Universiteit Antwerpen
Universitaire Instelling Antwerpen
Departement Taal -en Letterkunde

Dyslexia in the neural network DISCERN

Promotor: Erik Tjong Kim Sang

Eindverhandeling voor het behalen van de titel van licentiaat in de Germaanse Taal —en Letterkunde

Door: Koen Van Eyken

Antwerpen 2000

Table of contents

1. Theoretical background 3
1.1 Introduction 3
1.2 Dyslexia 4
1.2.1 Deep dyslexia 4
1.2.2 Surface dyslexia 5
1.3 Neural networks 5
1.3.1 What is an artificial neural network? 5
1.3.2 History 6
1.3.3 Representation 7
1.3.4 Problems 7
1.3.5 Example 8
1.4 Semantic networks 8
1.4.1 What is a semantic network? 8
1.4.2 History 9
1.4.3 Problems 10
1.4.4 Example 11
2 Cognitive models for language processing 12
2.1. DISCERN 12
2.1.1 What is DISCERN? 12
2.1.2 How does DISCERN work? 13
2.1.3 DISCERN in Artificial Intelligence 14
2.2 FGREPNET 16
2.2.1 What is FGREPNET? 16
2.2.2 Using FGREPNET 17
2.3 DISLEX 18
2.3.1 What is DISLEX? 18
2.3.2 The lexical representation of words 19
2.3.2.1 Orthographic representation 19
2.3.2.2 Phonetic representation 20
2.3.3 The semantic representation of words 20
2.3.4 Association of lexical and semantic representations 22
2.3.5 Modelling dyslexia 22
3. Experiments 23
3.1 The semantic representation of clusters 23
3.2 Identifying semantic concepts 24
3.3 The corpus Martial 25
3.4 Research questions and expectations 29
3.5 Results 30
3.5.1 Processing corpus Martial with FGREPNET and DISLEX 31
3.5.1.1 Modular error correction 31
3.5.1.2 Training the network 32
3.5.1.3 Error rate versus performance 32
3.5.1.4 The statistical approach 33
3.5.1.5 Model of human lexicon 34
3.5.2 Modelling dyslexia in DISLEX 34
3.5.2.1 Surface dyslexia in DISLEX 35
3.5.2.2 Deep dyslexia in DISLEX 37
a) Semantic map 37
b) Lexical to semantic 38
c) Semantic to lexical 40
d) The phonetic representations 41
4. Conclusion 42
Bibliography 46
Appendix 49

1. Theoretical background

In this chapter I will give introduction to this thesis, explaining the reasons why I choose this topic for my thesis. After that, I will provide some theoretical background by commenting on three central notions of this thesis. The neural disorder dyslexia and its two most important variants are discussed, followed by a brief introduction to artificial neural networks and semantic networks. This is necessary because the system I am studying, DISCERN, is a mixture of both neural and semantic networks.

1.1 Introduction

I was in doubt whether to write my thesis about computational linguistics or psycholinguistics. Both these fields are socially relevant and have the potential to help people in their everyday lives. I searched the list of possible thesis subjects till I found a combination of the two: neural networks. If a computer system can accurately model the human brain, it might be easier to find solutions to the many problems humans can have with this sensitive piece of equipment. A search in the library and on the Internet gave me a wide array of possible subjects in this field, but one article in particular caught my attention. The programme DISCERN by Risto Miikkulainen proclaimed to be a psychologically plausible story parser, generator and question answerer. Intrigued by this new connectionist approach I started to read and liked the subsymbolic approach and its possibilities. The book about DISCERN was quite a mouthful but I kept being fascinated by the modelling of dyslexia in DISLEX. If this model proves to be accurate, perhaps some day the affliction can be understood better and handled more effectively. The quest was over, I had found the subject of my thesis.

The main question of this thesis is whether DISLEX can handle a new domain. This leads to three other research questions. Will DISLEX map the lexical representations correctly to the semantic representations and vice versa? Do the semantic concepts generated by FGREPNET resemble the original classes? Which words will converge when I attempt to model dyslexic behaviour and do they resemble human dyslexic behaviour? The claims of Miikkulainen are that that DISLEX can mimic several cognitive processes such as deep dyslexia and surface dyslexia and that DISCERN is capable of handling 98% of its input correctly. It all sounds very promising for further research and I wanted to see these results for myself. I had to create a corpus for this purpose and started my experiments. The expectations are that, like

DISCERN, DISLEX will achieve a high percentage of accuracy and that dyslexia will indeed be modelled quite accurately by the system.

The thesis starts with an introduction to some basic concepts: dyslexia, neural and semantic networks. DISCERN is a mixture of the two networks. After this, I will describe DISCERN itself and two of its components, FGREPNET and DISLEX. Next, I will outline my experiment and the results thereof. Some final remarks conclude the thesis.

1.2 Dyslexia

Dyslexia can be defined as follows: a neural disorder that inhibits a person s ability to recognize and process graphic symbols, particularly those pertaining to language (Encyclop dia Britannica 1999). However, there is a tendency to include problems with oral language as well (IDA 1993). The simple definition of Miikkulainen himself gives a clear picture, he states that patients who have dyslexia confuse words that are semantically or visually similar (Miikkulainen in press). Sometimes dyslexia is considered to be a specific form of aphasia, a brain-generated defect in the pronunciation of words (Encyclop dia Britannica 1999). This would explain why certain characteristics of deep dyslexia are shared by aphasic patients. Though dyslexia is primarily a neural disorder, it is often included in the field of psychology since the two fields have so much in common. The distinction between neurolinguistics and psycholinguistics is therefore not always clear. This explains why the dyslexic behaviour of a computer system is said to be psychologically plausible. There are two main subdivisions in the area of dyslexia, i.e. deep dyslexia and surface dyslexia.

1.2.1 Deep dyslexia

Deep dyslexia is an acquired reading disorder marked by the occurrence of semantic errors ,e.g. cat instead of pussy (Plaut 1998). A deep dyslexic patient may read fluently but he does not say exactly what is on the paper. A subject I know, makes mistakes such as reading The dog is ill , while the text reads A doggy is sick . As surface dyslexia only uses the lexical pathway, deep dyslexic people seem to read almost entirely through semantics (Plaut 1998). Deep dyslexia can be an affliction by itself or part of a process such as semantic dementia or Alzheimer s disease (Plaut 1997). Other symptoms include visual and morphological effects in their errors and

an advantage for concrete over abstract words (Plaut 1993). Contrary to surface dyslexia all symptoms are related to the semantic categories (McLeod 2000).

1.2.2 Surface dyslexia

A person who has surface dyslexia has a problem with reading irregular words correctly. Words that obey the normal spelling-to-sound rules of the language are usually fine but other words create problems for the patient (Coltheart 1983). People with surface dyslexia rely almost exclusively on the lexical pathway. Brain damage that reduces or eliminates the semantic pathway shows the inadequacies of the lexical path (Plaut 1998). Sometimes a distinction is made between acquired and developmental surface dyslexia. People with developmental surface dyslexia have no problems with perceiving words when spoken but they do not recognise written words that easily. They are patients of normal intelligence who have to concentrate much harder than the average reader to read the correct words. They mirror the accuracy of a normal human when he is not concentrating (Plaut 1997). We all have dyslexic tendencies sometimes but dyslexic patients have these symptoms all the time.

1.3 Neural networks

In this section I will explain what an artificial neural network is and what the differences are with a biological neural network. The history of neural networks is highlighted and some comment is given on the representation of words and letters. The main problems of neural networks are mentioned and the section is concluded by some examples of neural networks used today.

1.3.1 What is a artificial neural network?

A conventional computer has two main components, a memory and some kind of processing device (Mikhailov 1999). An artificial neural network is made up of a set of simple processing units connected together in a network. These units are called neurons or cells because they have some similarity to biological neurons. The structure of the network allows the user to improve the computer capabilities and to test various hypotheses about information processing in the human brain. An artificial neural network (from now on called a neural network) contains three different kinds of layers: Input, Hidden Layer(s) and Output (Tjong Kim Sang 1998). The connections between the different neurons each have a weight. For instance going from node A to node B would have the weight 0.5 and from A to C the weight would be set at 0.9.

The computer now has metric numbers to compare. If a certain threshold is surpassed, the next node becomes active. So if the input at A is 0.6 the connection directly going to C does not become active but the one going to B does. From there on it might still go to C, depending on the different paths after node B.

Neural networks have many advantages (Mikhailov 1999). Because the cells can adapt themselves in the training phase, the system has the capability of learning. Knowledge is distributed over the system since there is no central memory, which also makes the system quite robust (it can handle faulty information or damage). The structure of a neural network makes it possible to do computations in parallel, giving it massive parallel processing capabilities. Neural networks are especially good at pattern recognition based on learnt cases, so that they can be used for a multitude of tasks.

1.3.2 History

Neural networks in computational linguistics are inspired by the neural networks in the brains of living beings. Research on neural networks was pioneered by Frank Rosenblatt, who constructed a mechanism called a perceptron in the 1960 s (Mikhailov 1999). Neural networks try to mimic certain processing abilities of the human brain. There are indeed several correspondences between a biological neural network and a computational one (Bengio 1996). Both networks learn from examples and therefore can be trained to perform different tasks. In the training examples there may be some faulty information, as long as there are more right examples than wrong ones. The two networks also use distributed processing, so they separate different goals and let different parts of the system obtain an answer. Finally both systems store their information in an associative manner. It is thanks to this approach to information storage that computational networks can make use of the nearest neighbour' rule. This means that if no exact match is found, the network searches the neighbouring nodes for the next best match. Of course there are differences between the two systems. Apart from the point that one exists only in living beings and the other in computer systems, there are disparities in the system itself. In the human brain there are 10^3 synapses per neuron. This means that there are that many pathways (synapses) between the different nodes of information (neurons). If you know that there are approximately 10^{12} neurons or nerve cells in our brain, we get an

impressive product of possible links. Today's hardware can compute massive numbers but the capacity of the human brain is still out of our reach.

1.3.3 Representation

Usually letters are preferred to words when choosing a level to work with. A neural network processes numbers and it is not obvious how it should handle words. There have been some experiments with exact numbers as symbols for the letter but they were not very successful. In this system *a* would be for instance 1.0, *b* 0.75 and c 0.50 (Tjong Kim Sang 1998). The main problem is that if the network hesitates between two choices it takes the average. So doubt about whether we need an *a* or a *c* would result in a *b* since the average of 1.0 and 0.50 is 0.75; this is clearly not what we are looking for. Another quite intricate way of representing letters is by primary numbers. This involves multiplying the numbers and adapting them according to the paths followed and seeing by which numbers the end result is dividable. The most common representation of letters is series of bits, in which each bit corresponds with a cell. In this system for instance *a* is 10000, *b* 01000, *c* 00100 and so on (Tjong Kim Sang 1998).

1.3.4 Problems

Since the probability of a word is the product of the probabilities of its characters, long words have a high chance of being rejected by the system. If you have to calculate the chances of the word "zre" versus the word "indifference", a system that does not consider the word length would give "zre" the best chance of being right. The chance that "indifference" is the more probable answer would be something like 0.2 *0.3 *0.9 *0.8 *0.7 *0.7 *0.5 *0.8 *0.7 *0.3 *0.4 *0.8, or a very small number indeed (Tjong Kim Sang 1998). Therefore it is imperative that a system takes the length of the words into account by for instance multiplying the result with a correction factor, which depends on the word length.

A more practical problem is that neural networks take a very long time to train, sometimes up to four months (Tjong Kim Sang 1998).

1.3.5 Example

Neural networks are often used today in the field of robotics and visual recognition (Mikhailov 1999, Miikkulainen 1993). A neural network can be used, for instance, to

create an intrusion detector that recognizes persons (Ryan 1998). It remembers the particular movements of persons, which it learned during training, and can trigger an alarm when there is an intruder. Neural networks are also used in more linguistic fields such as the recognizing of handwriting, speech recognition and data extraction (Kemsley 1992).

1.4 Semantic networks

This section gives an introduction to semantic networks, starting with the main features of this system. The history of semantic networks is given next, detailing how the idea of semantic networks, thought of centuries ago, finally shaped itself into a computational model. The main problems of this type of network are considered next, followed by an example of a semantic network used today.

1.4.1 What is a semantic network?

Semantic networks use semantic criteria in organizing information. The knowledge found in semantic network is represented as a pattern of interconnected nodes and arcs (Soja 1987). The nodes stand for concepts of entities, attributes, events and states. Graphically, it is usually represented by a circle, an ellipse or a square, depending on the author. Of course these different notations all denote the same thing. The arcs represent the relation between nodes. This relationship can be linguistic in nature (like Agent or Recipient) or spatial, temporal or even logical.

The nodes are hierarchically organised according to levels of generality. This has the effect that different values can be inherited throughout the structure. At the top of such a structure we find semantic primitives such as FOLK, DO, INGEST and THING (Schank 1973; Soja 1987; McDonalds 1987). Schank used a hundred of these top nodes. The network obtained by nesting the different graphs can also be represented in a linear way. A sentence as "The dog greedily ate the bone" (Soja 1987) for instance would give:

```
[eat] -    (Agnt) =:> [dog]
           (Object) =:> [bone]
           (Manner) =:> [greed])
```

Other semantic networks use verb-centred graphs. All other functions are complements to the main verb. Whether these higher order relationships are represented in a linear fashion or as a relation graph does not matter to the computer

system. The graph notation is semantically transparent to us if there are not too many nodes, but the semantic network has its own way of storing data. The computational linguist is once again saved from the discussions about the form of the network and can concentrate on the output, the performance of the programme.

1.4.2 History

When exactly the idea of semantic networks was conceived is not entirely clear. We can hardly consider Aristotle a computational linguist but he did say "Human behaviour is controlled by learnt associations between concepts" and "Concepts are made out of more primitive concepts" (Aristotle quoted in Bench-Capon 1990). This philosophy was carried on by e.g. Hume and Locke (Taels 1996). Several linguistic theories found their origin in philosophical ideas and this particular concept gave birth to semantic networks. Though Schank and Winston were firm advocates of this network, it all began with Quillian in the sixties (Bench-Capon 1990). Since then it has become one of two most popular networks around, together with neural networks.

Artificial Intelligence used to concern itself mostly with problem solving (Nilsson 1971). The goal of Artificial Intelligence is to make computers act as humans (Wepopedia 1999). Teaching a computer to play checkers or chess was the highest goal. And it must be admitted that huge successes were achieved in the field of games. One has only to think of Deep Blue’s victory (not with a semantic network) over the current world chess champion Kasparov to prove this point. Semantic networks however are not interested in solving a particular problem but in the relationships between two.

1.4.3 Problems

Teaching the system causal relationships and entailment is not an easy task (Vossen 1999). It should deduce, for instance, that if a man eats a hamburger, the hamburger is gone; or that when you die, you cannot eat, drink or sleep anymore. It should be able to represent the characteristics of each semantic notion. For instance, a cat can run but it cannot use a computer, while a human can do both. A good system needs to find a way to represent this ’world knowledge’.

A problem that sometimes arises lies in the hierarchy of the semantic nodes (Bench-Capon 1990). For instance, a hinny is a kind of mammal, yet it cannot reproduce. The semantic node denoting hinny should receive all the information from mammals but should bar the ability to reproduce. This is a very dangerous road because a hierarchy that contains blocking may lead to some very strange concepts. If you follow the right paths you could for instance describe a pencil as a kind of human, except that it does not talk, move, breathe, eat and so on.

One of the major disadvantages of semantic networks is that links cannot always be followed in the same way (Bench-Capon 1990). For instance the sentence "Rex is a dog, a dog is an animal, an animal is a living being". This means that Rex is a living being. Consider the next sentence "Antwerp is near Brussels, Brussels is near Liege, Liege is near France". Antwerp is not near France because 'near' is intransitive as opposed to 'is-a' that is transitive. This distinction can be learnt by the machine but there is a more difficult difference. Consider figure 1 (on the next page); we clearly see that the arc going from A to B and the one from B to C are of a different nature. Going from a particular instrument to a general class is not the same kind of link as from a class of instruments to a larger class of general items. The simplicity of these arcs is misleading. The inference mechanism with this many links is actually complicated, considering the different natures of arcs given the same name (such as is-a). The computer has to learn the difference between these by for instance calling them is-a1 and is-a2, which greatly slows down the system. When you want to query the system, easy questions like "What is the relation between Mary and Joseph?", the system will not be too complicated. The network just has to find the two nodes with these names and jump arcs till there is a common node. There is a clear trade-off between complexity of questions and complexity of system. This seems to be quite normal but we have to see whether or not it is feasible to use a network with that many in links with actual data. The research in this thesis for instance uses sentences from a restricted domain as its training data, but just how complex can the relationships be before the network expands beyond usability?

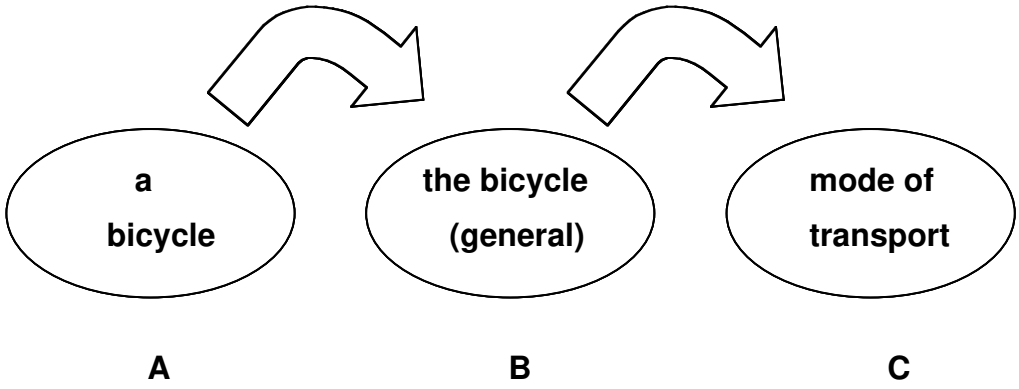


Figure 1.
The two arcs in this semantic graph appear to be the same but are of a different nature. The first is going from a specific item to the general class of those items, while the second arc goes from one class of items to a larger class of items. The computer system needs to differentiate the different types of arcs.

1.4.4 Example

A project still running today is EuroWordNet 1.6 (Vossen 1999). It is a semantic network based on Wordnet by Miller. Wordnet was a semantic network for the English language but EuroWordNet has higher goals. They are working on a network that uses semantic primitives as links between eight different languages. A certain word in one language is linked to a semantic concept and this node searches for the corresponding word in a second language. The major advantage of such a system is that a new language only has to be related to the central semantic nodes and not to all the other languages separately. A word-by-word translation is not possible of course. Therefore each language has its own personal structure with even some metaphorical elements in the system. Finally linguistic and cultural differences must be accounted for (e.g. 'liberal' in English versus 'liberaal' in Dutch). The European Union spends one third of its budget on translators (Van Der Auwera 1998) and new countries keep joining up, so if some of this work could be done by EuroWordNet millions would be saved.

2. Cognitive models for language processing

Cognitive models aim to be abstracted models of the human mind, and since DISCERN claims to be psychologically plausible, it is an excellent example. The structure and goals of DISCERN are given in this chapter, as well as its place in the field of Artificial Intelligence. The two modules necessary to model dyslexia in DISCERN, respectively FGREPNET and DISLEX, are discussed next. The first is used to create the semantic representations while the second is a model of the human lexicon and is therefore able to mimic dyslexic behaviour.

2.1 DISCERN

DISCERN is the network that lies at the heart of this thesis. Though the tests later are conducted with only two of its modules, it is important to understand the goals of the larger network DISCERN. It is a cognitive model, which can be inferred from its structure, and though it is mainly a neural network, it is also based on the semantic features of words. There are some qualities an intelligent entity should have and at the end of this section, we see how DISCERN fulfils most of the requirements.

2.1.1 What is DISCERN?

DISCERN is the name of the programme Risto Miikkulainen has written for his PhD. It is a natural language processing system that gives a clear image of Miikkulainen s mixture of semantic and neural networks (Miikkulainen 1993). It consists of distributed neural networks and handles stereotypical event sequences. The parallel distribution processing system, also referred to as a subsymbolic system, is able to store the narratives, paraphrase them and answer questions about them. This is possible through the parallel placing of the parser, the memory subsystem, the generating and the question-answering component. This creates a network of networks, the output of one is the input for another. An example is that the output of FGREPNET (2.2) is the input of DISLEX (2.3).

Evolution of this sort means that the required mappings change as representations evolve (Miikkulainen 1993). The central idea is that the lexical symbols are transformed into their semantic representations and vice versa. Simply put, the interaction between the two feature maps causes the vectors of the lexical items to

transform into those of semantic words and the other way around. These word semantics, as well as the appropriate script inferences, are extracted from the examples. The machine does not need to be extensively programmed such as is the case with normal semantic networks, but generates representations for words and meanings on its own. Each representation is stored as a specific vector and can be retrieved by activating this representation vector or its direct neighbourhood. The points on the map that activate that specific vector form activation bubble. It is therefore not necessary to give the precise vector, but a vector that activates that bubble. The user of the programme can specify the size of the activation bubbles. This is how dyslexic behaviour can be simulated, the activation bubbles of two (or more) words overlap and the system confuses them. This process is similar to that of other psycholinguists and neuroscientists (Plaut 1997, Mcleod [a.o.] 2000).

2.1.2 How does DISCERN work?

A central part in the DISCERN programme is the hierarchical feature map system (Moriarty & Miikkulainen 1998, Miikkulainen 1992). This system is able to recognise input stories as belonging to a particular script by classifying at three levels: scripts, tracks and role bindings. Especially this last level is very useful in question answering. With simple questions such as "Who did he kiss?", the system just has to locate the right role that is missing. In a hierarchical feature map system, extracting the different roles and tracks out of all the scripts is done automatically and independently for each script from script instantiations. The system itself finds the distinguishing properties of the scripts, no supervision of a knowledge engineer is required. The actual experience of the system is used to reflect the properties of the data. The scripts are put into major categories and finer distinctions appear with the addition of more scripts (Miikkulainen 1991, 1992). This also means that when the system is fully trained, the exact placement of all information is completely obscure to the external observer.

The data is not located in semantic compartments with strict boundaries but is stored in a continuous way with soft boundaries. The most frequently encountered scripts become the most detailed, thus creating an overall structure. This structure is a hierarchical pyramid of feature maps. The number of input lines is severely reduced as the lower levels receive their information from the higher. This also decreases training time considerably compared to single-level feature mapping, in some cases

from nine and a half hours to three minutes (Miikkulainen 1992). The ordering of feature maps in a hierarchical pyramid entails that information retrieval is a constructive process. In symbolic systems the story is represented by a single vector, which finds the allocated data. These systems have a monitor that can check if part of the system fails. If a part of the system fails during testing, a 'block' of knowledge can be replaced or corrected. Due to the unclearness of the storage of information this is a lot harder with DISCERN and with most neural networks. The data cannot be rewritten, the only option is to adapt the weights.

The subsymbolic approach of the system makes it very robust against damage and noise (Miikkulainen 1993). When damaged, the system degrades linearly so that the user has the time to notice something is wrong. Noise is also compensated for since small inaccuracies in the story are corrected by the script model. This particular method of storing data does have the downside that if normal activities in a story happen in an unusual order, the system does not remember this order. So if the grocery-story says that the man first paid and then chose the products, DISCERN will replace this order of events by the stereotypical one.

2.1.3 DISCERN in Artificial Intelligence

Neural networks in general are very popular in the field of Artificial Intelligence (Wepopedia 1999). This is understandable if we look at the requirements an intelligent entity should have according to Mikhailov (Mikhailov 1999). I will briefly state DISCERN s abilities to fulfil these requirements:

- Interact with its environment

DISCERN has a question-answering module that responds to natural language questions. New input can be given and the network adapts itself to it.

- Learn from information and experience

DISCERN is not programmed by a knowledge provider but deducts all its information from the input data. The experience with the training data constitutes its knowledge. This is what makes the system so robust, the generalisation is done automatically because of the structure of the network, not by computer programming to accomplish the same effect.

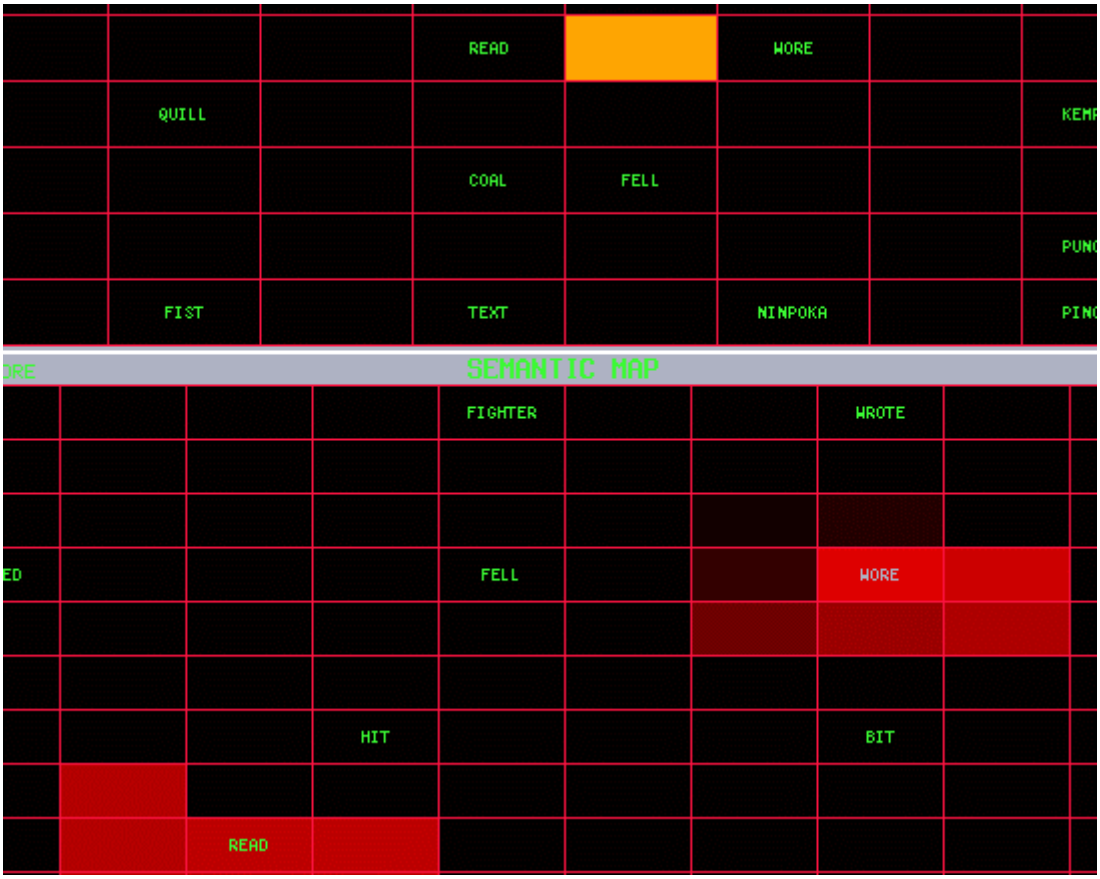


Figure 2.
When DISCERN has to choose between two items, it does not take the average of the two values like some neural networks. The system either chooses between the two items or makes the connections equally strong. In this figure we see that read and wore respond evenly to the chosen block on the top map.

- Make rational decisions

If there is doubt over which decision to make, the highest weight value wins. The strongest connection will always receive priority. In figure two, DISLEX (a module of DISCERN) is faced with the users choice of the space between two items. The problem with neural networks was that when they had to choose between two items, they took the average. As you can see on figure 2, the block of the top map between read and wore is perfectly correlated with the bottom map. The network chooses both READ and WORE, and not a word in the middle. When faced with a choice between two items the network either picks one of the two or both of them. It never gives a totally different correlation.

- Deduce new facts from given ones

DISCERN is able to deduce information that is missing in a story (see next point).

- Make sensible deductions when insufficient facts are available

A major advantage of DISCERN is that it can still function if part of the input is insufficient. The system can take the neutral value 0.5 as the value for this missing piece of information (Miikkulainen 1992). The system can also recognise incomplete stories and fill in the gaps (Miikkulainen 1993, 1992). Only the most salient information is needed to direct a script to a particular class. Sometimes the inferences made by the system are wrong, but this is the same kind of mistake a human would make. Often people add information to stories they are told that is not entirely accurate but very stereotypical. When talking about a trip to Disneyworld for instance the listener may naturally assume that it was a fun stay without the speaker dually mentioning this. It also means that it can answer slightly incorrect questions. If you ask the programme a question like "Who ate lobster?", and the only thing eaten in the story was steak, the system will spot the misunderstanding. If however you ask "Who ate at McDonalds?", while they actually ate at *Le President,* DISCERN cannot answer the question. These two dining places are not close enough to each other in its memory to decide that this was a momentary lapse of the user. This will all only work up to a certain point. If the information is truly insufficient, incorrect answers and false deductions will be the result.

2.2 FGREPNET

FGREPNET is used to find the semantic representations of words. This allows us to create semantic concepts, which can be used in the following module, DISLEX. In this section I will explain how FGREPNET forms semantic representations and what the training data should look like. The input data created for this thesis is fully discussed in section 3.2.

2.2.1 What is FGREPNET?

FGREPNET is a neural network based on the FGREP mechanism (Forming Global Representations with Extended backPropagation). Backpropagation is a supervised algorithm that allows the processing task to be learned from examples of input/output pairs (Miikkulainen 1993). The FGREP mechanism is an extension of backpropagation where the input/output mapping changes as a part of the learning process. FGREPNET is based on a three layer backward error-propagating network (input, hidden, output) and contains a lexicon with the numerical representations of words. At the end of each propagation cycle, also called an epoch, the

representations of input words are modified in the input layer according to the error signal. The new representations are loaded back to the lexicon. The word representations are only meaningful in relation to each other, there are no clearly identifiable features. The initially random representations will change and form categories. This means that representations for items used in the same way become similar after training. Interesting to note is that the algorithm sees the representations as an extra layer of weights (Miikkulainen 1993), which ensures that the most crucial features of each item are represented. The system itself is not attempting to categorise the words but is making the best possible representation related to the task: finding the semantic roles of words in a sentence.

2.2.2 Using FGREPNET

DISCERN uses case-role representation of sentences as an intermediate representation (Miikkulainen 1993). This representation was inspired by the thematic case roles of Fillmore and conceptual dependency theory. Sentences are represented by their main syntactic constituents and their case-role assignment. A simple example is the sentence the judoka threw the chain [1]: (the programme ignores all words but nouns and verbs)

Subject	Verb	Object	With	Agent	Act	Patient	Instrument	Modifier
judoka	threw	chain	_ [2]	judoka	threw	chain	_	_

The case-role representation gives a shallow semantic interpretation of the sentence. This is necessary to model ambiguity in sentences. If we look at the sentence The ninpoka hit the judoka with a sword there are two possible interpretations. Either the sword is used to hit the judoka or the judoka is wearing the sword. The syntactic constituents are the same, but the case-role assignments are different.

Subject	Verb	Object	With	Agent	Act	Patient	Instrument	Modifier
ninpoka	hit	judoka	sword	ninpoka	hit	judoka	sword	_
ninpoka	hit	judoka	sword	ninpoka	hit	judoka	_	sword

[1] All the examples are from the corpus Martial , which I have created for this thesis. For a discussion of this corpus see 3.3

[2] Empty roles are represented by an underscore.

Both parameters and output of FGREPNET are stored in the simulation file. The idea is that you can look at the output and see which parameters were used. This is very convenient when experimenting with parameters such as neighbourhood sizes and learning rates. In the simulation file you can also turn the parameter shuffling on or off during training, thus choosing whether the input sentences should be processed in a random order or not. The network performed better using a different shuffling programme[3]. It is therefore recommended not to rely on the shuffling programme of FGREPNET itself.

2.3 DISLEX

DISLEX is the module of DISCERN that works with the semantic representations generated by FGREPNET. It is a model of the human lexicon and works with a semantic map and a lexical map. DISLEX learns the associations between these maps during training and the correlations are very accurate. When noise is added the system is biased towards making mistakes that mimic human dyslexic behaviour.

2.3.1 What is DISLEX?

DISLEX stands for DIStributed feature map LEXicon and is an extension of the lexicon architecture of DISCERN (Miikkulainen 1993). For this reason the representations used in both networks are the same. It aims at a more accurate modelling of the human lexical system. DISLEX is necessary in DISCERN so that words and their meaning can be recognised and used correctly. For DISCERN to correctly interpret an input story and answer questions about it, it needs to know the syntactic and semantic features of each word. The memory of DISLEX consists of distributed representations of distinct concepts. This way it can represent two important characteristics of the human mind. Localized damage to the brain may result in the loss of semantic categories, resulting in deep dyslexic or aphasic behaviour. This shows that certain parts of our brain do correspond with certain information. DISLEX can model this through the use of semantic and lexical maps. However the severity of a brain injury is caused by its volume rather than its location (Miikkulainen 1993). Different memory traces are spread out over a region so that in the case of partial injury, the other parts are equally capable of producing the data. This is what the neural network in DISLEX does. Each unit and connection stands for

[3] Written by Tjong Kim Sang

different pieces of information. Representation and processing in this system is called subsymbolic because individual units and connections stand for unidentifiable pieces of information rather than for recognizable symbols.

DISLEX makes a distinction between lexical representations and semantic representations. This separation is supported by psycholinguistic evidence. Aphasic patients, for instance, sometimes have problems with semantic categories but have no problem distinguishing lexical components. DISLEX is a model of the human lexical system because it learns the association between the lexical and semantic representations as a human would. Later we will see that when the noise is added, the system is biased towards making mistakes that reflect human dyslexia.

The original DISLEX system allowed access to an episodic memory and non-linguistic features such as a sensimotor memory. These features were not available in the system I have, but since the scope of this research does not include memory or visual images, this does not constitute a problem.

2.3.2 The lexical representation of words

The lexical representations can be either orthographic or phonetic. Orthographic means we take the graphemes, the actual letters, as a basis. We can for instance say that the letters d and b are mirror images of each other. With the phonetic representation, we take the phonemes, the sounds, as a basis. Here we could say that the words slay and sleigh are pronounced identically. It does not matter to DISLEX which one is used as long as the similarities in each system are maintained. This means that loved and moved will be mapped close to each other on an orthographic lexical map but not on a phonetic one. I have used both representations to study the differences in mapping.

2.3.2.1 Orthographic representation

The orthographic representation of a letter is based on the number of black pixels the capital letter has in its bitmap representation in the MacIntosh Geneva font. All letters are given a value between 0.0 and 1.0. The darkness values of each word s letters are then concatenated into a representation vector. This simple technique ensures that words and letters that look alike have a similar representation. Each word consists of a fixed number of orthographic items, which is dictated by the number of

letters of the largest word. In the corpus Martial the largest word is ninpoka , consisting of seven letters. If a word has fewer letters, the non-existent letters are represented by 0.0. If we take the words day and ray for example and limit the vector to seven values (for seven letters) we get:

ray	0.78125	0.53125	0.375	0.0	0.0	0.0	0.0
day	0.68750	0.53125	0.375	0.0	0.0	0.0	0.0

2.3.2.2 Phonetic representation

The lexical feature map can be filled with phonetic representations of the words. The same words are used in both lexical representations and I would like to study what this changes in the final associative mappings. I have used the same phonetic symbols as Miikkulainen, whenever it was possible. The word ninpoka , for instance, would be represented by nInp@Uk@ . Each phonetic symbol corresponds with a vector consisting of five values. For instance the phoneme I , as in bit , is represented by 0.000 1.000 0.250 0.400 0.286. The values are not random but seem to be based on phonetic similarities. Miikkulainen has not specified the relation between these values and phonetic features.

The data that Miikkulainen used did not contain a word with th . Since my data has the word threw in it, I had to create a representation vector for this phoneme. I have taken the average of d and t , which only differ in the third value. When we look at b versus p , we see that once again the difference lies in the third value, so it is likely that that the third value of the vector stands for the feature voice, 0.250 meaning voiced and 0.750 not voiced. Except for the practical exchange of the symbol $ for O , all symbols and representations are the same as used by Miikkulainen in DISLEX. A small difference is that I used the British English pronunciation in my research while Miikkulainen used the American pronunciation.

2.3.3 The semantic representation of words

The semantic representations are generated by FGREPNET (see section 2.2). Input sentences taken from a corpus show the network the syntactic patterns of the different words. In a purely statistical way the network learns which words can be

Figure 3.
Each semantic representation consists of a representation vector. DISLEX gives as the numerical values of these vectors as well as a visual representation.

used as e.g. Agent or Modifier. Words that occur in similar sentences are clustered together. The central idea is that the semantic characteristics of a word can be deduced from their syntactic use. Otherwise put, semantics and pragmatics are the same.

The semantic representation of a word in this system is, like the lexical representation, a vector. For example:
pinched 0.235 0.298 0.486 0.345 1.000 0.328 0.579 0.260 0.235 0.057
samurai 0.845 0.537 0.951 0.467 0.228 0.195 0.346 0.587 0.285 0.258
Each of these vectors is graphically represented with a colour scheme (see figure 3).

Contrary to the lexical representation, the semantic representation vector does not have clearly defined characteristics. It is not clear which part of the vector stands for which semantic feature. Miikkulainen did find out that the fourth value in the vector was related to animateness (Miikkulainen 1993), but even when it was changed, animateness could still be present elsewhere in the vector. This has the disadvantage that the result of the FGREPNET process is completely opaque to the user.

When FGREPNET works properly, several words that are used in the same way get the same representation. This means that the system makes absolutely no distinction between words of the same semantic category. In the type of tasks considered, it would make no distinction between christ and judas because they are both names, or dog and cat because they both belong to the semantic concept PETS. Miikkulainen created the ID+ content technique to solve this problem. Several instances of the same semantic representation are given an extra code, called the ID pattern, at the beginning of the representation vector. This way, words with similar syntactic characteristics can still be distinguished. Unfortunately this technique is not

available to me, so this may become a serious problem for my research. If the semantic representations do not converge we get different semantics cluster as an output of FGREPNET. To make this visible I used a clustering programme[4] with two types of processing. The first was a horizontal tree with branches, thus making it easier for the human eye to discern individual semantic concepts. The second displayed the distances between the different clusters. An example of the two tree representations: is found in the appendix (section B).

2.3.4 Association of lexical and semantic representations

DISLEX needs three kinds of input: one with the lexical representation of all words, one with the semantic representations, and the relationship between the two. The association between the lexical items and the semantic concepts is learned during training. The association also offers the opportunity to show the system that a word is ambiguous. The lexical representation of an ambiguous word is entered twice, linked each time to another semantic concept.

2.3.5 Modelling dyslexia

One of the interesting aspects of DISLEX is that it was not created to mimic dyslexic behaviour. It was just a model of the human lexicon and apparently it was biased towards making mistakes that are psychologically plausible. In order to prove this claim, Miikkulainen gave several letters the same representation in DISLEX and added noise to the input. The system should then make mistakes that mirror the errors made by dyslexic humans. Miikkulainen only wanted to hint at the possibility of modelling dyslexia in DISCERN, so he took DISLEX as an example. The name DISLEX is therefore somewhat confusing since it does not mean that dyslexia is studied within this module.

Miikkulainen gave the most frequent letters the same representation. This means that the lexical representations of the letters E, R and D are the same. So the representation vectors for day and ray would be:

RAY	0.78125	0.53125	0.375	0.0	0.0	0.0	0.0
DAY	0.78125	0.53125	0.375	0.0	0.0	0.0	0.0

The network does indeed give errors that resemble those of dyslexic humans, but the manner of obtaining these results is not psychologically plausible. Dyslexic humans

[4] Hierarchical Cluster Analysis and Principal Component Analysis by Stolcke

only confuse letters that look like each other, otherwise we are more in the area of aphasic patients. The chances of confusing the letters E and R are very slim, because they are not orthographically similar. In my network, the dyslexic behaviour of the network is based on actual psycholinguistic data. The research of Vallaey & Vandroemme (Vallaey & Vandroemme 1997) shows that dyslexic persons often confuse the letters:

b,d and p I and L N, M and W C,G and O

This is because these letters are mirror images of each other or are simply the same symbol upside down. Dyslexic patients have a problem with distinguishing left from right (Vinegrad 1999), so they do not see the difference between b and d, which only differ in the placing of a straight line to the left or the right of the circle. They do perceive the letters but are unable to determine the direction. Most humans have this problem until they are halfway through the first grade of primary school. People with dyslexia are not able to get past this stage and remain unable to perceive the direction of the letters (Vallaey & Vandroemme 1997). So the letters that bare a resemblance to each other, as attested by dyslexic humans, are given the same representation vector in my research. I have used the character groups mentioned above, but only the capital letters since the orthographic representations are based on the words in capital letters.

3. Experiments

This chapter deals with the experiments I conducted to answer the research questions. First, there is some comment on the making of semantic concepts, followed by a discussion of the corpus Martial. I then give my expectations for the experiments, followed by the results. I made a distinction between the experiments to evaluate FGREPNET and DISLEX, and the experiments to appraise DISLEX s capabilities of modelling dyslexia.

3.1 The semantic representation of clusters

We need the semantic representation for the clusters that represent a semantic concept, not just the representation of each word. If each word is given its own semantic concept there is no generalisation, which would make DISLEX psychologically implausible. The clustering programme provided us with several possible clusters. One way to find the representation of a cluster and thus the

semantic concepts it represents would be to calculate the average of the words of one concept. This is perfect for finding the representation vector of FIGHTER:

NINPOKA	0.719	0.281	0.719	0.656	0.688	0.563	0.531
KEMPOKA	0.562	0.750	1.000	0.656	0.688	0.563	0.531
KENDOKA	0.562	0.750	0.719	0.656	0.688	0.563	0.531
FIGHTER	0.614	0.594	0.813	0.656	0.688	0.563	0.531

However, there is a problem when we are faced with ambiguous words. The two words text and haiku can be seen as something to read (WORDS) and as something physical (MATERIAL). Therefore the representations for WORDS and MATERIAL have to be different. The way to reach this result is by making FGREPNET process the semantic concepts themselves.

All semantic classes and semantic concepts are represented by capital letters in this thesis. When creating the input data (corpus Martial), I call a group of words with the same semantic characteristics semantic classes . When FGREPNET has clustered words together they form groups I call semantic concepts . Though these two names in fact denote the same thing in, I find it easier to separate what I have created from what FGREPNET has made. In most cases what I have dubbed the semantic class FIGHTER will result in the semantic concept FIGHTER, but in other cases FGREPNET does not use the same distinction as I did.

3.2 Identifying semantic concepts

We have to process the semantic concepts but there is still one problem, which is picking your semantic concepts. The clustering programme provides the distances between the different semantic representations but it does not give the clusters themselves. We need to choose a threshold value: if the distance between two items is smaller than this threshold, they belong to the same class. I thought of taking the threshold value of 0.1 since that covers the most important divisions in the training data. Unfortunately the largest clusters were discernable but threw and pinched were still one semantic concept. When the distances between the semantic

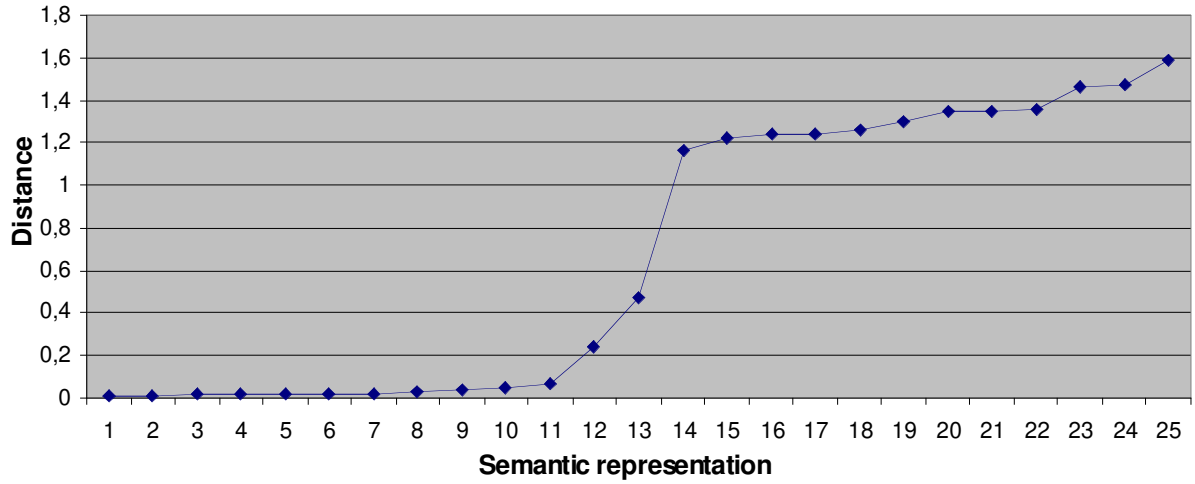


Figure 4.
The line graph shows the distances between the semantic representations of words. It changes radically after eleven, so this is where the boundary for semantic concepts lies. All semantic representations that differ less than this value are part of the same semantic concept.

representations of a particular run are put in a line graph however, a sudden increase can be discerned. The smaller distance values lie close to each other, but at a certain point the values suddenly increase (see figure 4). This sudden change in the line, after 11, is where I have put the threshold for clustering.

3.3 The corpus Martial

In order to familiarise myself with the system I conducted a test with a corpus called Test. It showed the shortcomings in the training data, described later in this section, and which words were useful for my research. This newly acquired information helped me create the corpus Martial. The name refers to topic, the field of Martial arts. This section explains the choice of constituents in the corpus Martial and compares it to Miikkulainen s corpus, which contains sentences from the domain of outdoor living. I will first give the constituents of the corpus Martial.

FIGHTER = "samurai kendoka kempoka ninpoka"
ANIMAL = "snake tiger"
PAPER = "text haiku"
CLOTHES = "kimono shoes"
PENS = "quill coal"
WEAPON = "fist teeth chain sword text haiku"
RWEAPON = "fist chain sword text haiku"
IRON = "sword chain text haiku"
RIRON = "sword chain"

FIST = "fist"
TEETH = "teeth"

All sentences in the corpus Martial are unique and original. Every sentence appears only once, so that there is no manipulation of the data. Sometimes Miikkulainen used repeated sentences to force the network to define the constituents more clearly. Since the network should be able to generalise the data as well as represent unique properties of verbs, I did not use this technique. The results for this corpus will not be as good of those of Miikkulainen but they will give a fair account of the network s abilities.

After working with the corpus Test, some shortcomings became apparent. The main problem was that there were too few semantic concepts. This first test corpus only had five semantic concepts, which were FIGHTER, FELL, WEAPON and two groups of verbs. It was unlikely that the system would make semantic mistakes with so few concepts. The network grouped the animate constituents together but had only one single semantic concept for all the inanimate constituents, which makes it impossible to create the templates for each semantic concept. When we have to train the semantic concepts, we need the templates for it. All we have to do is change the elements of the classes into the semantic concepts. For example:
"FIGHTER" = "samurai ninpoka sensei sempai kendoka kempoka ninja"
becomes "FIGHTER" = "FIGHTER"
Instead of filling the semantic class FIGHTER with its members, it is replaced by a semantic concept, which in this case also is called FIGHTER. Classes with more members undergo the same, e.g.:
"RWEAPON" = "fist chain sword text haiku"
becomes "RWEAPON" = "FIST IWEAPON MATERIAL"

Of course this will only work if the corpus was made with clear semantic concepts in the first place. This is where the corpus Test failed. Especially when ambiguous words are included in the corpus, clear semantic concepts are essential. Two ambiguous words were chosen so that there is a dual mapping from one lexical representation to two different semantic concepts. The words haiku and text can have two meanings. It can be a solid material as in the samurai hit the ninpoka with the text , then it is part of the semantic class MATERIAL, or the actual words e.g. the kempoka wrote a text with a quill , then it belongs to WORDS. If the same class, e.g.

PAPER, represented the two uses of text and haiku , the ambiguity is not solved that easily. Though the classes WORDS and MATERIAL contain the exact same words, it is advisable to make this division. Note that only these classes are identical, thought words such as fist and sword also appear in more than one class. The word sword , for instance, belongs to the class of RWEAPON because you can hit with, just as with a fist or a text. But sword also belongs to the class of IWEAPON since it can be worn. The word fist has many uses in common with sword but you cannot wear a fist . Therefore I had to diversify the classes. For the corpus Martial the class of Animals was added. It is a unique class because the members, snake and tiger , can hit and bite but not read scrolls or wear clothes.

The original class of Fighter had seven members, which was reduced to four. When testing the corpus Test for dyslexic behaviour, it became apparent that several of these words were redundant since they were never confused. New semantic classes means more sentences and to keep down the number of sentences, the number of words should be small. I therefore chose words that would likely be confused by dyslexic readers such as ninpoka , kempoka and kendoka . The word samurai was left in because it has the same semantic characteristics as the other three members while not resembling them too much on the lexical level.

Apart from the verb fell , all the verbs were divided into two semantic concepts, called VERB1 and VERB2. There were only two groups of verbs because the test was done with a very small amount of training data (forty sentences). This convergence means that in order to train DISLEX, we have to teach the programme what the behaviour is of the semantic concept VERB 1. It is impossible to make one description that can be used for diverse verbs as hit, bit and sliced . The first can be with almost all weapons in the object slot, the second only with teeth and the third only with sharp objects. This problem was solved by taking over 1300 sentences as training data for the corpus Martial. The semantic concepts of the corpus Test did show that the verbs should have a distinct meaning, so that they would not converge into a single concept.

The verbs read and write were added because they have the same object slot fillers but only write has an Instrument. In the corpus of 1418 sentences only two of them state that you can write **with** something. According to the claims of Miikkulainen the network should be able to remember this use of the verb even though it is used infrequently.

Though all the sentences are unique to this corpus, two constituents are also present in Miikkulainen s corpus. Those are hit and move . I have chosen hit because of its lexical similarity to bit and because the role of object is optional. I have chosen to use the verb move in my corpus because of its unique characteristics. Any animate subject can move itself and almost all other non-verbs, while being the Agent, e.g. the tiger moved the sword . An inanimate object can also move but then the role of Agent is empty, as in the sword moved . For corpus Martial this would be

Subject	Verb	Object	With	Agent	Act	Patient	Instrument	Modifier
tiger	moved	sword	_	tiger	moved	sword	_	_
tiger	moved	_	_	tiger	moved	tiger	_	_
sword	moved	_	_	_	moved	sword	_	_

A programme working on the frames could generate all sentences but the ones with just a subject and the verb moved. They had to be put in manually since the sentence-generating programme could not deduce the Agent of such a sentence. The programme does not know when the Agent slot should be empty or not. Usually the syntactic sentence is the almost the same as the underlying semantic sentence (except ambiguous sentences) but in this case an extra slot is needed. It is unfortunate that not all the sentences could be derived from the frames but the extra sentences were meticulously checked for errors.

In constructing the corpus Martial several of choices had to be made. I have for instance allowed the sentence the teeth moved but did not include the ninpoka bit the snake . Neither does it contain sentences with quill or coal as weapons. Though it is possible in real life to hit someone with a piece of coal, I have not included such sentences in the training data. The frames use to generate the sentences of corpus Martial are in section B of the Appendix. It is very plausible that, when working with a free text, not all possible uses of a word are found in this training data. The system has to build representations with the data it has. Words used in

different contexts will differentiate themselves easier, thus resulting in more semantic concepts.

3.4 Research questions and expectations

I expect that DISLEX will map all the lexical representations to their semantic representation correctly. Any problems with the semantic representations should be apparent at the FGREPNET level and the lexical representations should pose no problems at all. FGREPNET is reputed to be good at making generalisations while retaining individual traces. The verb wrote is only used with a Patient in 2 sentences out of 1418. FGREPNET should be able to differentiate between read and wrote , though they are quite similar in the training data. There are no verbs that are used in an identical way so the network should create a different semantic concept for each verb.

The semantic concepts found by FGREPNET should closely resemble the semantic classes used to generate the data. No noise is added and the training data is complete, no sentences were deleted after generating them. This means that since all the members of the class Fighter are used in identical sentences, the semantic representation of these words should be the same. The expected tree should resemble the one in figure 5. Since this version has no ID+ content technique, there is a high chance that the members of a class with identical usage will converge. The problem of class convergence and the lack of an ID+ content technique will be addressed when it surfaces. In FGREPNET the adaptation from semantic classes to semantic concepts should not pose any problems since the necessary semantic frames are available.

The dyslexic behaviour of the system with orthographic representations will probably let bit and hit converge, pinched and punched , most members of the concept FIGHTER and later the words wrote and wore . With the phonetic representations the same lexical items should converge and perhaps some of the shorter words. I expect the semantic convergence of all members of a single class. I do expect that system will let some words converge that have very little in common. The system is biased towards making mistakes that resemble dyslexic errors but it is not a programme written for the study of dyslexia. I expect that the larger part of the errors can be explained by surface dyslexia or deep dyslexia.

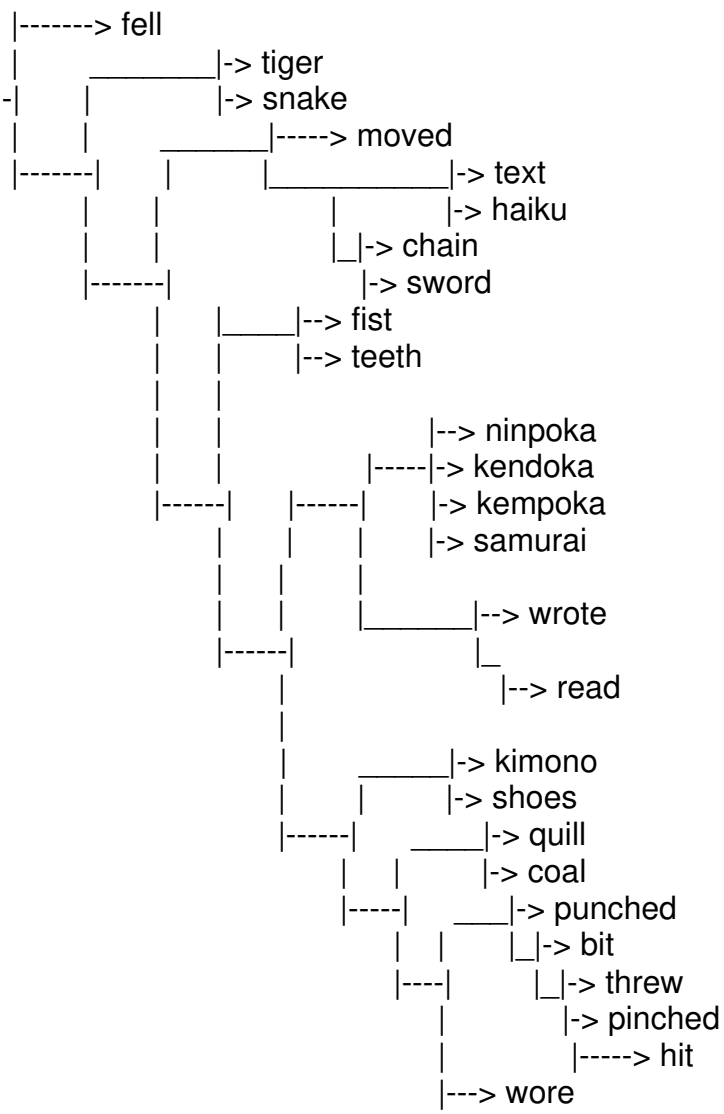


Figure 5.
The expected distances tree. The semantic representations of semantically related words should lie very to each other. Therefore this is the tree that is expected when FGREPNET processes the sentences of the corpus Martial

3.5 Results

I have first tested the abilities FGREPNET and DISLEX with a new domain by letting it work on the sentences of the corpus Martial. The results of this testing and comment on the system are given in the first part of this section. The second part deals with the modelling of surface and deep dyslexic behaviour in DISLEX.

3.5.1 Processing corpus Martial with FGREPNET and DISLEX

I have taken the corpus Martial and ran it through FGREPNET, which gave me the semantic representation of the words. I then used the clustering programme to see

which words belong together and should constitute a semantic concept. After training FGREPNET to teach it the characteristics of the semantic concepts, I could start with DISLEX. I ran a series of tests, some with the orthographic representations, and some with the phonological representations. I tested the same training data several times to see the differences. All of this was done with the standard settings. The remainder of this section deals with some comment on my findings.

3.5.1.1 Modular error correction

Contrary to the expectations, the members of a class never fully converged. There was no need for the ID+ content technique. Words that were generated as one class in the corpus are sometimes separate semantic concepts in the output. For example *snake* and *tiger* were the two elements of the semantic class ANIMAL in the corpus. In the training data they occur equally often in the same contexts. If we look at the results from FGREPNET however, we see that they are not clustered together. This means that when processing the semantic concepts I have to make two concepts called SNAKE and TIGER. When DISLEX uses the orthographic representation it maps *tiger* and *snake* close together but there is no convergence between the two. Theoretically DISCERN has modular error correction, meaning that a mistake in one module, is corrected by the following. When using this data, I only perceived this process once and even then at an intermediate point in the training phase. When the network is trained in DISLEX, SNAKE and TIGER are mapped onto the same semantic space. This means that the system regards the two as forming one semantic class, which was the case in the training data. DISLEX corrects the error made by FGREPNET. Unfortunately, after a hundred epochs the classes are split up once again and after 600 epochs (globally the best performance), they are two completely different semantic concepts. DISCERN consists of nine modules so I cannot evaluate that system, but there is almost no error correction from FGREPNET to DISLEX.

3.5.1.2 Training the network

Typical for neural networks is that longer training does not necessarily mean better results. The chances of convergence go up and the average error fluctuates. An example is that in my research with the corpus Test, the system could separate the natural weapons (teeth, fist, heel) from the iron weapons (chain, sword, star) after 300 epochs but not after 600 epochs. By then they have merged into one semantic

concept WEAPON. This overgeneralization makes it impossible to create correct templates. There has to be a division so that we can make templates such as the ninja wore a sword and exclude the ninja wore a fist . There is no golden rule to the number of epochs that is best. With a large number of sentences the average error fluctuates. The lowest average error always produces the most functional trees. Another example: At epoch 350 the verb pinched is placed in a branch that is very near the other verbs. After 400 epochs pinched is separated from these verbs and is given a branch of its own, thus obscuring the semantic similarities that all verbs share. After 450 epochs it is placed back with the other verbs.

When dealing with the semantic concepts the situation changes. There are fewer sentences and fewer words so the level of accuracy increases. The training of the semantic concepts of the corpus Martial stays almost stationary after 250 epochs. The network still becomes more accurate but the improvement is very small. There never appears more than a 0.0004 difference in average error between epochs 250 and 600 compared to an average of 0.013 in the previous experiments.

3.5.1.3 Error rate versus performance

The error rate correlates directly to the performance. The lowest error rate gives the best clustering tree. I have tested the same input file thrice with FGREPNET, shuffling the input sentences each time. The lowest error rates of each run were 0.0224, 0.0226 and 0.0265. When looking at the clustering trees it became clear that the first two were nearly identical. They differed only slightly, for example the first did not cluster chain and sword together while the second separated snake and tiger . The third tree however was not utilizable, since the correct semantic concepts could not be derived from it. I have also compared all the trees of a single experiment. This means taking the output of FGREPNET at seven different points of the experiment, with an interval of fifty epochs. Once again, the tree that was most productive corresponded to the epoch with the lowest average error.

3.5.1.4 The statistical approach

DISLEX aims to give an abstracted representation of the human lexicon and some characteristics are biologically plausible. As a test I used a statistical programme[5] to get the same results as FGREPNET. This way I could be certain any error would be

[5] Written by Tjong Kim Sang

due to the system and not the training data. The statistical programme was faster (about fifteen times) and provided a perfect clustering tree. It is also much easier to work with the statistical programme than with FGREPNET. The constant duality between computational models and cognitive models appears here. A cognitive scientist tries to understand the human mind by modelling it on computers (Mikhailov 1999). The computational view is that the machine has to get results, the way it gets those results is not important. Both approaches are used in Artificial Intelligence. But what do we take as the meaning of intelligence? A possible definition is Intelligence is a measure of success of an entity in achieving its objectives by interaction with its environment (Mikhailov 1999). If only the result is important and not the paths to this result, we have to admit that the statistical programme is much more cost effective than FGREPNET. But DISCERN aims to be a cognitive model and therefore uses hierarchical feature maps and neural networks. Only a cognitive system could provide a possible model for dyslexia. The aim is not just to get a computer system that can mimic dyslexic behaviour because a statistical programme could easily do so. The idea is that the system learns in a similar way to humans and produces errors for the same reasons humans do. The hierarchical feature maps are a plausible model of the human lexicon and are therefore relevant for the better understanding of dyslexia in human patients. The system, although a simplified model of a biological neural network, displays dyslexic behaviour while remaining psychologically plausible. This is vital to the understanding of DISCERN and therefore DISLEX, but what is the position of FGREPNET? If we feed the perfect tree the statistical programme gave us to DISLEX, we have perfectly discernable semantic categories and a lower average error. To study dyslexia in DISLEX, I think we should keep the idea behind DISCERN as a whole in mind. Since it is not the goal of DISCERN to be the best system but to be a psychologically plausible system, I think FGREPNET should still be used. It works with a neural network based on the human brain and we know that a purely statistical programme is not psychologically plausible. FGREPNET should stay a cognitive model and should not be replaced by a purely functional programme, this would corrupt cognitive view of DISCERN.

3.5.1.5 Model of human lexicon

The model is in accordance with the symptoms of aphasic and deep dyslexic patients. With some types of aphasia a certain semantic concept or related concepts may be lost. A patient may forget the names of things to write with (the semantic concepts PENS) while retaining all other mental capacities. In the case of deep dyslexia semantic errors occur. Patients then chose a word that is related to the one they mean. For instance kimono is chosen instead of shoes or even teeth

instead of bit . It is however tempting to explain all mistakes of the system by classifying them as aphasic or deep dyslexic behaviour. When we add the characteristics of surface dyslexic behaviour, practically all errors can be accounted for. If a phonetic representation of hIt is visually confused with tiT and this last one semantically confused with fist , we have a psychologically valid explanation for the semi-convergence of teeth and fist. There can be visual-then-semantic errors and semantic-then-visual errors. But there is one caveat: **all** sentences in the corpus Martial are about martial arts. This entails that all the words are related to the same subject. I have only mentioned the words that are either visually or semantically similar to each other in my research of dyslexic behaviour. Some words are both visually and semantically similar, which doubles the odds of (semi-) convergence. The different combinations of errors were not mentioned because, as I stated, there is an explanation for almost every possible error.

3.5.2 Modelling dyslexia in DISLEX

As stated before, the system should exhibit mistakes that resemble dyslexic behaviour when the neighbourhood sizes are increased. DISLEX can mimic dyslexic behaviour because the representations are subsymbolic, category membership is not a yes/no matter but a matter of degree. When the activity bubbles increase, the categories intertwine. When using the orthographic representations in the lexical map, some letters are given the same representation. Apart from the lexical maps and semantic map themselves, the association between two may result in dyslexic behaviour. The best results appear when the network stays under a hundred epochs. Naturally the dyslexic behaviour is different when working with orthographic and phonetic representations and I have therefore done experiments with both representations. When working with the orthographic representation I have used the character representation based on dyslexic patients (Vallaey & Vandroemme 1997). Several letters resemble each other but I have chosen to give the following groups the average of their representations: C & G, I & L, M & N and R, B & P. I had to choose between the groups M, W & V and M & N. Although dyslexic patients confuse these groups, it is not an associative process, the letters W and N are never confused. I choose M & N because those occur more frequently in the corpus. This gives the best effect, while still remaining psychologically plausible.

Normally the semantic and lexical neighbourhood sizes are both at four for the first hundred epochs and are then set to one. I have decided to change both equally for the orthographic representations. So noise is added by increasing the neighbourhood sizes of the lexical and semantic representations and keeping them constant. When the neighbourhood sizes are decreased after fifty epochs, the system shows its robustness by not displaying any obvious signs of dyslexic behaviour. Therefore the neighbourhood sizes are kept constant. When the phonetic representations were used with the lexical map and the neighbourhood sizes were kept equal, the lexical items did not fill the map but stayed close together. I have therefore chosen to start the run with a large neighbourhood size and decrease it after fifty epochs. I will first discuss the ability of DISLEX to model surface dyslexia, followed by a section about deep dyslexia.

3.5.2.1 Surface dyslexia in DISLEX

This is a test to see how well DISLEX can model surface dyslexia. In this section we examine the errors DISLEX makes, with regard to the characters that make up words. People suffering from surface dyslexia do not use their semantic pathways so the relationship between the lexical items on the lexical map is studied. Since surface dyslexia is a condition where the letters of a word are confused, only the orthographic representations are studied in this section.

When the system is training, it seems to collapse after ten epochs. All the representations converge into one single representation. One value for all lexical items is associated with one semantic representation that holds all semantic concepts. Fortunately this situation changes and the system distinguishes the representations again between epoch fifty and seventy.

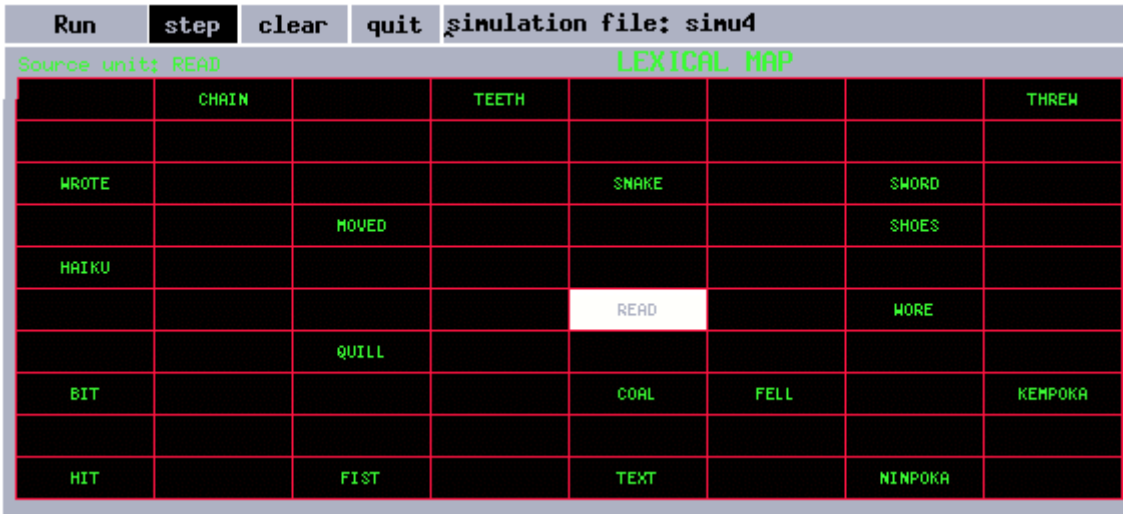


Figure 6.
The items of the lexical map with orthographic representations are ordered by orthographic similarity. Because all words are represented by a representation vector of eight values (for eight letters) and an empty slot is represented by the null value, word length is the most important feature. The words are grouped according to length rather than common letters.

When the neighbourhood sizes are increased from its default value of four to ten, several convergences occur. The words kempoka and kendoka converge; they form one lexical item. The two words were added because of their similarity in letters and length, so the system reacts as expected. The lexical items bit and hit converge. Since they are the only two words with three letters and they only differ in one letter, the convergence was predictable. The lexical item shoes converges with sword , and coal with fell . In both cases the only similarities are the word length and a common letter. It is striking to see that the system focuses on words with the same amount of letters, due to the layout of the lexical map. As you can see on figure 6, all lexical items are grouped according to word length. Since the increase of neighbourhood sizes means that the lexical items converge with their neighbours (they have the most in common), it naturally follows that DISLEX will focus on word length when trying to mimic surface dyslexia.

Dyslexic patients do confuse words of equal length but, as far as surface dyslexia is concerned, coal and fell for instance should resemble each other more closely. I had expected wrote and wore to converge by now since they have three letters in common. Apparently they are not mapped close too each other, probably because they do not have the same amount of letters. Because all words are represented by a representation vector of eight values (for eight letters) and an empty slot is

represented by the null value, word length is the most important feature on the lexical map.

When the neighbourhood sizes reach twenty, the words wore and read converge. These words have two letters in common (in the same order) and have the same number of letters. The convergence is plausible since they have half their letters in common. With a neighbourhood size of twenty-five, the system gives two more convergences: pinched and punched , and snake and teeth . The first pair is psychologically plausible because they have all letters, bar one, in common. The convergence of snake and teeth is slightly more problematical since they have only one letter in common but do have the same number of letters. As we increase the neighbourhood size, the system classifies the lexical items according to word length and looses its ability to give plausible convergences.

3.5.2.2 Deep dyslexia in DISLEX

If we want to study deep dyslexia, we can use the lexical map with phonetic representations to test the connections between the maps. A dyslexic person would not confuse the words pinched and punched when speaking simply because they resemble one another lexically, since the lexical pathway is not used that much. The patients do make visual errors when they are related to the semantic categories so the relationship between the semantic map and the lexical map must be studied as well. A person who has deep dyslexia would confuse the words haiku and text since they belong to the same semantic concept. He might even confuse text and sword since both are used to hit with. The relations between concepts on the semantic map are therefore very important.

a) Semantic map

The most important convergence is the one between TIGER and SNAKE, which happens as soon as the testing is started (neighbourhood sizes at ten). This is actually what I hoped for since they are created by the same semantic class ANIMALS. The semantic concepts TEETH and FIST also converge. They were in the same semantic class called NWEAPON (natural weapons) but have unique characteristics as well. As hoped for, the system does keep them apart when working properly but when the neighbourhood sizes are more than doubled, it begins to

cancel out the smaller differences. Both nouns are so much alike the convergence is plausible.

When the neighbourhood sizes are set at twenty, the concepts IWEAPON and MATERIAL converge. They were already very close to each other so the convergence was expected. Both concepts denote the same thing, that is an object to strike with. MATERIAL contains the elements *haiku* and *text* but only when they are used as a weapon (or at least to hit someone with). It is therefore plausible that this concept converges with the semantic concept that holds all other weapons. Strangely enough the semantic concepts PINCHED and PUNCHED converges at smaller neighbourhood sizes than their lexical counterparts. Their semantic representations were fairly close when FGREPNET generated them and now they have converged. They only had minor differences in use, PUNCHED having a possible Instrument (*fist*). Since the majority of the training data contained sentences wherein the two verbs were used identically, I once again find this a psychologically plausible convergence.

The semantic map, as stated before, remains quite accurate, even at neighbourhood sizes of thirty. The only new convergence is that of PINCHED/PUNCHED and THREW. Intuitively this seems a strange convergence since they appear to have quite different connotations. However, this was already a problem when working with FGREPNET, *pinched* and *threw* kept appearing as a separate semantic category. Since the system does not consider the actual meaning of a word but only its use, we have to conclude that they are after all very similar. When you take the philosophy that pragmatics is semantics, PINCHED and THREW belong closely together.

b) Lexical to semantic

When we look at the association from the lexical map to the semantic map with a neighbourhood size of ten, we see that the lexical items *wore* and *read* respond to

Figure 7.
The words wore and read have semi-converged. When one item is chosen on the lexical map, the semantic counterparts of both items respond equally strong. The system does not consider them the same word but it can no longer distinguish their associations.

the semantic concepts WORE and READ equally strong. I will call this process semi-convergence. Figure 7, taken from DISLEX, shows the effect of semi-convergence. In this image the lexical item wore is chosen on the lexical map. The image shows that WORE and READ lit up equally bright on the semantic map. The same image appears when read is chosen (it should be noted that DISLEX uses a colour scheme to illustrate the strength of the connections: the lighter the colour, the stronger the connection). The two words have the same number of letters and have two letters in common, which also explains why they later converge on the lexical map. This close lexical mapping makes the associations to the semantic map the same. The semantic concepts are both verbs but that is about all they have in common. The training data has no sentences with anyone wearing a text or reading a sword . The lexical proximity dominates over the semantic characteristics.

The lexical item chain connects equally strong to IWEAPON as to MATERIAL. This is quite interesting since the function of MATERIAL is often the same as IWEAPON. The semantic concept MATERIAL contains the words haiku and text when they are used as weapons (or at least to hit someone with); so it seems natural that this concept converges with the concept for all other weapons, i.e. IWEAPONS. Most lexical to semantic associations remain accurate but the strength of the connections lessens.

As the neighbourhood sizes increase, the associations from the lexical map to the semantic quickly deteriorate. With neighbourhood sizes of twenty, fell semi-converges with FELL, PENS and WORDS. This is not very plausible since the semantic concepts PENS and WORDS stand for nouns while FELL is a verb. There are no sentences in the training data where the words fell , haiku (in the sense of a poem) and quill are used in a similar fashion. In interesting situation is that of snake , chain and wrote . These three lexical items are mapped very close to each other and thus influence one another s connections to the semantic map. The word snake is associated correctly but chain takes over the associations of the two words it is surrounded by. This means that chain reacts equally strong to SNAKE/TIGER, MATERIAL/IWEAPON and WROTE. The word wrote connects to WROTE but also to MATERIAL/IWEAPON with equal strength. The connections are obviously becoming corrupt and are no longer psychologically plausible. If the neighbourhood sizes are increased even more, the associations become extremely weak and unreliable.

c) Semantic to lexical

The semantic to lexical connections reflect the relations described in the previous section. It is too bad that the software rarely allows the user to find the associations of semantic concepts. The semantic concepts that did respond remain amazingly accurate throughout the test. The connections do become weaker but remain correct. In some cases there is semi-converge as in WORE and READ that respond to wore and read equally strong. This is just the reflection of the process in the lexical map and the lexical to semantic mapping. I do not believe that the semantic concepts WORE and READ are so much alike in use that they influenced the lexical items. I therefore hold on to my earlier explanation of the lexical semi-convergence.

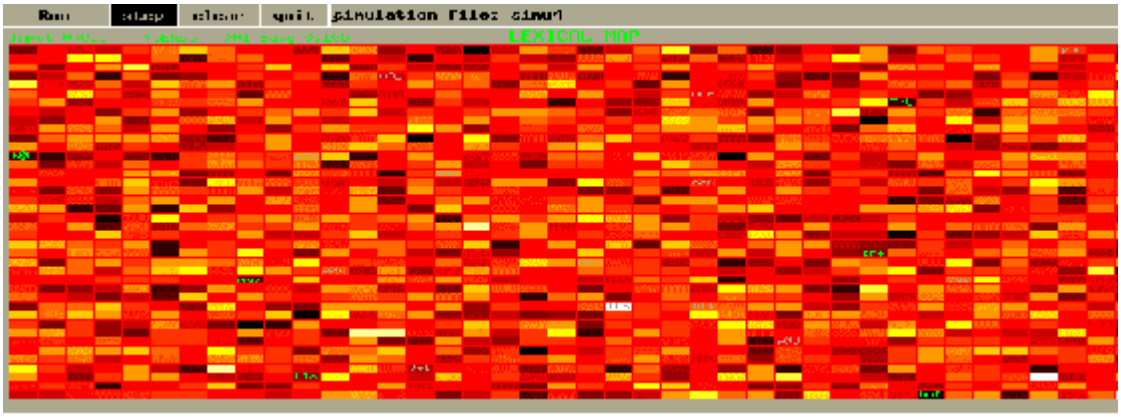


Figure 8.
The lexical map with phonetic representations is much larger than the one with orthographic representations. Because a representation vector of forty numbers is used to represent each lexical item, the items are spread out over the map, which makes convergences impossible. The semantic map is much smaller and has completely converged before the lexical items come close to each other.

d) The phonetic representations

If we compare the findings of the system with the orthographic representations and those of the phonetic representations we find that the latter is of less use to the study of dyslexia. Several errors were the same as with the orthographic representations and of course the semantic map remained the same. The lexical map looks entirely different when using phonetic representations. With the orthographic representations, bit and hit were the only words with three letters. With the phonetic representations there are seven words with three phonemes, with newcomers such as sOd (sword), red (read) and tiT (teeth). The map also looks completely different because the lexical representation consists of vectors of forty numbers instead of twelve (see figure 8). This has the disadvantage that the lexical items are spread out immensely and that convergences are quite hard to obtain. As said before, starting with a large neighbourhood size and reducing it halfway through the run gave the best results. If we do this we do get some interesting webs of associations. The semantic concept THREW, for instance, responded equally strong to pIn∫ed (pinched), pun∫ed (punched) and TrU (threw). This predicts the later convergence of the semantic concepts PINCHED, PUNCHED and THREW. The semantic concept PINCHED responds equally strong to the same three lexical items but PUNCHED does not. Again reflecting the finds of FGREPNET, PUNCHED does not connect to TrU but only to pIn∫ed and pun∫ed . Another interesting find is that the phonetic lexical item tekst (text) responds equally strong to WORDS and

IWEAPON and slightly less strong to MATERIAL. This alludes to the convergence of MATERIAL and IWEAPON but ironically tekst responds stronger to the erroneous semantic concept.

4. Conclusion

Every system has its faults and merits. DISLEX is the start of subsymbolic neural networks but it sets high goals. Some goals are achieved as I stated in previous sections but there are some drawbacks to the system. I will explain some of my criticism on the system in general and add some warnings about the use of FGREPNET and DISLEX. The answers to the research questions are stated briefly and my opinion about the future use of DISCERN conclude this thesis.

The largest problem is the training data. It took quite a while to create a corpus that generates a productive clustering tree. Because the network requires a large data set, I had to create templates myself. This does make it easier to evaluate the processing of the material with FGREPNET but it is a long way from this kind of input and actual texts. DISLEX is but an attempt of modelling the lexicon but this will make it very hard to deal with free texts. When I accidentally put in one erratic sentence in the training data, it had quite an impact on the result. One sentence in a training set of 1300 changed the average error rate considerably. I have refrained from testing the level of noise that the system can handle in the training data because without noise it was already difficult to produce clustering trees that would result in semantic concepts. At the other hand, when testing the system for dyslexic behaviour, we see that the lexical and semantic maps do not converge until the neighbourhood sizes are dramatically increased. As long as the neighbourhood size is made small halfway through the run there is no full breakdown till the default size (four) is multiplied by forty, which makes the system quite robust from this perspective. The results also deteriorate gradually so that changes are quite predictable. We also have to consider that DISLEX is a subdivision of DISCERN. In 3.5.1.1, I have given an example of modular correction. DISCERN works with nine modules so there the possibility of mistakes being corrected by the following modules.

Another problem is that you have to feed FGREPNET and DISLEX a lot of information manually. The system does not have its own clustering programme and the semantic representations need to be copied from FGREPNET to DISLEX. When

the semantic representations of the words are created in FGREPNET, you yourself have to write the correct frames for each semantic concept. It would have been preferable that FGREPNET could generate its own semantic concepts so that we did not need to give it a precise manual of the semantic concepts. Miikkulainen recognises this shortcoming but does not see a solution. I do not know how this problem could be solved without the aid of some kind of symbolic monitor or human operator but I do think that it takes away a lot of the self-organizing quality of the system. The method I used to create the boundaries of semantic concepts could be employed automatically by the system. That way, the user still has to give the concepts names but at least the programme already discerns the different semantic concepts. It would be another step towards a completely self-organizing system.

There are two smaller shortcomings in the system. As I mentioned the programme called DISLEX was never really tested for psychologically valid dyslexic behaviour. The book explains a lot about the possibility of the system but it never mentions an actual test. When the lexical representations were studied, I found that the most frequent letters in the English language were given the same representation. This resulted in the convergence of words but is not in accordance with human dyslexic behaviour. When I asked Risto Miikkulainen why he had not uses more plausible dyslexic behaviour, he stated he only wanted to hint at the possibility. This was never put clearly in his articles and the name DISLEX is also somewhat confusing in this respect. The second strange occurrence happened when I used the shuffling option in FGREPNET. FGREPNET has a higher error rate when the data is shuffled and an even higher error rate when an outside shuffling programme is used. The only conclusion possible is that the shuffling programme incorporated within DISLEX is not trustworthy. I therefore advise to use an external shuffling programme.

If we look at the research questions, we see that all of them are answered. DISLEX does map the lexical representations correctly to the semantic representations, but needs to be fed a lot of information. The semantic concepts of FGREPNET closely resemble the original semantic classes, but the training data is artificial and processing free texts with FGREPNET is still far off. Even with a corpus designed for this system, words with the same semantic features are not always clustered together. Words of equal length converge when dyslexic behaviour is mimicked, resulting in some plausible convergences but also some improbable.

I would not recommend this programme to someone who is just interested in results. Miikkulainen said it could rival the best symbolic systems and proves it with impressively low average errors. My personal experience is that it still has a long way to go. It takes fairly long to understand how to work with the system and I only worked with two modules. The training data takes a long time to perfect, which makes it improbable that this system will be used on actual texts any time soon. Getting the perfect tree out of FGREPNET is quite hard and DISLEX has to be fed too much information. It must be said that DISLEX is very robust and gives accurate results under a large amount of noise but FGREPNET is quite intolerant to noise. If I compare the results of the two modules with those of statistical programmes, the results of the statistical systems are better and faster

As a cognitive model however, DISCERN is quite good. It gives a plausible model for several psycholinguistic phenomena. The modelling of deep dyslexic behaviour is especially good. When errors occur in the semantic map or in the connection between the two semantic and lexical map, they are in accordance with the symptoms of deep dyslexic behaviour. Unfortunately it is not the other way around, not all deep dyslexic symptoms can be mimicked by DISLEX. The better results with concrete words than abstract notions, for instance, could not be explained by this model. The modelling of surface dyslexia is not bad when the neighbourhood sizes are kept rather small. However, too much emphasis lies on the length of words. Lexical items that resemble each other and have the same amount of letters are confused but similar words with a different word length are almost never confused. The lexical map might not be adequate for modelling the lexical items in the human lexicon.

I believe the future of DISCERN, and DISLEX in particular, lies in the cognitive area. I have only studied the dyslexic qualities of this system and they are quite plausible. It is a promising theory but still needs some adjusting. If other psycholinguistic phenomena can be explained in a similar fashion, DISLEX may appear as a very close model of the human brain and all its wonders.

To fully investigate the wider definition of aphasia, so not just deep dyslexia, several subjects should be put in a single corpus. In the corpus Martial, as stated before, all semantic concepts are related to each other in some way because they are all taken from the domain of martial arts. If the corpus has sentences about ten different areas such as shopping, hair styling, a circus and martial arts, the semantic concepts would not be so closely related. If these areas contained words with equal length and similar semantic meanings, the aphasic possibilities of DISLEX could be researched. To create such a corpus and train it would take quite some time and as such is beyond the scope of this thesis.

Bibliography

Bench-Capon. *Knowledge Representation.* Academic Press, 1990. The Apic Series 32.

Bengio, Yushua*. Neural Networks for Speech and Sequence recognition.* International Thomsom Computer Press, 1996.

Cotheart,M., Byng, S. [a.o.]. Surface dyslexia*.* In: *Quarterly Journal of Experimental Psychology*, 1983, p469-495.

Encyclop dia Britannica. *Dyslexia.* 1999. http://www.britannica.com

International Dyslexia Association, The (IDA). *Basic Facts about Dyslexia: What Every Layperson Ought to Know.* 2nd edition, 1998. http://www.interdys.org

Kemsley, David. An overview of neural networks. *International Journal of Neural Networks*, vol. 2, 2/3/4, 1992, p123-133.

McDonalds, D. Natural language generation. In: Shapiro, Stuart (ed.). *The Encyclopedia of AI.* John Wiley & Sons, 1987, Volume I.

McLeod,P., Tim Shallice & David Plaut. Attractor dynamics in word recognition [ ]. In: *Cognition,* 74. Elsevier Science, 2000, p91-113.

Miikkulainen, Risto. Self-organizing process based on lateral inhibition and synaptic resource redistribution*.* In: *Proceedings of the International Conference on Artificial Networks*. Elsevier, 1991.

Miikkulainen, Risto. Script Recognition with Hierarchical Feature Maps*.* In: Sharkey, Noel (ed.). *Connectionist Language Processing.* Kluwer Academic Publishers, 1992.

Miikkulainen, Risto. *Subsymbolic Natural Language Processing: An Integrated Model of scripts, Lexicon, and Memory.* MIT Press, 1993.

Miikkulainen, Risto. *Dyslexic and Category-Specific Aphasic Impairments in a Self-Organizing Feature Map Model of the Lexicon.* University of Texas, in press.

Mikhailov, Ludi. *CT343/443 Intelligent Decision Technologies.* Department of Computation, University of Manchester Institute for Science and Technology, 1999.

Moriarty, David and Risto Miikkulainen. Hierarchical evolution of Neural Networks*.* In: *Proceedings of the 1998 IEEE Conference on Evolutionary Computation*. IEEE, 1998.

Nilsson, Nils. *Problem-solving methods in Artificial Intelligence*. McGraw-Hill book company, 1971. McGraw-Hill Computer science series.

Plaut, D.C. Computational Modeling of Word Reading, Acquired Dyslexia,and Remediation. In: *Converging methods in reading and dyslexia*. MIT Press, 1998.

Plaut, D.C. Structure and Function in the Lexical System: Insights from Distributed Models of Word Reading and Lexical Decision. In: *Language and Cognitive Processes*, 12, 5/6, 1997.

Plaut, D.C. & Shallice. *Deep dyslexia: A case study of connectionist neuropsychology.* In: Cognitive Neuropsychology 10, 1993, p377-500.

Ryan, Jake, Lin Meng-Jang and Risto Miikkulainen. Intrusion detection with neural networks.In: Jordan, M. I., Kearns, M. J., and Solla, S. A. (ed.). *Advances in Neural Information Processing Systems 10 .* MIT Press, 1998.

Schank, Roger and Colby. *Computer Models of Thought and Language.* WH Freeman & Company, 1973.

Soja, *J.* Semantic networks. In: Shapiro, Stuart (ed.): *The Encyclopedia of AI*. John Wiley & Sons, 1987, Volume II.

Taels. Lecture within the course *Moraalfilosofie* at the University of Antwerp, 1996.

Tjong Kim Sang, Erik. *Neural Networks*. Lecture at CLIF meeting in Antwerp, April 2nd, 1998.

Vallaey,M. & G. Vandroemme. *Psychomotoriek bij kinderen.* Acco, 1997.

Van Der Auwera, Johan. Lecture within in the course *Taaltypologie* at the University of Antwerp, 1998.

Vinegrad, Michael. *Dyslexia test.* Adult Dyslexia Organisation, 1999. http://www.futurenet.co.uk/charity/ado

Vossen, Piet. *EuroWordNet*. Lecture at CLIF meeting in Antwerp, May 22nd, 1999.

Wepopedia, online encyclopedia dedicated to computer technology. Artificial Intelligence. 1999. http://www.internet.com

Appendix

A. Frames

Capital letters denote word classes. The word articles and the word with are not used in the actual corpus.

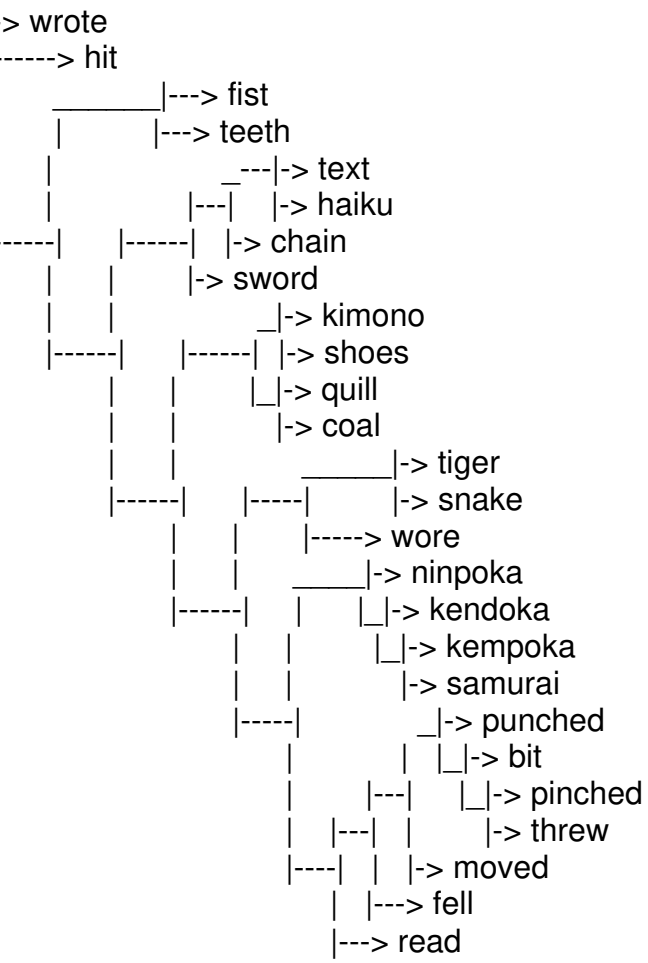
FIGHTER = "samurai kendoka kempoka ninpoka"
ANIMAL = "snake tiger"
PAPER = "text haiku"
CLOTHES = "kimono shoes"
PENS = "quill coal"
WEAPON = "fist teeth chain sword text haiku"
RWEAPON = "fist chain sword text haiku"
IRON = "sword chain text haiku"
RIRON = "sword chain"
FIST = "fist"
TEETH = "teeth"

FIGHTER fell _ _ _
IRON fell _ _ _
ANIMAL fell _ _ _
CLOTHES fell _ _ _
PAPER fell _ _ _
PENS fell _ _ _
FIGHTER bit _ _ _
ANIMAL bit _ _ _
FIGHTER bit FIGHTER _ _
ANIMAL bit FIGHTER _ _
FIGHTER bit FIGHTER TEETH _
ANIMAL bit FIGHTER TEETH _
FIGHTER bit FIGHTER _ PENS
FIGHTER bit FIGHTER _ IRON
FIGHTER bit FIGHTER _ CLOTHES
FIGHTER hit _ _ _
ANIMAL hit _ _ _
FIGHTER hit FIGHTER _ _
ANIMAL hit FIGHTER _ _
FIGHTER hit ANIMAL _ _
_ hit ANIMAL RWEAPON _ ⇒ the chain hit the snake
FIGHTER hit ANIMAL RWEAPON
FIGHTER hit FIGHTER RWEAPON _ ⇒ the ninpoka hit (the kendoka) with the sword
FIGHTER hit FIGHTER _ IRON ⇒ the ninpoka hit (the kendoka with the sword)
FIGHTER hit FIGHTER _ PENS

_
FIGHTER hit FIGHTER _ CLOTHES
_ hit FIGHTER RWEAPON _
_ hit FIGHTER RWEAPON IRON
_ hit FIGHTER RWEAPON PENS
_ hit FIGHTER RWEAPON CLOTHES
FIGHTER threw FIGHTER _ _

FIGHTER threw ANIMAL _ _
FIGHTER threw RWEAPON _ _
FIGHTER threw FIGHTER _ IRON
FIGHTER threw FIGHTER _ PENS
FIGHTER threw FIGHTER _ CLOTHES
FIGHTER wrote PAPER _ _
FIGHTER wrote PAPER PENS _
FIGHTER read PAPER _ _
FIGHTER wore CLOTHES _ _
FIGHTER wore RIRON _ _
FIGHTER pinched _ _ _
FIGHTER pinched FIGHTER _ _
FIGHTER pinched ANIMAL _ _
FIGHTER pinched FIGHTER _ IRON
FIGHTER pinched FIGHTER _ PENS
FIGHTER pinched FIGHTER _ CLOTHES
FIGHTER punched _ _ _
FIGHTER punched FIGHTER _ _
FIGHTER punched ANIMAL _ _
FIGHTER punched FIGHTER _ IRON
FIGHTER punched FIGHTER _ PENS
FIGHTER punched FIGHTER _ CLOTHES
FIGHTER punched _ FIST _
FIGHTER punched FIGHTER FIST _
FIGHTER punched ANIMAL FIST _
FIGHTER moved ANIMAL _ _
FIGHTER moved FIGHTER _ _
FIGHTER moved RWEAPON _ _
FIGHTER moved PENS _ _
ANIMAL moved ANIMAL _ _
ANIMAL moved IRON _ _
ANIMAL moved PENS _ _
ANIMAL moved _ _ _
FIGHTER moved _ _ _
CLOTHES moved _ _ _
WEAPON moved _ _ _
PENS moved _ _ _

B. Example of two trees



The tree above is visual, the distances tree below gives the exact distances between the clusters of words. The smaller the distance between two groups, the closer they are clustered together.

0.008457 (shoes) (kimono)
0.009938 (coal) (quill)
0.019384 (threw) (pinched)
0.019539 (samurai) (kempoka)
0.022790 (haiku) (text)
0.023465 (chain) (text haiku)
0.034815 (kempoka samurai) (kendoka)
0.034974 (quill coal) (kimono shoes)
0.035055 (pinched threw) (bit)
0.048895 (sword) (text haiku chain)
0.056110 (bit pinched threw) (punched)
0.060274 (snake) (tiger)
0.064029 (kendoka kempoka samurai) (ninpoka)
0.126242 (moved) (punched bit pinched threw)
0.519575 (teeth) (fist)
0.694482 (fell) (punched bit pinched threw moved)
0.736353 (read) (punched bit pinched threw moved fell)

0.761402 (punched bit pinched threw moved fell read) (ninpoka kendoka

kempoka samurai)
1.076960 (wore) (tiger snake)
1.110929 (ninpoka kendoka kempoka samurai punched bit pinched threw moved fell read) (tiger snake wore)
1.297745 (tiger snake wore ninpoka kendoka kempoka samurai punched bit pinched threw moved fell read) (kimono shoes quill coal)
1.298007 (kimono shoes quill coal tiger snake wore ninpoka kendoka kempoka samurai punched bit pinched threw moved fell read) (text haiku chain sword)
1.380019 (text haiku chain sword kimono shoes quill coal tiger snake wore ninpoka kendoka kempoka samurai punched bit pinched threw moved fell read) (fist teeth)
1.370809 (fist teeth text haiku chain sword kimono shoes quill coal tiger snake wore ninpoka kendoka kempoka samurai punched bit pinched threw moved fell read) (hit)
1.478090 (hit fist teeth text haiku chain sword kimono shoes quill coal tiger snake wore ninpoka kendoka kempoka samurai punched bit pinched threw moved fell read) (wrote)