parse state and the remaining sentence. The parser follows this procedure until the sentence is completely parsed. The MBSR parser discussed in this paper is deterministic, works in a strict left-to-right fashion, and constructs a binary tree.

We will first give a short introduction to SR parsing and MBL. Then, we discuss the experiments in which we compare the MBSR parser to SARDSRN on a simple parsing task as described in MM-99. We show that the memory-based model has a better performance (in terms of generalisation accuracy) on this parsing task than the neural model. In the final discussion we discuss four arguments in favour of memory based modeling: memory based models i) have proven to be scalable for several human language processing tasks, ii) offer comparable explanatory advantages as neural models, iii) are fast and accurate, and iv) are robust to noise and missing data.

Shift-reduce parsing

For our experiments we have used a shift reduce (SR) parser (Hermjakob, 1997; Tomita, 1986), which breaks the parsing process down into an ordered sequence of small and manageable parse actions. The central data structure of an SR parser contains a parse stack and an input buffer, which together represent the parse state. In the SR parser discussed in this paper these parse states contain words and constituents built up from these words. At the start of the parsing process the stack is empty and the input buffer contains the plain words from the sentence to be parsed. Each `shift` parse action moves a word from the input buffer onto the stack. A `reduce` parse action reduces two or more parse entries on the stack to one constituent. The sentence is fully analysed and the parser stops at the `fin` parse action. In Table 1 we give an example of an SR parse of the sentence: "The boy who liked the girl chased the cat".

Memory-Based Learning

In memory-based reasoning and learning (Stanfill & Waltz 1986; Daelemans, van den Bosch and Zavrel 1999) it is asserted that reasoning is the result of direct use of stored experiences rather than of the application of abstractions (e.g. rules) extracted from those experiences.

There is evidence that in syntactic processing analysis is influenced by similar past experiences (Juliano & Tanenhaus 1993; Mitchell, Cuetos & Corley 1992). In memory-based language processing, language is processed by analogy to previously stored experiences. If a new utterance is to be analysed, the most similar instances are retrieved from memory, and used for extrapolation. In this performance-oriented paradigm, learning is a straightforward process of storing instances and adapting similarity metrics to the particular linguistic domain. Processing directly exploits the linguistic knowledge implicit in the stored instances.

Characteristics of many linguistic phenomena are i) the large number of instances needed to cover all of a language's (ir)regularity, ii) symbolic feature values, and iii) diverse features of varying importance. In previous research, we have developed a number of methods to adapt MBL, which originates in the fields of statistical pattern recognition and machine learning, to handle these properties by:

- efficient memory-indexing for fast access,
- using distributional properties to construct similarity metrics for linguistic symbols,
- weighting features by their relevance for a particular task. This feature weighting metric (information gain) is based on the entropy of the instance base.

Memory-based shift-reduce parsing

To construct the MBSR parser we have been inspired by Simmons & Yu (1992), who propose a context sensitive SR parser. In their work a parse action is assigned to each parse state based on information on the parse stack and the input buffer. Their parser works on the level of POS tags and windows over the text with a context of five words to the left and to the right. Other work on MBSR parsing can be found in Hermjakob (1997).

A simple example of the workings of the MBSR parser is given in Table 2. In the learning stage of MBL the parse actions of the analysed sentence: "The boy chases the girl" are stored in the instance base, shown in the leftmost table. The stack contains five words/constituents; an empty position in the stack is indicated with: "–". The right context or the input buffer contains five words. The instances are formed by the stack, the right context and the correct parse action. In the performance stage the sentence "The girl chases the boy" is parsed. The trace of this parse can be found in the rightmost table in Table 2. As we want to analyse this sentence we start with an empty stack and the five first words of the sentence in the input buffer. The nearest neighbour (NN), instance number '1' from the instance base, is retrieved from the instance base; the class of this NN is `shift`, so we shift one word from the input buffer onto the stack. Based on this new instance we again retrieve the most similar instance from the instance base (instance number '2') with class `shift`. We shift one word from the input buffer onto the stack. We follow this procedure until the whole sentence is parsed.

Experiments and Results

We carried out two series of experiments. In the first we tested the MBSR parser on the MM-99 data and we looked at the effect of having more complex constructions in the test set than in the train set, and at the effect of using the feature relevance metric information gain. In a second series of experiments we compared the MBSR parser to SARDSRN. We made a rough estimation of the generalisation accuracy of the SARDSRN parser and we found that the MBSR parser has a better generalisation accuracy for all investigated tasks.

Methods

For the experiments reported in this paper we have used TiMBL[1], an MBL software package developed in our group (Daelemans, Zavrel, van der Sloot & van den Bosch, 2000) as classifier for the MBSR parser. For each new sentence we start with a stack with five empty positions and the first five words of the sentence in the input buffer. The obvious first parse action is `shift`; the next parse state has the leftmost word in the sentence on the stack and the five following words as right context. A next `shift` action puts the second word on the stack and shifts the right context one word to the right. This way we compute the parse action for each parse state dynamically; we construct the next parse state based on the predicted parse action and the current parse state. The parse action `fin` is assigned if the whole sentence is parsed, i.e. the right context is empty and only the rightmost field of the stack is filled.

Since the corpus is relatively simple we have used the word form and the complete constituents as the only feature values (e.g. without reducing the constituent to

[1] TiMBL is available from: http://ilk.kub.nl/.

Table 1: An example of a shift reduce parse of the sentente: "The boy who liked the girl chased the cat". In this case the parser takes a context of five words/constituents to the left (the stack) and five words/constituents to the right (the input buffer).

Stack	Right Context	Action
– – – – –	the boy who liked the	SHIFT
– – – – the	boy who liked the girl	SHIFT
– – – the boy	who liked the girl chased	REDUCE
– – – – [$_{NP}$the boy]	who liked the girl chased	SHIFT
– – – [$_{NP}$the boy] who	liked the girl chased the	SHIFT
– – [$_{NP}$the boy] who liked	the girl chased the cat	SHIFT
– [$_{NP}$the boy] who liked the	girl chased the cat –	SHIFT
[$_{NP}$the boy] who liked the girl	chased the cat – –	REDUCE
– [$_{NP}$the boy] who liked [$_{NP}$the girl]	chased the cat – –	REDUCE
– – [$_{NP}$the boy] who [$_{VP}$liked [$_{NP}$the girl]]	chased the cat – –	REDUCE
– – – [$_{NP}$the boy] [$_{RC}$who [$_{VP}$liked [$_{NP}$the girl]]]	chased the cat – –	REDUCE
– – – – [$_{NP}$[$_{NP}$the boy] [$_{RC}$who [$_{VP}$liked [$_{NP}$the girl]]]]	chased the cat – –	SHIFT
– – – [$_{NP}$[$_{NP}$the boy] [$_{RC}$who [$_{VP}$liked [$_{NP}$the girl]]]] chased	the cat – – –	SHIFT
– – [$_{NP}$[$_{NP}$the boy] [$_{RC}$who [$_{VP}$liked [$_{NP}$the girl]]]] chased the	cat – – – –	SHIFT
– [$_{NP}$[$_{NP}$the boy] [$_{RC}$who [$_{VP}$liked [$_{NP}$the girl]]]] chased the cat	– – – – –	REDUCE
– – [$_{NP}$[$_{NP}$the boy] [$_{RC}$who [$_{VP}$liked [$_{NP}$the girl]]]] chased [$_{NP}$the cat]	– – – – –	REDUCE
– – – [$_{NP}$[$_{NP}$the boy] [$_{RC}$who [$_{VP}$liked [$_{NP}$the girl]]]] [$_{VP}$chased [$_{NP}$the cat]]	– – – – –	REDUCE
– – – – [$_{S}$[$_{NP}$[$_{NP}$the boy] [$_{RC}$who [$_{VP}$liked [$_{NP}$the girl]]]] [$_{VP}$chased [$_{NP}$the cat]]]	– – – – –	FIN

Table 2: An example of a shift reduce parse of the sentente: "the girl chased the boy". The instance base contains the parse actions for the sentence "the boy chased the girl". In this example a context of five words/constituents to the right (the stack) and five words to the left (the input buffer) is used. See text

The instance base				Parsing a new sentence			
ins	stack	input buffer	cl	stack	input buffer	NN	pr
1	– – – – –	the boy chases the girl	S	– – – – –	the girl chased the boy	1	S
2	– – – – the	boy chases the girl –	S	– – – – the	girl chased the boy –	2	S
3	– – – the boy	chases the girl – –	R	– – – the girl	girl chased the boy –	3	R
4	– – – – [the_boy]	chases the girl – –	S	– – – – [the_girl]	chased the boy – –	4	S
5	– – –[the_boy] chases	the girl – – –	S	– – – [the_girl] chased	the boy – – –	5	S
6	– –[the_boy] chases the	girl – – – –	S	– – [the_girl] chased the	boy – – – –	6	S
7	–[the_boy] chases the girl	– – – – –	R	– [the_girl] chased the boy	– – – – –	7	R
8	– – [the_boy] chases [the_girl]	– – – – –	R	– – [the_girl] chased [the_boy]	– – – – –	8	R
9	– – – [the_boy] [chases_[the_girl]]	– – – – –	R	– – – [the_girl] [chased_[the_boy]]	– – – – –	9	R
10	– – – –[[the_boy]_[chases_[the_girl]]]	– – – – –	F	– – – – [[the_girl]_[chased_[the_boy]]]	– – – – –	10	F

their heads or adding extra lexical information). The context was fixed at five words/constituent to the left and five words to the right. We performed experiments with and without the feature weighting metric information gain.

Data

For our experiments we used the same corpus as MM-99. This corpus features relative clauses and agreement between verbs and arguments. The grammar that generated this corpus is shown in Table 3.

For a first series of experiments we randomly took 20% of the corpus as test set. The remaining 80% of the sentences was taken as basis for the train data. The sentences early in the corpus are simpler than the sentences later in the corpus; certain syntactical construction only occur in the end of the corpus. To simulate the effect of incremental learning we first trained the parser on only the first half of the train set. This way the parser is trained on just the simple sentences and tested on a combination of simple and complex sentences. In a second and a third experiment we used 75% and 100% of the sentences from the train data to train the parser. As a last experiment we trained the parser on the first sentence only.

Results

The results of the first series of experiments are shown in Table 4. The measure we use for evaluating the experiments is the $F_{\beta=1}$ value (van Rijsbergen 1979), an average between precision and recall. Precision is percentage of predicted constituents that is actually correct, recall is the percentage of correct constituents that is predicted. We computed the precision and recall with *evalb* (Sekine & Collins 1997). The results in Table 4 show that if we use the full train set for training we reach an $F_{\beta=1}$ value of 100%. This can be explained by the fact the the grammar in Table 3 produced a noiseless and

Table 3: The phrase structure grammar that generated the corpus. The corpus thus contains relative clauses and agreement between subjects, objects and verbs.

$S \rightarrow NP(n)\ \ VP(n,m)$
$VP(n,m) \rightarrow Verb(n,m)\ \ NP(m)$
$NP(n) \rightarrow Det\ \ Noun(n)$
$RC \rightarrow who\ \ VP(n,m)$
$NP(n) \rightarrow RC(n)$
$RC(n) \rightarrow whomNP(m)\ \ Verb(m,n$

—l—

$Noun(0) \rightarrow boy$	$Noun(1) \rightarrow girl$
$Noun(2) \rightarrow dog$	$Noun(3) \rightarrow cat$
$Det \rightarrow the$	
$Verb(0,0) \rightarrow liked, saw$	$Verb(0,1) \rightarrow liked, saw$
$Verb(0,2) \rightarrow liked$	$Verb(0,3) \rightarrow chased$
$Verb(1,0) \rightarrow liked, saw$	$Verb(1,1) \rightarrow liked, saw$
$Verb(1,2) \rightarrow liked$	$Verb(1,3) \rightarrow chased$
$Verb(2,0) \rightarrow bit$	$Verb(2,1) \rightarrow bit$
$Verb(2,2) \rightarrow bit$	$Verb(2,3) \rightarrow bit, chased$
$Verb(3,0) \rightarrow saw$	$Verb(3,1) \rightarrow, saw$
$Verb(3,3) \rightarrow chased$	

deterministic corpus. We first discuss the experiments with the use of information gain (IG) (see the middle column in Table 4). Note that the $F_{\beta=1}$ value is already at 74% if we train the parser on just the very first sentence (viz. "the girl chased the cat") in the corpus. This very simple parser is able to recognise simple NPs and complete sentences, but it fails hopelessly on more complex embedded structures, since it was not trained on such structures. Taking the first half of the train set as training corpus gives an $F_{\beta=1}$ score of 82%. This parser is able to generalise to structures that it has not seen before. The same holds for the parser trained on the first 75% of the train set. The only errors produced by this variant were sentences in the form: "The cat saw the girl whom the girl saw", as it had not been trained on sentences with "whom". If we look at the rightmost column in Table 4 we can see that the results for the experiments without IG are on average 3%-4% lower.

Table 4: Results of the MBSR parser. 20% of the sentences is first taken out at random as test material. The parser is trained on the rest of the sentences, on respectively the first sentence, on 218 (50%), 327 (75%) and 435 (100%) sentences. The results with information gain are given in the second column, without IG in the third column.

train set	$F_{\beta=1}$	
	IG	no IG
1 sentence	74%	71%
50%	82%	79%
75%	95%	91%
100%	100%	100%

Comparison to SardSRN

The MBSR parser is trained to predict the bracket structure of the sentence, using the original wording; constituent precision and recall seems a natural choice to evaluate the MBSR parser. SARDSRN generates a compressed, distributed representation of the various stages of analyses of the sentence and learns mappings of various stages of analyses. Hence, sentences parsed by SARDSRN can contain words that were not in the original sentence and vice versa. This can the reason that MM-99 decided not to use constituent precision and recall as a measurement of the performance of their parser but the "average mismatches per sentence". This measure "reports the average number of leaf representations per sentence that are not correctly identified from the lexicon by nearest match in Euclidean distance". These different ways of evaluation and the fact that MM-99 work with a validation set and a test set (whereas the MBSR parser only uses a test set), make it somewhat difficult to compare SARDSRN with the MBSR.

Nevertheless, in order to be able to compare the performance of SARDSRN and the MBSR parser we made a rough approximation of the $F_{\beta=1}$ of the results reported in MM-99. We used the following formula:

$$F_{\beta=1} = 100\% - \frac{MM*100\%}{NoC}$$

where $F_{\beta=1}$ is the average over precision and recall, MM is the average number of mismatches per sentence, and NoC is the average number of constituent per sentence in the corpus (in our case this is 8).

See Table 5 for an comparison of the results of SARDSRN to the MBSR parser. Note that the results reported for SARDSRN are just a rough approximation based on Figure 5 in MM-99. However, in spite of this approximation we can still see that the MBSR parser does not make any error on the corpus (except for one error in the 20% experiment). Note that the $F_{\beta=1}$ for SARDSRN when trained on 20% of the data is lower than the $F_{\beta=1}$ for the MBSR parser when trained on the first sentence (see Table 4).

Table 5: Comparison of the results of the MBSR parser to the SARDSRN parser. 20%, 40%, 60% and 80% is taken at random as train set, the rest is used as test set. We report the average over precision and recall ($F_{\beta=1}$) of the MBSR and an estimation of the average number of mismatches per sentence of the SARDSRN parser as given is MM-99 and the derived estimation of average over precision and recall.

	MBSR	SARDSRN	
train set ratio	$F_{\beta=1}$	mism.	est. $F_{\beta=1}$
20%	99.9%	≈ 3	$\approx 63\%$
40%	100%	≈ 2	$\approx 75\%$
60%	100%	≈ 1	$\approx 88\%$
80%	100%	≈ 0.5	$\approx 93\%$

Discussion

In this paper we have presented a memory-based learning model of parsing. MBL models have been applied to several areas in language technology and human language processing tasks, such as word sense disambiguation (Veenstra, van den Bosch, Buchholz, Daelemans & Zavrel, 2000), shallow parsing (Buchholz, Daelemans & Veenstra, 1999), PP attachment (Zavrel, Daelemans & Veenstra 1997), POS tagging (Daelemans, Zavrel, Berck & Gillis, 1996), morphology (van den Bosch & Daelemans 1999) and pronunciation (van den Bosch 1997). All these MBL applications are real-life, performed on large amounts of data and show state-of-the-art performance.

MBL stores instances in an instance base. To classify a new instance, the most similar instances are retrieved from this instance base. The predicted class is extrapolated from the classes of these retrieved instances.

We believe that MBL has a serious credibility as a model for human parsing. The psychological underpinning of MBL is founded in the fields of similarity-based categorisation, analogical reasoning and case-based reasoning. MBSR parsing is a viable model of human sentence processing since it: i) works in a left-to-right fashion, (which humans also do); ii) is context sensitive (words or constituents in different contexts can play different roles); iii) is robust to noise and missing data (the MBSR parser already has a reasonable performance when trained on only 50% of the data); iv) can be used for psycholinguistic modeling (e.g. the influence of leaving out certain information (lesioning) or incremental learning); v) easily integrates information from different sources, such as word class, semantic and pragmatic information; vi) is scalable to real world language data; and finally vii) uses a similarity-based classification strategy, which gives it a solid foundation in the psychological literature.

Some issues that arise from the experimental results in the paper are:

Feature relevance Information gain is an entropy-based feature relevance metric that can be used in MBL. In the first series of experiments we have shown that adding information gain causes an important improvement in generalisation accuracy for the parser. Feature weighting can be used to boost crucial information and to weed out irrelevant features. If we want to add information from several sources feature weighting can be used to suppress noisy sources.

Generalisation The MBSR parser reached an $F_{\beta=1}$ value of 100% when trained on a large enough partition of the data. This errorless performance is not caused by mere table-lookup. For the MBSR parser, trained on the complete train set, only 10% to 20% of the parse action classifications is based on an exact match; the other 80% to 90% of the parse action classifications are extrapolated from non-exact matches. The performance component of the MBSR parser generates new knowledge by generalising over instances in the instance base.

Comparison to SARDSRN The neural network parser SARDSRN offers a psychologically motivated model of human parsing. The results in MM-99 show that such parsers can also be applied to embedded structures and long distance dependencies with reasonable success. However, if we compare SARDSRN to the MBSR parser we see that the MBSR parser outperforms SARDSRN, both in terms of accuracy (the MBSR parser showed a better generalisation accuracy in all experiments) and in terms of efficiency (it took around four days to train the SARDSRN parser, compared to a training time in the order of seconds for the MBSR parser). Moreover, MBL has proven to be scalable, since a lot of MBL applications work on large corpora, with hundreds of thousands of instances, as in POS tagging (Daelemans et al., 1996) and shallow parsing (Buchholz et al., 1999).

Future research As the corpus we used in this paper is relatively simple we have not really shown that the MBSR parser also works on real-world language data. We are currently working on extending the MBSR parser to a large English corpus: the Wall Street Journal corpus (Marcus, Marcinkiewicz and Santorini, 1993), a corpus with over a million words of financial newspaper text, annotated with syntactic and some rudimentary semantic structure. Previous work on MB shallow parsing (Buchholz et al., 1999) shows promising results. We also plan to work on lesioning studies with MBL: what effect does the omission of certain information sources sort on the performance and do these effects have explanations in psycholinguistics. We also plan to extend our research to the field of child language aquisition. Some previous work in the area of memory-based models of child language acquitition has been done by Kempen, Blom, Gillis, & Wijnen (1999).

Acknowledgements

This research was carried out in the framework of Induction of Linguistic Knowledge (ILK), sponsored by NWO (Netherlands Organisation for Scientific Research).

References

van den Bosch, A. (1997). *Learning to pronounce written words.* Dissertation, University of Maastricht.

van den Bosch, A., & Daelemans, W. (1999). Memory-based morphological analysis. In: *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics, ACL'99.* Maryland, USA.

Brill, E. (1993). *A Corpus-Based Approach to Language Learning.* Dissertation, University of Pennsylvania.

Daelemans, W., van den Bosch, A., & and Zavrel, J. (1999). Forgetting exceptions is harmful in language learning. In: *Machine Learning, special issue on natural language learning,* **34**.

Daelemans, W., Zavrel, J., Berck, P., & Gillis, S. (1996). MBT: A Memory-Based Part of Speech Tagger-Generator. In: *E. Proceedings of the Fourth Workshop on Very Large Corpora, WVLC-96.* Copenhagen, Denmark.

Daelemans, W., Buchholz, S., & Veenstra, J. (1999a). Memory-Based Shallow Parsing. In: *Proceedings of the Conference on Computational Natural Language Learning (CoNLL-99)*, Bergen, Norway.

Daelemans, W., Zavrel, J., van der Sloot, K., & van den Bosch, A. (1999b). *TiMBL: Tilburg Memory Based Learner, version 2.0, Reference Guide.* ILK Technical Report 99-01, 1999 `http://ilk.kub.nl/papers.html` .

Elman, J. (1993). Learning and development in neural networks: the importance of starting small. *Cognition,* **14,2**.

Hahn, U., & Chater, N. (1997). Concepts and Similarity. In: *Knowledge, Concepts and Categories*, ed. Koen Lamberts & David Shanks. Psychology Press.

Hermjakob, U. (1997). *Learning Parse Decisions From Examples With Rich Context.* Ph.D. Thesis, Department of Computer Sciences, University of Texas at Austin.

Juliano, C. & Tanenhaus, M.K. 1993. Contingent frequency effects in syntactic ambiguity resolution. in: *Proceedings of the Fifteenth annual Conference of the Cognitive Science Society.* Hillsdale NJ.

Kempen, M., Blom, E., Gillis, S., & Wijnen, F. (1999). Do Dutch children learn root infinitives from their mothers? In: *Pragmatics in 1998: Selected papers from the 6th International Pragmatics Conference*, **2**. Antwerp Belgium.

Marcus, M., Marcinkiewicz, M., & Santorini, B. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. In: *Computational Linguistics*, **19,2**.

Mayberry, M., & Miikkulainen, R. (1999). SARDSRN, A Neural Network Shift-Reduce Parser. *Proceedings of the 16th Annual Joint Conference on Artificial Intelligence (IJCAI-99)*. Stockholm, Sweden.

Miikkulainen, R. (1996). Subsymbolic case-role analysis of sentences with embedded clauses. *Cognitive Science*, **20**.

Mitchell, D., Cuetos, F. & Corley, M. 1992. "Statistical versus Linguistic determinants of parsing bias: cross-linguistic evidence". in: *Fifth annual CUNY conference on human sentence processing*. NY.

Nosofsky, R. (1988). Exemplar-based accounts of relations between classification, recognition, and typicality. In: *Journal of Experimental Psychology, Learning, Memory, and Cognition*, **14**.

Pollack, J. (1990). Recursive distributed representations. In: *Artificial Intelligence*, **46**.

Ratnaparkhi, A. (1997). A linear observed time statistical parser based on maximum entropy models. In: *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing, EMNLP-2*. Providence, Rhode Island.

van Rijsbergen, J. (1979). *Information Retrieval*. Buttersworth, London, UK.

Sekine, S., & Collins, M. (1997). *evalb - bracket scoring program* retrievable from: `http://cs.nyu.edu/cs/projects/proteus/evalb/`.

Simmons, R., & Yeong-Ho Yu, (1992). The Acquisition and Use fo Context-Dependent Grammars for English. *Computational Linguistics*, **18,4**.

Skousen, R. (1989). *Analogical modeling of language*. Kluwer, Dordrecht NL.

Stanfill, C. & Waltz, D.L. 1986. Toward Memory-based Reasoning. *Communications of the ACM*, **29**.

Tomasello, M. (2000). Do young children have adult syntatic competence?. In: *Cognition*, **74**.

Tomita, M. (1986). *Efficient Parsing for Natural Language*. Kluwer, Dordrecht NL.

Tversky, A. (1977). Features of similarity. In: *Psychological Review*, **84**.

Veenstra, J., van den Bosch, A., Buchholz, S., Daelemans, W., & Zavrel, J. (2000). Memory-Based Word Sense Disambiguation. In: *Computers and the Humanities, special issue on Senseval, Word Sense Disambiguations*, Ed. Adam Kilgarriff and Martha Palmer, **34,1-2**.

Zavrel, J., Daelemans, W., & Veenstra, J. (1997). Resolving PP Attachment Ambiguities with Memory-Based Learning In: *Proceedings of the Conference on Computational Natural Language Learning (CoNLL-97)*. Madrid, Spain.