Cognitive Dynamics - Dynamic Cognition?

Reginald Ferber[1]

Fachbereich 2

Universität – GH Paderborn

D-33095 Paderborn, Germany[2]

Abstract: In the last ten years a paradigm shift took place in cognitive science. While during the seventies problems were commonly attacked by symbol processing approaches, in the last decade many researchers employed connectionist models. These models can be seen as dynamical systems on metric spaces. There is not yet a developed theory of the behavior of these systems, but they seem to be a challenge for future research. The purpose of this paper is to introduce the problem and the dynamic approach to it.

1 Cognitive Processes

The subject of cognitive science is the description and simulation of cognitive processes and structures, especially in the areas of memory, problem solving, verbal behavior, and image identification.

Some cognitive processes, for example the production and the parsing of sentences, seem to employ sophisticated symbol manipulating operations, other processes, such as image identification or access to word meaning, seem to rely on fast processing of huge amounts of rather vague knowledge that has been learned in many different situations. This learning is performed in a smooth way, enabling generalization, context sensitivity and noisy inputs. This use of experience makes it necessary to compare situations, i. e. to decide if a new situation is equal to or resembles an old one. This comparison might be achieved by use of distance measures which are sensitive to many parameters of the situation. Distances between symbolic objects are rather artificial constructions, while they are elementary to elements of metric spaces.

Another controversy in cognitive science, which is closely related to the question of symbolic processing, is the question, to what degree the cognitive system is modular ([8]). A modular view assumes that the cognitive apparatus consists of independent modules between which data are exchanged. This assumption seems to be the natural consequence of a symbol processing approach. On the other hand, it seems difficult to explain the high speed of perception processes with modular symbol processing systems based on rather slow biological neurons. There are also empirical data that seem to contradict a strictly modular approach.

[1] This research was supported by the Heinz-Nixdorf-Institut and by the Deutsche Forschungsgemeinschaft (DFG) Kennwort: Worterkennungssimulation

[2] E-mail: ferber@psycho2.uni-paderborn.de

To explain these aspects, models with distributed memory and parallel processes have been proposed that can be interpreted as dynamical systems on metric spaces. These models are known under many names, such as connectionism, parallel distributed processing (pdp), neural networks ([10], [12], [1], [7], [9], [13]), and are defined in many different ways. In the following paragraph some formal definitions will be given that catch the central aspects of these models to unify terminology (See also [3]).

2 Neural Networks

The following very general definition includes most of the deterministic models used in literature. Beside these, there are non-deterministic or probabilistic models.

2.1. Cellular Structure

Let W be a set and I a countable set. $c : I \to W$ is called a *configuration* of *values* of W on the *cells* of I. $C = W^I = \{c : I \to W\}$ denotes the space of all configurations.

For every cell $i \in I$ let $N(i) \subset I$ be a finite, ordered subset, the *neighborhood* of cell i. The set $N = \{N(i) \mid i \in I\}$ of all neighborhoods defines a directed graph with node set I and the set of edges $\{(i, j) \mid j \in N(i)\}$, the *connection graph, net structure* or *grid* of the cellular structure.

For every cell $i \in I$ let $f_i : W^{N(i)} \to W$ be a *local function*. Let further $f = \{f_i \mid i \in I\}$ be the set of all local functions. Then $Z = (I, W, N, f)$ is called a *cellular structure*. $F : C \to C$ with $F(c)(i) := f_i(c(N(i)))$ is called the *global function* of Z.

If I is finite, $Z = (I, W, N, f)$ is called a *finite cellular structure*.

If W is finite, $Z = (I, W, N, f)$ is called a *cellular automaton*.

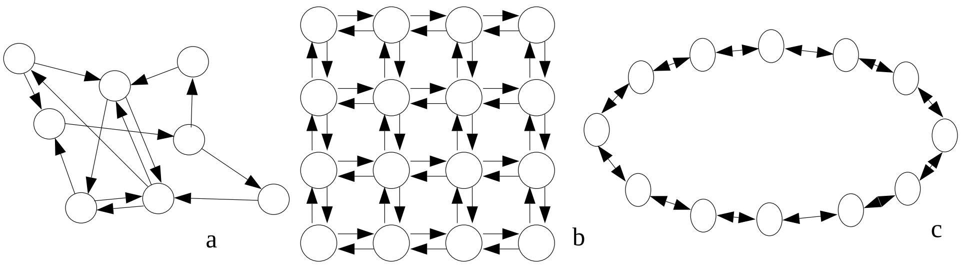

Figure 1: Three Different grid structures. Neighbors are indicated by an arrow from the neighbor to the cell itself. a) Arbitrary grid, b) Rectangular grid with VON NEUMANN *neighborhood, c) One dimensional circular grid.*

The global function defines an autonomous dynamical system on the configuration space. The behavior of the system can be influenced by the structure of the grid and by the nature of the local functions. Both kinds of restrictions are used to construct models of cognitive behavior. The following restriction on the local functions is used frequently:

2.2. Neural Net

1. A cellular structure $Z = (I, W, N, f)$ with $W \subset \mathbb{R}$ and

$$f_i(c(N(i))) = \xi_i \left(\sum_{j \in N(i)} w_{i,j} c(j) \right) \text{ with } w_{i,j} \in \mathbb{R} \tag{1}$$

and monotonic not decreasing functions $\xi_i : \mathbb{R} \to W$ is called a *(deterministic) neural net*. ξ_i is called the *output function* of cell i. $w_{i,j}$ is called the *weight* from cell j to cell i.

2. A function of the form

$$\sigma_\theta : \mathbb{R}^n \to \{0,1\}; \ \sigma_\theta(x) = \zeta_\theta \left(\sum_{i=1}^{n} w_i x_i \right) = \begin{cases} 1 & if \ \sum\limits_{i=1}^{n} w_i x_i - \theta \geq 0 \\ 0 & otherwise \end{cases}$$

is called a *linear threshold function* with weights $w_1, ..., w_n$ and threshold θ.

The dynamic behavior of a neural net on a given grid is determined by the output functions and the weights. In many cases the same output function is chosen for all cells. Then the behavior of the system depends only on the weights. They can be chosen to achieve a desired behavior of the system. This can be done either in one single step (see for examples [6], [4]) or in a longer adaptation process of small smooth changes either before the use of the net as a model or even during the use of the system. This construction of appropriate weights is often called *learning*.

The following restriction on the structure of the net forces a simple dynamical behavior:

2.3. Feed Forward Net

Let $Z = (I, W, N, f)$ be a cellular structure. The set $S_0 = \{i \in I \mid N(i) = \emptyset\}$ is called the set of *input cells*.

Let $S_n \subset I$ be the set of cells that can be reached from the cells of S_0 by passing through exactly n edges of the connection graph of Z. If $S_0 \neq \emptyset$ and all S_n are disjoint, the grid is called a *feed forward grid* and $S_n \neq \emptyset$ are called *layers* of the grid. S_0 is called *input layer*, S_n with $n = max\{m \in \mathbb{N} \mid S_m \neq \emptyset\}$ is called *output layer* and all layers in between are called *hidden layers*. $Z = (I, W, N, f)$ is called a *feed forward net* or a *feed forward network*.

Feed forward neural nets are used to transform an input pattern of values on the input layer into a pattern of values on the output layer of the grid. A well known example with three layers is the *perceptron* developed 1962 by F. ROSENBLATT [14] and extensively studied by M. MINSKY and S. PAPERT in [11]. Other examples with more layers and continuous output functions are *back-propagation networks*. The name is due to the way in which the weights are computed: First the weights are set to random values;

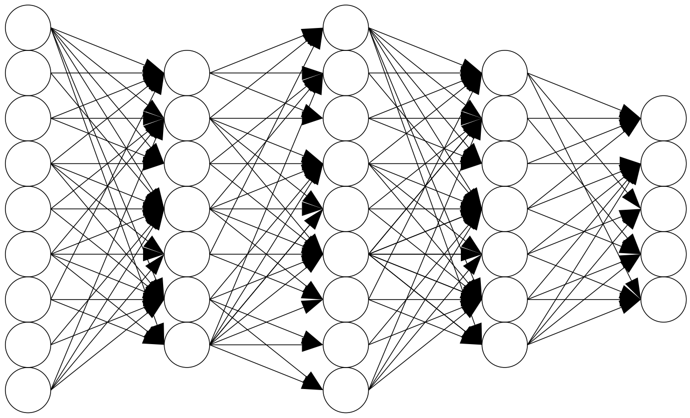

Figure 2: A feed forward grid with 5 layers

then, using a sample of given input and target patterns as training material, the pattern on the output cells produced by the network from the input pattern is compared with the target output. Using a gradient descend method, the weights are changed in such a way that for this input pattern the difference between the output of the net and the target is reduced. To adapt the weights between earlier layers, error values are propagated back to these layers, and a correction for the weights is computed using these error values. (For details compare [9].)

The dynamic of a feed forward net is quite simple: Starting with an arbitrary configuration $c \in C$, the values of the input cells in the first iterate $F(c)$ depend only on their local functions since they have no input (argument). From the second iteration on the values of the cell in S_1 are constant, since they have only the constant input from the cells of S_0. In this way the values of the subsequent layers become constant in subsequent iterates. In a net with n layers the iteration sequence reaches the same fixed point from every configuration within $n + 1$ iterations.

3 Example

In the following we shall concentrate on experiments investigating the process of word recognition. One goal of these experiments is to answer the question of modularity of the cognitive system, in this case the question, if there is an independent module functioning as a *mental lexicon* where the words known by a person are stored.

First we shall give a brief description of the experimental situation in which data on human word recognition are collected. Then we shall outline a simulation of such data using a back-propagation network. Finally a dynamic model is proposed.

3.1 Word Recognition and Priming

Word recognition is an experimental paradigm that is used frequently to investigate cognitive processes in verbal behavior. The basic idea is to measure the time people need to respond to the presentation of a written word, the so called *target*. The requested reactions are either to name the target (*naming experiment*), or to decide, if a presented string of characters is a word of the native language of the person or not, by pressing an appropriate button (*lexical decision experiment*). In both cases the time elapsing between the onset of the presentation of the target and the onset of the reaction is measured. There are many studies investigating the effect of

- frequency of the target in language
- regularity of pronunciation
- length of the target

and the like.

Priming experiments investigate the effect of context on naming and lexical decision. In this case the presentation of the target is preceded by the brief presentation of another word, the so called *prime*. This prime can be related to the target in different ways. It can

- be the same word typed differently (upper vs. lower case) (*identity priming*).
- be semantically related (*semantic* or *associative priming*)
- precede the target frequently in natural language (*syntactic priming*)
- be similar as a string of characters (*graphemic priming*)

If the presentation of a target that is related to the preceding prime leads to a quicker reaction, then the mental lexicon is probably not completely modular.

The results show complex behavior (see [5], [16] for an overview and references). While some studies found some of the priming effects, others did not. There seem to be many factors influencing the results. At least it seems to be rather unlikely that a mental lexicon exists that is completely modular.

3.2 A Back-propagation Model

We shall now present a model of word recognition that catches some of the features of a parallel and distributed system.

3.2.1. The Model

In 1989 M. SEIDENBERG and J. McCLELLAND proposed a "Distributed, Developmental Model of Word Recognition and Naming" [15]. They used a modified back-propagation model and were able to simulate "many aspects of human performance including (a) differences between words in terms of processing difficulty, (b) pronunciation of novel items, (c) differences between readers in terms of word recognition skill, (d) transition from beginning to skilled reading, and (e) differences in performance on lexical decision and naming tasks."[15: page 523].

The net they used, consisted of 3 layers: an "orthographic" input layer S_0 of 400 cells, a hidden layer S_1 with 100 to 200 cells, and an output layer S_2 that was divided in two parts: a "phonological" output part with 460 cells and an orthographic part that was similar to the input layer. The phonological part of the output was used to simulate naming data, the orthographic part was used to simulate lexical decision data. The layers were fully forward connected, i. e. for $m \in \{1, 2\}$ and $i \in S_m$ it holds $N(i) = S_{m-1}$.

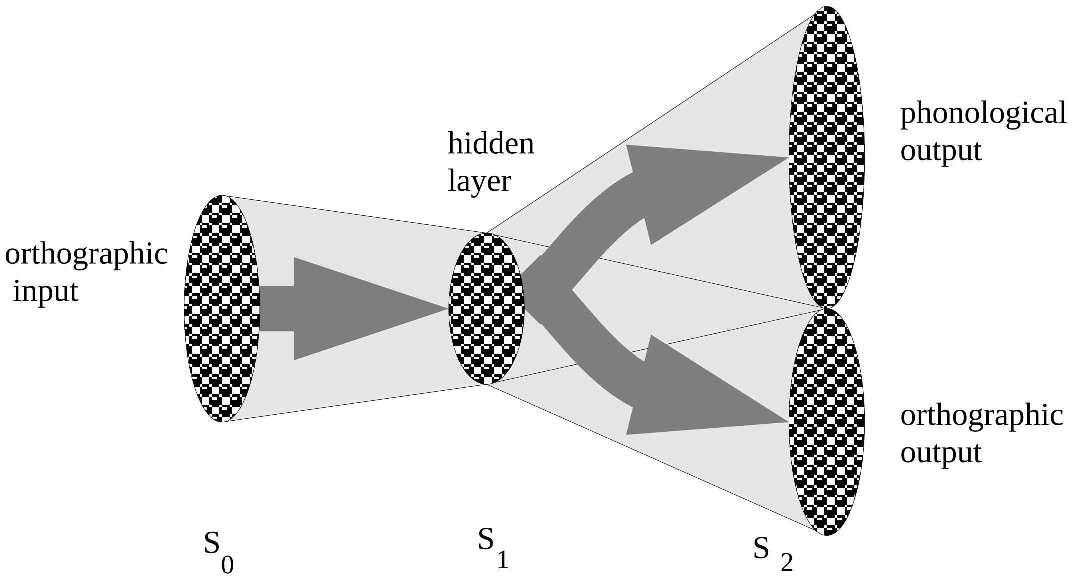

Figure 3: Structure of the back-propagation grid used by M. SEIDENBERG *and* J. McCLELLAND. *It can be seen as one grid from S_0 to S_2 or as two grids, one from the orthographic input to the orthographic output and one from the orthographic input to the phonological output.*

The model was built for monosyllabic words consisting of more than 2 letters which were not foreign words, abbreviations or complex words that were formed from the addition of a final *–s* or *–ed* inflection. The representation of these words on the input layer and, as targets, on the orthographic part of the output layer was constructed by a complicated transformation of the letter triplets occurring in the word. The representation of the phonological targets was a transformation of triplets of phonemes occurring in the pronunciation of the word. (For details compare [15] and [9 vol.2, chap.18])

To simulate the reaction time the mean quadratic error between the output pattern computed by the net on the respective part of the output layer and the target pattern was used. The assumption was that a convergence process takes place, in which an erroneous pattern converges to the correct one. This process should take more time, if the error is big.

The model was trained with 2,884 stimulus – target pairs, presented from about 14 times for low frequent words up to 230 times for the most frequent words. With every presentation the weights were changed for the orthographic part of the output and the phonological part of the output. Thus the weights from the input to the hidden layer were trained twice, for the orthographic to phonological net and the orthographic to orthographic net.

3.2.2. Remarks

Several remarks can be made on the model described above (3.2.1).

1. The model realizes a "metric" system, since input and output are elements of an n dimensional space. It can be seen as a continuous mapping from $\mathbb{R}^{400}$ to $\mathbb{R}^{460} \times \mathbb{R}^{400}$. This continuity is probably one of the reasons for the ability of the model to exploit regularities of the training material and generalize them to new material.
2. The effectiveness of the continuity in generalization depends on the representation of the input. On the one hand it has to represent enough of the crucial information of the individual input to distinguish it from other inputs, on the other hand it has to generalize over the individual inputs to extract features they have in common. The representations used in the model are very sophisticated, hence a good deal of its power may be due to the "constructed" representations.
3. As the authors mention the number of cells in the hidden layer has a strong influence on the performance of the model. It determines how much information can be passed through this layer, i. e. how detailed or generalizing the treatment of a single input can be.
4. The special structure of the net with the hidden layer in common for the orthographic to phonological net and the orthographic to orthographic net, can be a reason for the model's generalization behavior in the simulation of the lexical decision task. The representation of the information on the hidden layer has to take account of both the phonological and the orthographic properties of a word.
5. The authors stress the point that their model has no lexicon. But the orthographic to orthographic net is a device that reproduces a word from a string of letters. Due to the continuity it is somewhat robust against small perturbation. It will produce the correct output even if only partial information is given as input. Hence with an appropriate functional definition of a lexicon, it is just a distributed implementation of a mental lexicon, including phonetic influences as described in the last remark (4).

6. The authors view their model as part of a larger model including layers for meaning and context. In the present implementation it is not visible how these additional components should be integrated. Hence the simulation of further processes such as priming is not possible.
7. Because of the feed forward structure of the net, there is no possibility to explain the influence of previous inputs or stronger influence of a longer input. To the model it makes no difference if the input is presented once or for a longer time.

4 A Dynamic Model of Word Recognition and Priming

The model outlined in 3.2.1 simulates reaction times by distances between patterns of activities on parts of a neural net and expected target patterns. It is assumed that larger distances result in longer times for the formation of correct patterns as input to the next component of the cognitive system.

In the remaining part of the paper we shall outline some ideas how a simulation could work that uses the convergence of a dynamical system on a metric space to simulate word recognition processes.

4.1 Basic Assumptions

First some assumptions are listed that point out the basic principles of the dynamic model.

4.1.1. Cognition as Dynamical Process

The first assumption of the dynamical model is, that cognitive processes are simulated by a dynamical system given by the global function of a neural network. The cognitive states are represented by the configurations, the time course of the process is simulated by the iteration sequence. If the iteration sequence approaches a small attractor, for example a fixed point, this either corresponds to a stable cognitive state, for example the meaning of a word or image, or it is a constant or periodic input to other neural nets stimulating further processes. In both cases the assumption is central, that only configuration sequences that have some stability over (a short period of) time can cause something to happen.

4.1.2. Learning: Detecting Regularities in the Input

The second basic idea is, that the neural net is slowly but constantly changed by its input in such a way that co-occurring events are associated, i. e. the configurations resulting from frequent and frequently co-occurring events in the input of the system should be stable. This enables the net to “detect” regularities in its input (compare [7]). From the point of view of the dynamical system this means that by changing the weights of the neural net, the attractors of the global function and their basins of attraction have to be changed in such a way that frequent input patterns become attractors.

4.1.3. Constantly Triggering Input

In contrast to 3.2.1 it is assumed, that the grid has no pre-defined structure, especially no feed forward structure, but that the structure develops during learning. It should be not very densely connected and it should contain loops. Input is presented to the net in such a way that the input pattern is added to the activities of the input cells for several applications of the global function; i. e. the system is no longer an autonomous system, but is triggered by the external input. $c_{n+1} = F(c_n + i_n)$ with $(c_n + i_n)(j) := c_n(j) + i_n(j)$. The input cells are only a small fraction of all cells of the net. From this fraction the influence of the input spreads out to the other cells. There it can match with the existing patterns (of the previous attractor) or it can force them to change, moving the system to the basin of a different attractor. This constant triggering allows on the one hand to control the duration and strength of the input, on the other hand influences of previous inputs are preserved for a while to interact with new influences, as it is necessary to simulate priming effects (compare 3.2.2.7).

4.1.4. Subnets and Modularity

The distributed representation of the processed information as pattern on the cells of the grid allows complicated interactions, including modularization. It is possible that a subset of cells is strongly interconnected, but has only a few connections to other cells. Such a subset or subnet could be called a module. It is also possible that the system converges for a while relatively independent on two such subnets towards sub-patterns of different attractors, and that later on conflicts arise between these two sub-patterns. For example there might be subnets incorporating “meaning”, and “context”, as proposed by [15]. In such a case the configuration coming from the (orthographic) input may converge on one part of the net (say meaning) to one attractor but on the other part (context) it may converge to another attractor, because the surrounding information points toward a different interpretation. This may lead to a contradiction and finally one of the two attractors will win.

The idea of shaping attraction basins is very powerful. It opens possibilities for the explanation of many effects in word recognition. On the other hand it is not yet in such a concrete state that any one of these explanations can be more than a hypothesis.

4.2 Simulation of Word Recognition Processes

In terms of this model the processes involved in naming, lexical decision and priming can be described in the following way:

4.2.1. Naming

For the naming task the system has to stimulate the pronunciation of the written word. In a modular approach it is assumed that this is done by the production of a phonological code, which in turn is the basis for the generation of a motor code that controls articulation. A comparable system is also possible for the dynamical model, as a cascade of neural nets, one stimulating the next one as soon as it has reached a stable state (see also [2]). The dynamic model can explain several other phenomena: Frequent words are named faster, since their attractors are strong; regularly pronounced words are named faster, since the sequence of letters are more frequent and hence lead to faster convergence.

4.2.2. Lexical decision

The lexical decision task requires to distinguish between character strings representing words and character strings that do not represent words. In general the words used for this purpose are well known, short, and frequent words of the native language of the subject. The non–word strings are constructed in such a way that they have the same length and that they are pronounceable. From 4.1.2 it should follow that there is no attractor for these strings since they are new to the system, and there is no meaning associated to them. Hence in those parts of the grid whose configurations represent meaning there should be no convergence. Of course there can be convergence just by chance, but that is equivalent to a wrong answer of a person.

4.2.3. Priming

Priming effects occur, when the system is moved by the influence of the prime towards the attractor of the target: The input of the prime changes the configuration of the net in such a way that, if the following target is related to the prime, the configuration will be already closer to the attractor of the target, than it has been before the prime influenced the net. Hence the attractor is reached faster than without the prime.

4.2.3.1 Identity priming. If the target is the same word as the prime but written in lower case letters, while the prime was written in upper case letters, most of the patterns induced by the two strings will be the same. Hence the impact of the prime on the net will be very similar to that of the target.

4.2.3.2 Semantic priming. If the prime and the target are semantically related, they appear more frequently together (see [18]). Hence they can lead to the same attractor concerning "meaning" and "context": the influence of the prime moves the system closer to an attractor that is in many respects also a possible attractor for the target.

4.2.3.3 Syntactic priming is based on frequent co-occurrence of words in language. According to 4.1.2 this should lead to faster convergence.

4.2.3.4 Graphemic priming is based on the similarity of character strings, i. e. the prime is a string of characters in which only very few characters are changed compared to the target. If the strings are entered by activating input cells that represent short sequences (tuples) of characters, most of these tuples will be the same in the prime and the target. Hence a weak form of identity priming will take place.

4.2.4. Priming with ambiguous words

Of special interest are experiments with ambiguous targets, i. e. letter strings that have several meanings. In general a semantic priming effect is observed only for the *primary meaning*, i. e. the more frequent meaning. If the prime has a strong impact towards the less frequent meaning (*secondary meaning*), for example if a whole sentence is used to prime that meaning, the reaction is also faster. A closer analysis of the processes ([17]) shows that at first both meanings are activated according to their frequency. While the primary meaning quickly reaches a high availability, the availability of the secondary meaning grows slower. After about 300 ms the secondary meaning reaches nearly the same availability as the primary meaning. Afterwards its availability decreases again.

These data could be explained by a process like that described in 4.1.4. First there is an relatively independent evolution of patterns on different parts of the net, one representing the primary meaning, one representing the secondary meaning. After a while the developing patterns grow so large that they get into a conflict in which the pattern of the primary meaning suppresses that of the secondary meaning.

Figure 4: Two ambiguous figures: Left the so called Necker Cube: Either vertex a or vertex b can be seen as being in front. The figure on the right can either be seen as two black faces or as a white candlestick.

A similar process could cause the well known switching effects for ambiguous figures like those shown in figure 4: The two meanings are represented by neighboring attractors of the dynamical system. The influence of additional information moves the system from the basin of one attractor to that of the other.

References

[1] Arbib, M. A. (1987). *Brains, Machines, and Mathematics*. Springer, (Expanded edition).

[2] Dell, G. (1986). A spreading-activation theory of retrieval in sentencde porduction. *Psychological Review 93*(3), 283-321.

[3] Ferber, R. (1992). Neuronale Netze. In *Jahrbuch Überblicke Mathematik*, S. C. Chatterji, B. Fuchssteiner, U. Kulisch, R. Liedl, & W. Purkert, Eds. Vieweg, Braunschweig, Wiesbaden, pp. 137–157.

[4] Ferber, R. Vorhersage der Suchwortwahl von professionellen Rechercheuren in Literaturdatenbanken durch assoziative Wortnetze. In *Mensch und Maschine – Informationelle Schnittstellen der Kommunikation. (Proceedings ISI '92)* (1992), H. H. Zimmermann, H.-D. Luckhardt, & A. Schulz, Eds., Universitätsverlag Konstanz, pp. 208–218.

[5] Gorfein, D. S., Ed. (1989). *Resolving Semantic Ambiguity*. Springer-Verlag.

[6] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA*.

[7] Kohonen, T. (1988). *Self-Organization and Associative Memory*. Springer Verlag, Berlin, (Second edition).

[8] Levelt, W. J. M. (1991). Die konnektionistische Mode. *Sprache & Cognition 10*(Heft 2), 61–72.

[9] McClelland, J. L., Rumelhart, D. E., & the PDP Research Group (1986). *Parallel Distributed Processing Explorations in the Microstructure of Cognition*. The MIT Press, Cambridge Massachusetts.

[10] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics, vol. 5*.

[11] Minsky, M. L., & Papert, S. A. (1988). *Perceptrons*. The MIT Press, Cambridge, Ma., (Expanded edition, first edition 1969).

[12] Palm, G. (1982). *Neural Assemblies An alternative Approach to Artificial Intelligence*. Springer.

[13] Ritter, H., Martinez, T., & Schulten, K. (1990). *Neuronale Netze*. Addison-Wesley.

[14] Rosenblatt, F. (1962). *Principles of Perceptrons*. Spartan, Washington, DC.

[15] Seidenberg, M. S., & McClelland, J. L. (1989). Distributed, developmental model of word recognition and naming. *Psychological Review No. 4*, 523–568.

[16] Sereno, J. A. (1991). Graphemic, associative, and syntactic priming effects at a brief stimulus onset asynchrony in lexical decision and naming. *Journal of Experimental Psychology: Learning, Memory, and Cognition No. 3*, 459–477.

[17] Simpson, G. B., & Burgess, C. (1985). Activation and selection processes in the recognition of ambiguous words. *Journal of Experimental Psychology: Human Perception and Performance*(11), 28–39.

[18] Wettler, M., Rapp, R., & Ferber, R. (1993). Freie Assoziationen und Kontiguitäten von Wörtern in Texten. *Zeitschrift für Psychologie 201*.