Hierarchical Self-Organization of Corpora
The Conflict between Theory and Practice

David M. W. Powers
AI Group, Dept of Computer Science
The Flinders University of South Australia
David.Powers@flinders.edu.au

Abstract

Machine Learning and Natural Language are major but distinct subfields of Artificial Intelligence. This paper looks at the question of what Machine Learning techniques are appropriate in Natural Language applications, and presents empirical results that apparently fly in the face of the theoretical results on language learning.

We review the theoretical results about what cannot be learned by particular systems and paradigms, and discuss the way they have led to Machine Learning and Neural Nets concentrating on supervised learning paradigms. We then consider the Self-Organizing models and Clustering and Classification techniques which have been known since at least the 1940s and look at results achieved using these strictly less powerful paradigms.

This paper has two aims. The first is to clarify the difference between the supervised and unsupervised approaches. The second aim is to present evidence that unsupervised learning is appropriate for Natural Language, the theoretical results notwithstanding.

Keywords

Machine Learning, Natural Language, Self-Organization, Supervision, Connectionism

1.0 Background and Introduction

1.1 Self-Organization and Parsimony

Since at least Turing's work at the end of the 1940s on "How the Leopard got its Spots", Self-Organization has been a serious contender as a model and theory to explain aspects of the structure and organization of biological systems. In a self-organizing system, a relatively small and simple set of constraints are responsible for a relatively large and complex structure, the exact form of which may be influenced by the environment external to the system.

Since Shannon's elucidation of the principles of Information Theory, we have had the mechanism for a formalization of the principle of Parsimony, which dates back at least as far as William of Occam. Today, Parsimony and Information Theory combine to give us the principle of Minimum Description Length. Information Theory allows us to measure the redundancy in a system, and thus to compress the information contained in the system so as to represent it in the minimal amount of space. Parsimony suggests that the smallest and simplest theory or model of a phenomenon is most likely to be the most useful one, and is indeed most likely to be the correct one. Thus the problem of devising a good theory can be reformulated as a problem in compression.

The connection between compression and learning was recognized in 1971 in the context of language learning [Wolff,92] and was generalized to a more general theory of scientific methodology, cognition and computation[Wolff,91]. More recently the label Minimum Description Length has been applied to the use of this approach to deciding between competing concept descriptions developed by a Machine Learning system.

The older self-organizing neural networks and models have now been largely displaced by backpropagation networks. More generally, in Machine Learning, supervised learning is the primary learning paradigm, and unsupervised learning is on the endangered list. Why is this? Is there good reason for eschewing unsupervised learning? Is there good reason for pursuing unsupervised learning?

1.2 Paradigms and Experimental Setup

We now proceed to review the concept of supervised and unsupervised paradigms. We will illustrate the kinds of learning and analysis which can be carried out unsupervised. We will subsequently investigate in some detail the extension of the unsupervised paradigm to hierarchical learning.

Often, when people speak of supervised and unsupervised learning, they have in mind particular algorithms. Thus in the case of Neural Nets one thinks of Backpropagation or Kohonen, respectively. Unfortunately, there is a tendency for the mention of Neural Nets to evoke solely Backpropagation and its variants, along with a particular kind of supervised training paradigm. Similarly when one thinks Machine Learning, particular algorithms tend to come to mind, such as ID3 [Quin81] and its successors, again implying a particular kind of supervised training paradigm. While this is particularly true outside of the relevant research communities, there seems nonetheless to be a trend even within the communities to focus on these paradigms — and we will see that there are indeed theoretical grounds for doing so, although there seems to be a tendency to an unnecessarily broad interpretation of these results.

The archetype of unsupervised learning is perhaps the self-organizing neural net of von der Malsburg [vdM73], in which recognizers for lines at different angles emerge from a combination of neural plasticity and a sombrero-shaped lateral interaction function. There is no external intervention in this or similar experiments. There is no attempt to train the network to recognize specified patterns. Thus, in no variant of the experiment is there any feedback to indicate “correct” or “incorrect” outcomes. Indeed, in some variants of the experiment, no external stimuli are required at all!

The most well-known supervised learning techniques today are probably ID3 [Quin81] and Backpropagation [PDP86]. Both of these require the provision of an indication of the “correct” answer for each of the cases which constitute their training sets. Whilst this may seem to characterize and constrain supervised learning quite closely, again there is quite a degree of variability in the way the experiment may be setup. Furthermore, it actually possible to employ these techniques within an unsupervised learning situation.

It is therefore useful to focus on the learning *paradigm*, the experimental method, as distinct from the learning *algorithm*. Indeed the theoretical results about learning, and in particular the negative theoretical results about language learning [Gold67], are predicated on the choice of learning *paradigm* rather than the choice of a particular *algorithm*. This is clear, since they claim to show that learning is impossible in a given situation, given an arbitrary or even an optimal choice of algorithm. Unfortunately, an understanding of the significance of the role of paradigm, indeed familiarity with the very concept of paradigm, is all too rare among practitioners in the field and others who quote the theoretical results on learning.

The traditional distinction between the supervised and unsupervised paradigms focuses on negative information. When we talk about a paradigm, we are referring to an experimental framework or a teaching situation. In this paper we mean both! Traditionally, the experimental framework for machine learning consists of a corpus of examples or cases. In an unsupervised paradigm, we typically have a corpus of positive examples and no example of incorrect sentences (in the linguistic context). In a supervised paradigm, we classically have a set of positive and negative examples, with appropriate annotation as to which class they belong too.

Both stereotypes admit considerable variation. For example, our unsupervised and supervised training sets may both admit a certain amount of noise (which we will treat as synonymous with classification error). The supervised paradigm may provide labels or tags other than the binary positive-negative classification. However, if the tags are mutually exclusive, such datasets can be reformulated classically. If there is overlap between classes, the paradigm is weakened (for example English lexemes may belong to multiple parts of speech, which represents a problem which the learning algorithm must address). Conversely, if the labels are structured, there is more information available than in the classic formulation (as when we use a treebank rather than a simple tagged corpus) and we have a stronger paradigm [cf. Angl88].

This distinction seems rather straightforward, but there are two points worth making. First, there are a number of features of the common learning paradigms which do not fit with our intuitions of natural human learning, and thus appear artificial. The choice of, and isolation of, the examples used is performed by the trainer or supervisor, who thus potentially has two roles: *teacher* (provider of examples) and *critic* (provider of negative information in any of the forms just discussed). Indeed even classical unsupervised paradigms are ‘supervised’ to the extent that they have a teacher providing them with selected examples. But most of a child’s learning, particularly of language and his understanding of the world (ontology), occurs before the child is involved in a formal teacher-student relationship. Although the parent may be viewed as playing a similar role, there is abundant evidence that she does not provide overt supervision in either the sense of *teacher* or *critic*.

The second point arises out of this: negative information may vary along a dimension from implicit to explicit. It is not necessarily dichotomously present or absent. By presenting the information in an order which exercises all rules applicable in sentences of a given length before going on to longer sentences, we have a form of negative information implicit which states that no sentence of this length is correct that is not derivable using the illustrated rules.

1.3 Formal Results on Learnability

There are many significant results on learnability which have been developed over the last three decades. For the purposes of this paper, two are relevant: The first is the death of Neural Networks in the guise of the Perceptron (triggered by Mins69]) and the rebirth a decade later as Parallel Distributed Processing in the form of Multilayer Backpropagation networks. The various results on learnability of formal languages, and in particular those of [Gold67], form the second.

Of the first result, it suffices to note that the problem with the Perceptron had nothing to do with learning paradigms. The fact that Perceptrons cannot encode XOR was a direct result of the single layer model. AND and OR and NOT gates can all be coded using Perceptrons. Thus Multilayer Perceptrons can handle XOR. More interesting is the fact that Perceptrons can encode (and learn) the detection of convex objects but cannot encode (or learn) the detection of connected objects. In this case, the fact that Perceptrons sought to be true to real-life neurons and limited connectivity to a certain subset of other neurons is implicated: this is known as the locality constraint and in Backpropagation this has also been dropped in favour of universal connectivity of neurons in adjacent layers.

The second set of theoretical results deals with language and is thus germane to the results which we wish to present. It is assumed that we declare the language learned when we get to the point where no more errors are made (in identifying sentences of the language). In fact, we can never be sure that we won’t find another sentence of the language which isn’t correctly identified by our grammar. But it seems reasonable to suppose that children have finished learning the language at some point, as judged by a sufficiently large period of competent performance. The result says that without negative information, given sentences from a language that contains an infinite number of arbitrarily long sentences (viz. there is no direct limit on sentence length) it is impossible to distinguish the correct grammar from a grammar which describes exactly the set of sentences seen and no others (and thus defines a finite language) or which describes any finite or infinite language which includes that set of sentence (and there are infinitely many such languages).

Interestingly, it turns out that the so-called “anomalous text” [Gold67] of the kind described at the end of the previous subsection provides implicit negative information sufficient for learnability. Conversely, just saying that a particular sentence is ungrammatical may not provide sufficient information to determine why it is wrong, let alone what the grammatical version would be (assuming the sentence is being generated from some semantic or ontological representation using the grammar being learnt). This is the issue of “focus”, which is addressed by the more highly structured supervised paradigms. Finally, it is clear that non-linguistic cues and knowledge play a role in focusing our language learning apparatus, and it is possible to construct supervised training sets from arbitrary corpora using a simple filtering process guided by ontological knowledge.

We illustrate this using as our example the past tense learning paradigm used by [R&M86]. In this experiment, pairs of present and past tense forms of a set of verbs were associated and learnt by a neural net. Is this supervised or unsupervised? It doesn’t correspond well to the standard positive/negative concept of criticism, but in that the unique “correct” answer is supplied for each of the present tense forms (and conversely, for past tense forms) it has the flavour of one of our variant supervised paradigms. This is even more the case if we consider that we are trying to learn which words are regular (positive examples which may be derived by an -ed rule) and which are irregular (negative examples which must be treated as exceptions). Alternatively we may acknowledge a finite number of productive rules (-ed plus vowel changes) and constrain the number of exceptions to be finite, and in this case we are back to the finite number of possible classifications for each word. What is clear is that in order to generalize beyond the words actually seen, some kind of rules must be learned and be represented somehow in the network (without making any assumptions about the symbolic or distributed nature of the representation). Conversely, unseen ‘irregular’ forms which the network correctly handles can’t really be regular — there are more rules evident in English verb morphology than just the -ed one.

On the basis of our definition of the paradigms, we would tend to see the paradigm used as supervised. But note that the feedback was a full past tense form, not a label: -ed, o->e, etc. There was still work to do extracting the relevant features. Furthermore, consider the child’s situation: At the point when he is learning past tense forms, and then rebalancing the -ed rules with secondary rules and irregular forms (after an apparent overgeneralization phase), he has been hearing the forms for some time and has already attached meaning to them. The combination of an association of meaning, an association of the morphological stem and a distinction of past versus present context are certainly sufficient to turn the corpus of sentences he is hearing into the form required for supervised learning. And there is no evidence of children mastering past-tense morphology before they have any concept of past!

Note that the algorithm used has played no role in the discussion above. Moreover, “unsupervised” techniques find patterns in the data, and addition of criticism clearly adds to the patterns it can find. Of course, it assumes that critics are absolutely sure that their analyses are correct. Indeed, the addition of critical information will in general change the outcome of applying the algorithm - as it is now “supervised” and it will tend to learn the patterns the supervisor has provided for it to learn (if it doesn’t it will be discarded as useless).

On the other hand, it is also possible to use “supervised” techniques in an autocorrelation mode for unsupervised learning. In this case, the aim of the experiment is usually recovery of the whole instance from incomplete patterns, but generalization is still performed and recognizable classifiers and rules are still formed.

Of course, generally it makes sense to use algorithms which are optimized for the paradigm in which one will be employing them. Indeed, there are further formal results (which we will not discuss) that insist that that choice of representation is also critical to successfully learning the *intended* concept — and that often there are other concepts which could equally well have been learned, and which will appear more parsimonious in some representation. The problem of representation is quite topical right now, and we will also be straining at the bounds imposed by these results in what follows, but we do not treat it as a major issue.

1.4 Common Forms of Unsupervised Learning

Clustering, Classification, Data Mining, Linear Prediction, Markov Models, Self-Organization and Emergence are all terms which can construed as referring to unsupervised learning in particular contexts, and there are many others. It is perhaps worth clarifying that the distinction between Self-Organized and Emergent behaviour has its roots in intentionality. And the question of whether an algorithm is used in an unsupervised or a supervised context is related to the cost of setting up the data in the form required by the algorithm — and whether human involvement was involved at an example by example level.

Self-Organization implies that a system has been designed for the function under study, and that it succeeds in this using a configuration which is apparently underspecified, but in fact specifies boundary conditions for the type of behaviour which should be acquired. Emergence implies that the behaviour under study is being achieved as a by-product of a system whose primary function is quite different. There are, moreover, contexts in which the use of neither term necessarily refers to learning as such, or where the acquired function cannot appropriately be classified as behaviour (consider the leopard's spots [Turi52]).

We will tend to use the terminology of classification in the rest of the paper, and will categorize the algorithms we discuss as clustering algorithms, even though they are frequently better known by other names. In fact, the precise algorithms used play relatively little role in the investigation being described and by and large the algorithms we have investigated will neither be described nor discussed. Similarly, we will be working within a broad class of possible representations, and one goal of the research is that the precise choice of representation be immaterial.

2.0 Extraction of Hierarchical Structure

Suppose that we have a corpus of untagged data, language unspecified and irrelevant. In other words we have a set of sentences, a book, email, recorded speech, or some such. The point is that we have done no preanalysis to identify parts of speech, phrases, or (in the case of speech) phones, syllables or words.

Traditionally, we would treat this data as a sequence of words structured as a sequence of sentences and treat speech as a quite separate problem. Traditional syntax focuses on the syntactic problem of determining the substructure which mediates between sentence and word. Linguists use this information to check that the sentence is indeed a legal grammatical sentence. Computational Linguists use this structure as a basis for extracting meaning or translating between languages, etc.

In English and other Indo-European languages (as well as those whose orthography has been developed under such influences), words are delimited by spaces, and sentences by full-stops (or a small collection of variants thereof).

Automated language analysis, or machine learning of natural language, traditionally adopts the linguists' position and seeks to bridge from word to sentence. The task is easy to set up, as both words and sentences are so clearly delimited. But the task itself is probably impossible in a pure syntactic unsupervised learning paradigm!

I, however, believe that unsupervised language learning is possible, and that the rules and structures which compose sentences out of words are learnable within a totally unsupervised paradigm. The remainder of this paper describes the learning paradigm which I and my colleagues are pursuing, argues its sufficiency to the task and presents preliminary results from implemented versions.

2.1 Specification of the task

The preceding two paragraphs may appear to contradict each other. However, careful readers will have noted that they were set up: I claimed that the traditional word-to-sentence task was impossible, but that I could learn this information in an unsupervised paradigm; The assertion I am making here is that restriction of the task to word-to-sentence is ill-advised.

The first point worth making in support of this claim is the following: Words are ill-defined; Sentences are ill-defined; The definition of both varies considerably across languages, as well as across styles and sublanguages. An example of each: Consider English “write out” versus German “ausschreiben”; Consider what is delimited by semicolon and colon in this paragraph and the preceding one. (We ignore here languages like Chinese which don’t even meet the basic assumptions made above.)

We claim that words are not *the* basic units of language (unBLOODYlikely) and sentences are not *the* point of transition from recursion to iteration.

A second point relates to the example above of learning the past tense, where we claimed that “the combination of an association of meaning, an association of the morphological stem and a distinction of past versus present context are certainly sufficient to turn the corpus of sentences he is hearing into the form required for supervised learning”. This is clearly not possible “in a pure syntactic unsupervised learning paradigm” restricted to the word-to-sentence problem!

A general rule of Computer Science is that if the original problem is too hard, we should seek to solve it as a special case of a more general problem. In this case, we generalize to the full language hierarchy, and possibly to the full ontological hierarchy (encompassing our sensory-motor interaction with and understanding of the world).

In this paper we focus on paradigms that employ the full information available from an arbitrary untagged corpus. Our research program aims to explore the limits of what can be achieved within this paradigm. Our hypothesis is that there is a level between simple phrase and full clause level where this approach will run out of steam, and an appeal to the ontological unsupervised learning paradigm will be necessary (for which we use a simulated robot world).

Some of the experiments described here were in fact performed in the word-to-sentence domain, but the majority were performed in a domain which extended below the word level. Thus we typically start out with individual characters, or sometimes phonemes, phones or even speech code-vectors. We will use the term unit in our subsequent discussion to make it independent of choice of initial or current level.

2.2 Contextual classification

A classification technique seeks to classify points (or vectors) in a multidimensional attribute space into classes of similar items. In the unsupervised corpus domain, we seek to classify units which are similar on the basis of attributes drawn from the surrounding units, or context. This kind of approach has often been applied at the word level, interestingly more often with the aim of discovering semantic classes than syntactic, and then often with the aim of probabilistically improving the performance of a speech recognition or machine translation system. We have applied it at both word level and at various lower levels. (The first applications at character level date back to cryptography in World War II [Sukhotin62/63(Russian)] as cited in [Boy77(German)]. My first application at character level was a follow-up of initial success at word level [Powe83;85;89], recognizing that subword morphs such as endings and vowel changes required subword analysis and that the same techniques should be capable of this.)

If we use the immediate context, we do actually discover syntactic classes. If we use the broader context, we tend to discover classes of semantically related terms. There have been several studies which have explored the effects of choice of context [Finc93; Scho91; Schi94; Schu93].

With all classification techniques, there is a problem as to where to draw the line between one class and the next. In some cases, this is a property of the parameters of the model. In others, we simply join units or clusters together in a binary fashion as nearest neighbour clusters according to some metric, and thus we end up with a binary tree or “dendrogram” showing a classification hierarchy for our entire set of units (lexicon or alphabet).

While it is relatively easy to distinguish sensible classes as subtrees of the dendrogram, we have no sound and consistent basis for doing so, as yet. Interestingly Singular Valued Decomposition (SVD) and related matrix analysis techniques provide what may well be a useful clue [Schi94]: many of the eigenvectors capture exactly one linguistically significant dimension, and the SVD pruning can actually lead to a more intuitive dendrogram.

In our terminology, units may be (part of) either the context or the putative concept which is a candidate for membership in the current class. We therefore use the terms concept and context in a rather nonstandard way to refer to the units in focus and the units in a specified window of context, respectively. The concepts categorized together we denote as a class, and the set of contexts (which are represented by the vectors actually classified) associated with a concept or a class is termed its coset.

2.3 Hierarchical classification

Ironically, what we really want of an automatic analysis of a corpus is in fact a hierarchical analysis: a parse tree. But the hierarchical analysis which we obtained above is something quite different: it is more like an ontology or a taxonomy.

However, it is relatively straightforward to modify our classification paradigm to generate context-free rules, and hence parses. Moreover, this approach may also help to solve the class bounding problem discussed in the previous subsection. Finally, we are investigating variants of the scheme which seem to be capable of generating more general grammars, similar to unification grammars.

We discuss each of these developments of the classification procedure in turn.

Suppose that we have generate an initial class by initial classification. Typically we have used the most frequent unit as a seed, and obtain from that what invariably turns out to be the most frequent class. In our initial experiments, we constrained the class to be in a certain range (e.g. seven +/- two) and maximized one of various arbitrary metrics designed to approximate coverage of the corpus [Powe83,89,91]. In fact, certain of the metrics used always identified the same class.

At the character level, this initial class was the vowels (precise membership varied across languages, and according to treatment of accents, umlauts and palatalization). At the word level the class was the articles plus possessives (in a pilot experiment on English and French using corpora on children’s readers).

Already at this level, we are faced with the problem that the orthography may use two or more letters to represent a single phoneme (e.g. German “sch”). This was the initial motivation for introducing the technique which allowed learning of grammatical rules. (In fact, there is currently a debate as to whether German “tcsh”, as in “Deutsch”, represents a single phoneme.)

Not only do we constrain the class to a certain size, we constrain both the context and the concept to specified ranges. For example, in German we allowed the concept to be one to three characters. Note that we also investigated representation of the accented and umlauted vowels of French and German, respectively, both as special individual characters as well as as sequences of diacritical plus vowel. We also investigated phonemic representation with precisely one character per phoneme and speech vectors with a many-to-many mapping.

Ideally, our classification should be independent of representation, and although the ideal of complete independence was never achieved, and indeed the classes produced are sensitive to the precise algorithms and metrics used, useful automatic analyses have been obtained in all cases [Powe92; Schi94; Winn95].

Once we have a class, we can replace the member concepts by the class, corresponding to replacing terminal symbols by a non-terminal, and then repeating the classification process. If we were to use the standard single-symbol classification, we would obtain exactly the dendritic structure of the previous subsection.

However, if we allow multiple units to constitute a single concept and then classify these concepts together, what we obtain is a hyperclass of context-free rules: i.e. the new non-terminal we assign to the discovered class can be replaced by any of the sequences of units which are members of the class, which can be represented as a series of context-free rules with the new non-terminal on the LHS and one of the concept sequences on the RHS.

This process can then be repeated until we obtain a single superclass.

The resulting rule sets are very grammar-like, although the rules are not always what tradition would lead us to expect. Moreover, having experimented with many different classification techniques and several alternate metrics, the results have almost always been linguistically plausible (the single exception I have seen was essentially degenerate). However, the grammar tends to reduce to a single symbol in around eight steps.

Note that if we use a simple binary approach using nearest neighbour classification, the algorithm itself determines when to extend a class, and when to introduce a new context-free rule. This suggests a whole new range of possibilities as to when to close off a class. For example, we can compile away singleton concepts from classes with no binary or ternary rules (assuming the one to three symbol range in our concepts, as suggested above). Similarly, we could limit the concept range to two symbols and compile away any singleton classes (thus German “sc” would retain no identity separate from “sch”).

At the moment, we are experimenting with retaining some of the information which is thrown away at each stage in the process described above. Clearly we are throwing away too much information, as in the above scenario in which we retain no information other than that implicit in the occurrence of the non-terminal with which the original sequence of symbols (terminal or non-terminal) is replaced.

It will be noted that the initial class found at word level was a closed class, and the vowels at character level are also in a sense a closed class. The sequence of classification continues to revolve around closed classes, giving rise to a whole new interpretation of phrase structure grammar [Powe91; cf. M&M91].

The lower the level a class occurs, the more closed it seems to be. We are experimenting with retaining the more closed part of the non-trivial concept sequences as features, and exploring how the resulting model relates to existing feature-based formalisms.

3.0 Conclusion

This paper has sought to give an overview of the view of paradigm and the approach to hierarchical self-organization which underlie our research program. Specific results and algorithms have not been discussed, and we refer the interested reader to the following work, and in particular the SHOE Proceedings [Dael92].

References

[Angl88] Dana Angluin, “Queries and Concept Learning”, Machine Learning 2#4:319-342.

[Boy77] Joachim Boy, “Dechiffrierungsalgorithmen zur phonetischen identifikation von Buchstaben”, Bochum Dissertation, Studienverlag Brockmeyer.

[Dael92] Walter Daelemans and David M. W. Powers, Eds, Background and Experiments in Machine Learning of Natural Language: First SHOE Workshop (282pp), ITK Proceedings 92/1, Tilburg University NL.

[Finc93] Steven Finch, “Finding Structure in Language”, PhD Thesis, University of Edinburgh UK [Gold67] E. Mark Gold, “Language Identification in the Limit”, Information and Control 10, pp 447-474.

[M&M91] David M. Magerman and Mitchell P. Marcus, “Distituent Parsing and Grammar Induction; The Automatic Acquisition of Linguistic Structure from Large Corpora.” pp. 122-125 in Powers & Reeker, Proceedings AAAI Spring Symposium on Machine Learning of Natural Language and Ontology, Document D-91-09, DFKI, Univ. Kaiserslautern FRG.

[vdM91] Christoph von der Malsburg, “Self-Organization of Orientation Selective Cells in the Striate Cortex”, Kybernetik 14:85-100.

[PDP86] Parallel Distributed Processing, Vol1: Foundations, David E. Rummelhart, James L. McClelland and the PDP Research Group; Vol2: Psychological and Biological Models, James L. McClelland, David E. Rummelhart and the PDP Research Group, MIT Press 1986.

[Powe89] David M. W. Powers and Christopher Turk, Machine Learning of Natural Language, Research Monograph, Springer-Verlag, 1989.

[Powe91] David M. W. Powers, “How far can self-organization go? Results in unsupervised language learning.” pp. 131-137 in Powers & Reeker, Procs AAAI Spring Symposium on Machine Learning of Natural Language and Ontology, Document D-91-09, DFKI, Univ. Kaiserslautern FRG.

[Powe92] David M. W. Powers, “On the Significance of Closed Classes and Boundary Conditions: Experiments in Lexical and Syntactic Learning”, pp. 245-266 in [Dael92a].

[Quin81] J. Ross Quinlan, “Induction of Decision Trees”, Machine Learning 1#1:81-106

[R&M86] James L. Rumelhart and David E. McClelland, pp216-272 in [PDP86:Vol.2]

[Schi94] Georg Schifferdecker, “Finding Structure in Language”, Diplom Thesis, University of Karlsruhe FRG (May 1994).

[Scho91] Jan C. Scholtes, “Learning simple semantics by self- organization”, pp146-151 in Powers & Reeker, Proceedings AAAI Spring Symposium on Machine Learning of Natural Language and Ontology, Document D-91-09, DFKI, Univ. Kaiserslautern FRG.

[Schu93] Hinrich Schuetze, “Part of Speech induction from scratch” in Proc ACL 31, pp251-258, 1993.

[Turi52] Alan Turing, “The chemical basis of morphogenesis”, *Phil. Trans. Royal Society*, London Series **B 194**, 431-445.

[Winn95] Winn, Tiffany, ‘‘A Comparative Analysis of Automated Extraction of Phoneme Classes Using Unsupervised Learning’’ submitted to IJCAI-95 Workshop on New Approaches to Learning for Natural Language Processing.

[Wolf82] Jerry Wolff, “Language acquisition, data compression and generalization.” *Language and Communication,* **2**, 57-89.

[Wolf91] Jerry Wolff, *Towards a Theory of Cognition and Computing.* Ellis Horwood.