MODELLING LATE BILINGUALISM WITH NEURAL NETWORKS

An investigation of the bilingual's phonological space

BA Thesis

As part of the BA Linguistics

and the Honours Program of the Faculty of Humanities,

at the University of Amsterdam

Maartje ter Hoeve

10190015

7th of July, 2014

Supervisor: Prof. Dr. Paul Boersma

Second reader: Mw. Dr. Silke Hamann

Universiteit van Amsterdam

CONTENTS

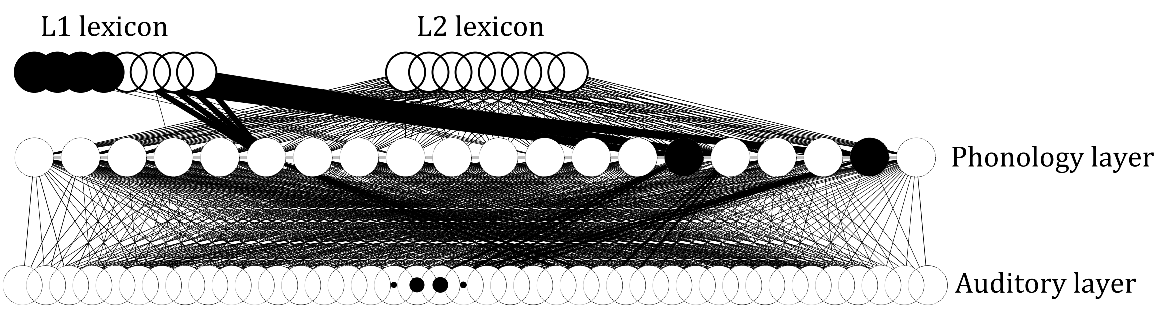
English abstract

A computational neural network can be seen as a simplified model of the human brain. It exists of several layers of nodes, which are connected just like the neurons in the brain are connected. For this study neural networks are used to answer the question how the phonological development of late bilingual speakers proceeds. Three scenarios are taken into account: (1) A speaker acquires a second language that contains more phonemes than her first language, (2) a speaker acquires a second language that contains fewer phonemes than her first language, and (3) a speaker acquires a second language that contains a contrast that is similar to a contrast in her first language, although the boundary between the two categories is different in the two languages (e.g. a difference in Voice Onset Time).

Considering the phonological system of a bilingual speaker two common theories are taken into account: (1) the bilingual speaker has two separate phonological systems, one for her first language and one for her second language, and (2) a bilingual speaker develops a bilingual phonological system, in which the phonologies of the two languages are merged.

In order to test the two theories for all the three scenarios three neural networks are used: one per scenario. These three networks consist of four layers: (1) an auditory layer, (2) a phonological layer, (3) a lexicon for the first language and (4) a lexicon for the second language (see figure).

In most cases the networks show the development of two separate phonological systems. This was predicted by the theory (1). However, the choice for separate lexicon layers and the amount of language input the networks are exposed to play an important role in the way the networks are able to separate the phonological systems and whether the networks forget their first language.



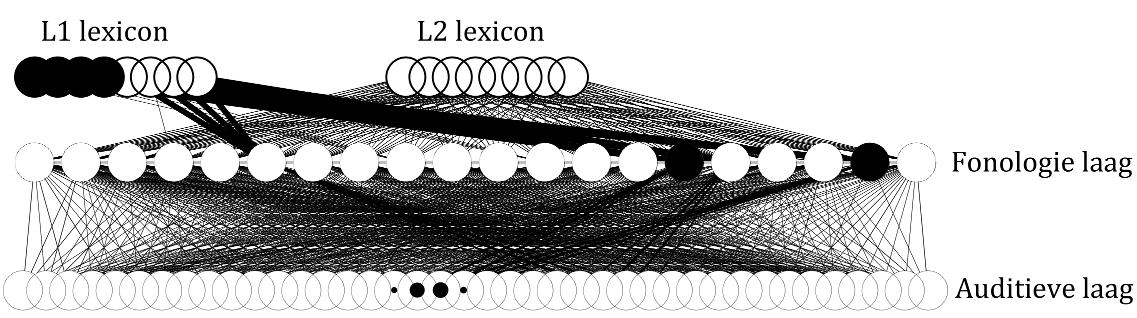
Dutch abstract

Een computationeel neuraal netwerk kan gezien worden als een vereenvoudigd model van het menselijk brein. Het bestaat uit verschillende lagen knopen die met elkaar in verbinding staan, net zoals zenuwcellen in het brein met elkaar verbonden zijn. In mijn scriptie heb ik neurale netwerken gebruikt om de vraag te beantwoorden hoe de fonologische ontwikkeling van laat-tweetalige sprekers zich ontwikkelt. Daarbij is gekeken naar drie scenario's: (1) een spreker leert een tweede taal die meer fonemen bevat dan haar eerste taal, (2) een spreker leert een tweede taal die minder fonemen bevat dan haar eerste taal, en (3) een spreker leert een tweede taal die eenzelfde soort contrast bevat als haar eerste taal, maar waarbij de grens tussen dit contrast verschilt in de twee talen (bv. een verschillende Voice Onset Time).

Voor het fonologische systeem van een tweetalige spreker zijn twee mogelijkheden: (1) de spreker heeft twee aparte fonologische systemen, één voor haar eerste taal en één voor haar tweede taal, of (2) een tweetalige spreker ontwikkelt een tweetalig fonologisch systeem, waarin de fonologie van beide talen wordt samengebracht en zelfs aangepast wordt op basis van de twee talen.

Om deze twee hypotheses voor de drie geschetste scenario's te testen is gebruik gemaakt van drie neurale netwerken; één voor elk scenario. Deze drie netwerken bestaan uit vier lagen: (1) een auditieve laag, (2) een fonologische laag, (3) een lexicon laag voor de eerste taal, en (4) een lexicon laag voor de tweede taal (zie afbeelding).

In de meeste gevallen ontwikkelen de netwerken twee gescheiden fonologische systemen, zoals voorspeld werd door hypothese (1). Echter, keuze voor twee gescheiden lexicons en de hoeveelheid taalinput die de netwerken krijgen zijn sturend voor de manier waarop de netwerken de fonologische systemen scheiden en of de netwerken hun eerste taal verleren.



PART I: INTRODUCTION

1. GENERAL INTRODUCTION

Over the course of time phonologists and psychologists have investigated how late second language learners learn phonemes that exist in their second language, but that do not exist in their first language, how late second language learners learn a differently located boundary between two phoneme categories, or how late second language learners learn a second language that contains fewer phonemes than their first language. In this thesis I use computational neural networks to gain more insight in this question. Please note: I use the terms artificial neural networks, neural networks, artificial neural nets and neural nets interchangeably.

This thesis consists of four parts. The first part includes, apart from this general introduction, definitions for the most commonly used terminology in this work. The second part of this thesis starts with a closer look on the phonology acquisition of a second language learner. After that more modalities that may affect the phonology of the second language learner are discussed. In all cases the second language learner is referred to as a female speaker. The second part concludes with a detailed explanation of computational neural networks. In the third part the neural networks that are used in this thesis are explained in more detail, after which the results that are obtained by the neural networks are presented and discussed. The fourth and final part of this work contains the conclusion of this study. Next to that ideas for further research are discussed. The scripts that are used to model the neural networks can be found as appendices at the end of this thesis.

2. TERMINOLOGY

Although many readers may be familiar with the terminology used in this thesis, in this section I will provide an overview of the most important terminology used in this study.

Early and late bilinguals

The distinction between early and late bilinguals is not very clear-cut. Studies have shown an age effect for learners of a second language (e.g. Lee, 2011; Lehtonen, Hultén, Rodríguez-Fornells et al., 2012; Piske, Flege, MacKay and Meador, 2002). However, the exact border between 'early' and 'late' is very difficult to define. Famous work by Lenneberg (1967) has introduced the so-called *critical period* for language learning. During this period the human brain is still able to learn a new language at a native-like level. After this period this capacity diminishes, which results in a foreign accent in the second language. Before Lenneberg's study was published, Penfield and Roberts (1959) already proposed a comparable sensitivity to language learning at a young age. However, Penfield and Roberts on the one hand and Lenneberg on the other hand disagree on when this critical period exactly ends and what the exact causes are. Penfield and Roberts state that children can learn a new language till approximately the age of nine, as after this age the brain has lost the plasticity it needs to learn a second language fluently, whereas Lenneberg argues that by the onset of puberty the process of lateralization in the brain has completed, which causes the end of the critical period. More authors have contributed to the discussion on the critical period hypothesis, which has resulted in several borderlines for the end of the critical period (e.g. Scovel, 1988, who argues that the critical period ends around the age of twelve, also due to a lack of brain plasticity during that age, or Patkowski, 1980, 1990, who discusses a critical period for speech and morphosyntax that ends at the age of fifteen). Generally it seems that the earlier someone has learned a second language the more native-like this person becomes in her second language, and that several modalities (like phonology and phonetics) may be more difficult to learn at a native-like level than other modalities (like syntax), but see exceptions like Julie (Ioup et al., 1994), who started to learn Egyptian Arabic at the age of 21. Native speakers of Egyptian Arabic were not able to recognize a foreign accent in her Egyptian Arabic. Because of exceptions like Julie some people argue against the notion of the critical period hypothesis. After all, one exception is already enough to reject a hypothesis, according to these authors, in the spirit of Popper (1959).

The research on the critical period hypothesis presented until now, but criticised by e.g. Birdsong & Molis, (2001), DeKeyser (2000), Birdsong, (2005) and Birdsong (2006), assumed a very clear decrease of the ability to learn a second language at the end of the critical period. However, other research has reasoned against this clear border and has proposed another view in which the ability to learn a second language continuously decreases over the course of time (e.g. Hyltenstam & Abrahamsson's, 2003).

For this thesis I decided to stay on the safe side and to teach the neural networks their first language very thoroughly, to make sure that late bilingual speakers were modelled. In this thesis, 'bilingualism' or 'bilingual speaker' always refers to late bilingualism, or late bilingual speakers, even if this is not explicitly mentioned.

First and second formant

According to Ladefoged (1996), the vocal tract can be seen as a tube. Air in this tube is set into motion by the vocal cords flipping apart and together at the beginning of the vocal tract. The vocal tract is not one straight tube, but contains some inequalities, holes and curves. The air in the vocal tract resonates in these inequalities, holes and curves. The resonances of the vocal tract are called 'formants'. The speech signal contains many formants, but for vowels the first two formants are the two most important ones to look at. The first formant is referred to as the F1 and the second formant is referred to as the F2. The formants of a vowel can be found in the spectrum and the spectrogram of the vowel. The spectrum of a vowel shows the resonances of the vocal tract and in a spectrogram the energy of the speech signal is plotted against the time.

Language modes

The notion of language modes has been introduced by François Grosjean (cf. Soares and Grosjean, 1984; Grosjean, 1988). Grosjean states that bilinguals can move along a language mode continuum. The ends of this continuum are occupied by the first and the second language and represent monolingual modes. The middle of the continuum is occupied by the bilingual language mode. See figure 2.1.

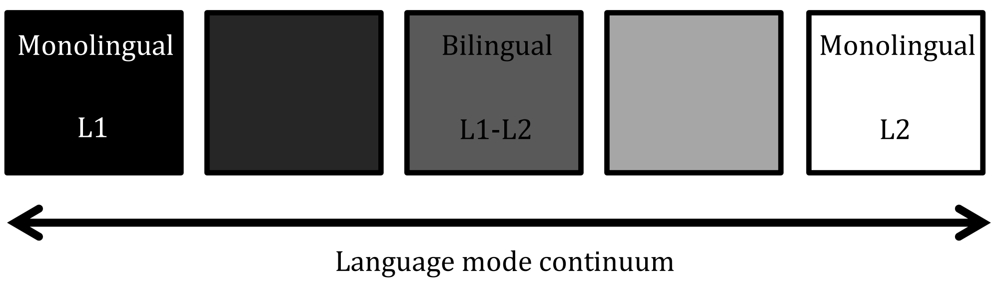


Figure 2.3 – Language mode continuum (based on Grosjean)

The language mode of a bilingual speaker changes depending on the situation. For example, if a bilingual speaker is surrounded by monolingual speakers of her first language, then she will only speak her L1, as speaking her L2 would cause misunderstandings. In this situation the bilingual speaker is in a complete monolingual L1 mode. However, if she is in surroundings where two languages are spoken, she will not arrive in one of the two ends of the continuum, but she will stay in her bilingual mode in the middle of the continuum.

In this study the term 'language mode' is used in a slightly different way than just described. As it turned out the neural networks that are used to simulate bilingual speakers were not able to switch between their two languages as easily as real bilingual speakers. The networks needed quite some language input in the target language. Due to this terms like 'L1 language mode' or 'L2 language mode' still refer to a state in which the networks perceive and produce sounds in either of their two languages, but the difference is that the networks are not able to switch as fast as real humans.

PART II: THEORY

3. THE SECOND LANGUAGE LEARNER: PHONOLOGY

Before the acquisition of sound categories can be modelled by a neural network, it is important to address some theoretical issues considering the acquisition of phonology. This chapter starts with briefly discussing the acquisition of phonological categories in the first language. After that three possible scenarios for the phonological categories in a second language are presented. Considering the phonological system of a late bilingual speaker two theories are taken into account: (1) a late bilingual speaker has two separate phonological systems, one for her first language and one for her second language, and (2) a late bilingual speaker develops a bilingual phonological system, in which the phonologies of the two languages are merged. Later in this study neural networks are used to try to provide evidence against one of the two theories. Ultimately the goal of this study is to choose one of the two theories as the most probable one. This chapter ends with a section in which *fossilization* is addressed. The question of that section is to what extent late bilingual speakers are still able to learn their second language on a native-like level.

3.1 The first language acquisition

When infants are born they are still sensitive to all phonemic contrasts (cf. van der Stelt and Koopmans-van Beinum, 2000). This means that they can distinguish between phonemic contrasts that exist in the language that is spoken in their surroundings, but also between phonemic contrasts that do not exist in the language that is spoken in their surroundings, but that do exist in other languages. This ability diminishes during the language development of the infant. Around their first birthday children have acquired a language-specific sound perception: children perform better on discriminating language-specific phoneme contrasts than before their first birthday and they are no longer able to distinguish between phonemic contrasts that are not part of the language spoken in their surroundings. Kuhl, Kritani et al. (1997) illustrate this development with an experiment on the perception of the [r] and [l] in Japanese. Japanese does not have the phonemic contrast between those two sounds, whereas American English (amongst others) does have this contrast. Kuhl, Kritani et al. showed that at 6-8 months of age American and Japanese children were both able to perceive the phonemic contrast between [r] and [l]. However, at 10-12 months of age American children had become better at the contrast between the two sounds whereas the performance of the Japanese children had decreased.

Distributional learning plays a major role in the acquisition of language-specific sound perception by infants. Figure 3.1a and 3.1b show for both American English and Japanese how the [r]-[l] continuum can be roughly displayed. On the y-axis is plotted how much the infants are exposed with a certain sound along the [r]-[l] continuum. The [r]-[l] continuum is plotted on the x-axis.

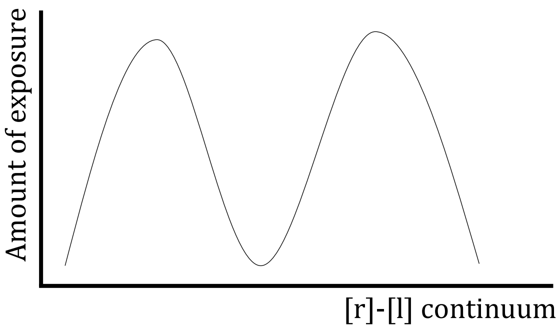


Figure 3.1a – Bimodal distribution for the [r]-[l] continuum in American English

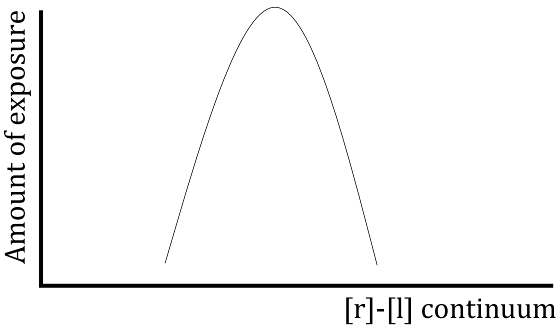


Figure 3.1b – Unimodal distribution for the [r]-[l] continuum in Japanese

Infants use this statistical distribution of speech sounds. Maye, Werker et al. (2002) showed that infants that are exposed to a new bimodal distribution are able to learn the distinction between the two ends of the continuum, whereas children that are only exposed to a unimodal distribution are not able to make this same distinction.

The language-specificity of sound perception shows that speakers of different languages perceive the same phonetic sounds differently on a phonological level. Speakers have to map the auditory signals they hear in their surroundings to the right phonological units in their language. Escudero (2005) argues that a listener follows a maximum-likelihood strategy for an optimal target perception, i.e. an optimal listener will perceive and categorise the different sounds she hears in her surroundings according to the distribution of phonetic contrasts in her language. Later, when speakers have developed a lexicon, combinations of these phonological units have to be mapped to the lexicon. After the mapping to the lexicon the right concept is assigned to the selected word in the lexicon, which gives the semantic meaning to the auditory signal (Escudero, 2005). After the child has learned the phonological categories of its language and after the child has learned some words of its language, the child's speech comprehension can be displayed as in the scheme in figure 3.2.

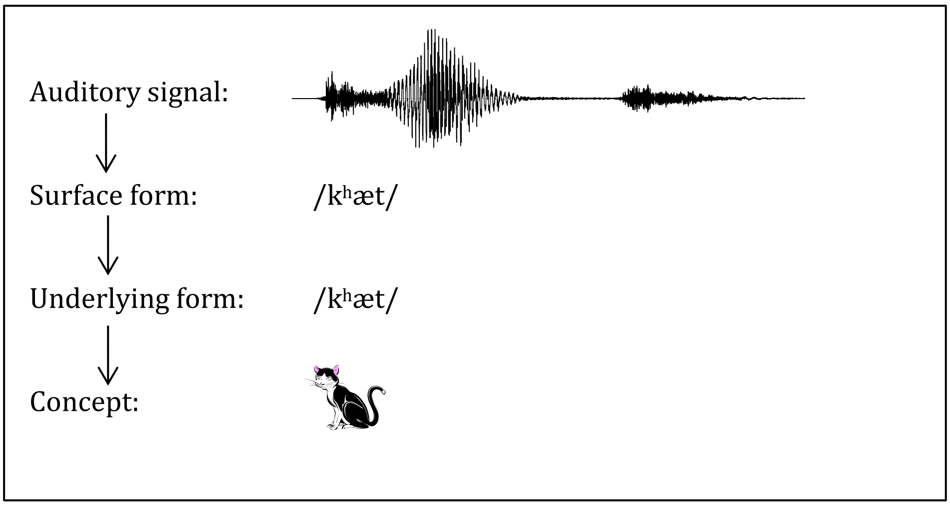


Figure 3.2 – Human speech comprehension

3.2 Three possible modifications to the phonology layer

When a monolingual speaker learns a second language, this may involve learning a new distribution of phonological categories, in order to develop an optimal L2 target perception and production. One can imagine three possible scenarios for this (cf. Escudero, 2005). Here those three scenarios are explained and a closer look is taken at what this means for the bilingual speaker. Examples are given of research that argues that the bilingual speaker has two separate phonological systems and of research that argues that the bilingual speaker has one, merged phonological system.

3.2.1 – The new *scenario*

In this scenario the second language learner needs to learn new sound categories that do not exist in her first language. This means that one category in the L1 needs to be divided into two or more categories in the L2. This situation is schematically displayed in figure 3.3. Figure 3.3a shows the abstract situation and in 3.3b the situation is clarified with an example, adapted from Escudero (2005). This example shows an L1 speaker of Spanish, who has to learn English as her L2.

$$\begin{tikzcd}
\textbf{L1} & \textbf{L2} \\
& \text{Category 1} \\
\text{Category 1} \arrow[ur] \arrow[dr] & \\
& \text{Category 2}
\end{tikzcd}$$

Figure 3.3a – Situation 1

$$\begin{tikzcd}
\textbf{L1 (Spanish)} & \textbf{L2 (English)} \\
& /\text{i}/ \\
/\text{i}/ \arrow[ur] \arrow[dr] & \\
& /\text{ɪ}/
\end{tikzcd}$$

Figure 3.3b – Example situation 1

Whenever a monolingual speaker starts to learn a second language, she will start in her L1 'position'. E.g. a speaker whose first language is Spanish, will not hear the difference between the English /i/ and /ɪ/. Several studies show that the ability to hear the difference between two categories in the second language that both belong to the same category in the first language increases with the increase of proficiency in the L2 (e.g. Boomershine, 2013; Flege, Bohn and Jang, 1997; Fox, Flege and Munro, 1995; Boersma and Escudero, 2002).

To the best of my knowledge, all research on the acquisition of new sounds in a second language shows that L2-learners are, at least to a certain extent, able to learn the new categories in their second language. In the remaining part of this section, the two theories about the bilingual phonological system(s) that are briefly presented in the beginning of this chapter (the bilingual speaker either has separate phonological systems or the bilingual speaker has one merged bilingual phonological system) are further discussed with respect to the new scenario.

Escudero (2005) presents the L2LP model for second language acquisition. This model predicts that the bilingual speaker ends with an optimal L1 perception and with an optimal L2 perception, if the bilingual speaker has received enough language input. In order to achieve this optimal end state the bilingual language learner starts with an optimal L1 perception, also for her second language. During the acquisition of the second language the second language learner ultimately acquires two separate phonological systems: one for her first language and one for her second language. However, note that Escudero (2005) has not been able to present evidence for this last prediction yet.

Evidence that bilingual speakers do not have separate phonological systems is presented as well. Here I will provide some research that is based on the Speech Learning Model (SLM), by Flege (1995). With the SLM Flege tries to cover all aspects of speech learning across someone's lifespan, which includes the acquisition of a second language. The SLM proposes that processes and mechanisms that children use to learn the sound system of their first language do not disappear over the course of time. However, other aspects prevent monolingual speakers from acquiring a second language later in their lives, for example the fact that most bilinguals continue to speak their first language (Flege, 2002). Moreover, the SLM states that the phonological categories of the bilingual’s L1 and of the bilingual’s L2 interact with each other through *assimilation* and *dissimilation* (Flege, 2003; Flege, Schirru and MacKay 2003). Assimilation is explained as the merging of an L1 sound category and an L2 sound category. Assimilation is expected to happen when these two sound categories do not differ enough to receive different categories in the bilingual speaker’s phonological system. Dissimilation, on the other hand, is expected to take place when a new L2 sound category differs enough from the already existing L1 sound categories to receive its own phonemic category in the phonological space. However, dissimilation also means that the distance between the new category and the old categories is increased as much as possible. Due to this increase in distance the new and the old sound categories move further apart from each other. In the end all the sound categories that have moved apart are not native-like anymore, but adapted to the bilingual situation.

Flege, Schirru and MacKay (2003) argue that assimilation and dissimilation have been found in a research on four groups of Italian-English participants. The participants in the first group had learned English early in their lives and still had a high L1 use. The participants in the second group had learned English early in their lives as well, but did not use their L1 as often anymore. The participants in the third group had learned English later in their lives and had a high L1 use, whereas the participants in the fourth had learned English later in their lives but had a low L1 use. The participants of the four groups were all asked to pronounce the English /e$^{\text{ɪ}}$/. Italian lacks this phonemic category. Instead it contains the /e/, which is produced with less tongue movement than the English /e$^{\text{ɪ}}$/. The results were compared to the results of native English speakers. The participants in the third and in the fourth group (the late bilingual speakers) were found to pronounce the English /e$^{\text{ɪ}}$/ with less tongue movement than native English speakers. Flege, Schirru and MacKay take this result as evidence for the assumption that the speakers merged the phonetic properties of the English /e$^{\text{ɪ}}$/ and the Italian /e/: a case of assimilation. On the other hand, the participants in the second group (the early bilinguals with a low L1 use) tended to pronounce the English /e$^{\text{ɪ}}$/ with more tongue movement than native English speakers. This was taken as evidence for the assumption that these speakers had acquired a new category for the English /e$^{\text{ɪ}}$/ from the Italian /e/, but in such a way that the two categories moved as far apart from each other as possible: a case of dissimilation.

3.2.1 – The similar *scenario*

In this scenario a sound contrast exists in both the L1 and the L2. However, the boundary between these two sounds is differently placed in the L1 and the L2, see figure 3.4a. An example of this, adapted from Escudero (2005), is shown in figure 3.4b. Figure 3.4 shows the difference between Canadian French (CF) and Canadian English (CE), considering /ɛ/ and /æ/. Speakers of either language that learn the other language, will have to learn that the boundary between the two categories is different in the two languages.

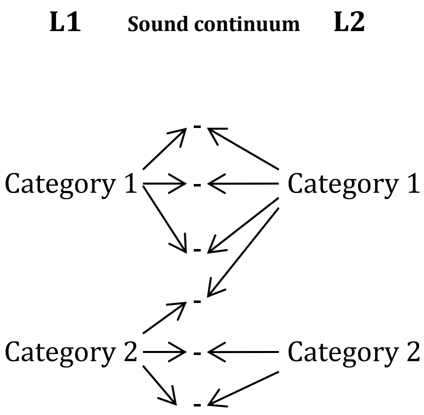


Figure 3.4a – Situation 2

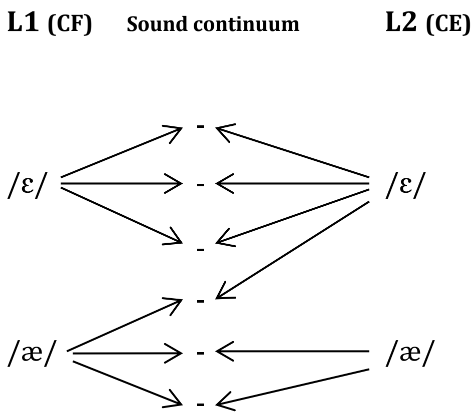


Figure 3.4b – Example situation 2

Also for this scenario one can find research that presents evidence for the assumption that bilingual speakers shift between two phonological systems, depending on the language environment and research that is supporting the SLM (Flege, 2005). Evidence for two separate phonological systems is presented by Garcia-Sierra, Diehl and Camplin (2009). This research investigated the categorization of the /ga/-/ka/ continuum by Spanish-English bilinguals and monolingual English speakers. Garcia-Sierra, Diehl and Camplin found a phonemic boundary shift, depending on the language environment, both for bilingual speakers and monolingual speakers. However, bilingual speakers were found to make a larger shift than monolingual speakers. I.e. these speakers classified certain sounds differently, depending on the language environment.

On the other hand, after the development of the SLM (Flege, 1995), Flege (1987) was adopted as evidence for the notion of assimilation. In this research Flege (1987) compared the production of /t/ and /d/ by English monolinguals and French monolinguals with the production of /t/ and /d/ by late English-French and late French-English bilinguals. All the participants in the four groups were asked to read English and French phrases like "Two little X" and "Tous les X". The Voice Onset Times (VOTs) of the /t/ pronounced by French monolinguals were found to be longer than the VOTs of the /t/ pronounced by English monolinguals, whereas the VOTs of the /t/ pronounced by bilingual speakers were found to lay in between the monolingual VOT values.

One needs to realise that the first research presented here focusses on the perception by bilingual speakers whereas the second research focusses on the production by bilingual speakers.

3.2.1 – The subset *scenario*

In this scenario two L1 categories merge into one category in the L2. The set of sound categories in the L2 forms a subset of the set of sound categories in the L1. This is the case for L1 speakers of English that have to learn that in Dutch the distinction between /æ/ and /ɛ/ is not made. Figure 3.5a and figure 3.5b show this situation, again adapted from Escudero (2005).

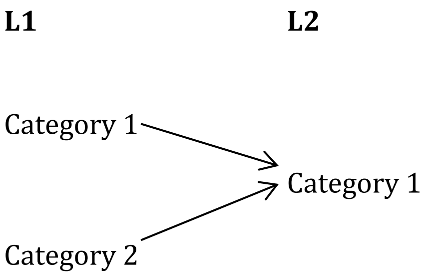


Figure 3.5a – Situation 3

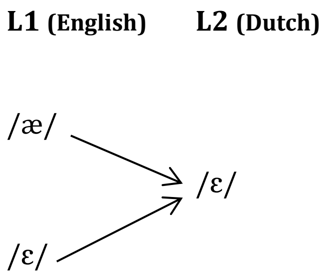


Figure 3.5b – Example situation 3

Also in this scenario the L2-learner starts in the initial state of an optimal L1 speaker and listener. This means that the L2-learner distinguishes more phonemic categories than the optimal L2 speaker and listener. There is a not a lot of research on this scenario. Escudero (2005) argues that this may be because at first sight scenario 3 does not result in a lexical problem for the L2 learner. However, Escudero and Boersma (2002) do present evidence for what they call *Multiple-category assimilation (MCA)*. In their research is focussed on Dutch learners of Spanish. The Dutch vowel inventory contains twelve monophthongs, namely (i y u ɪ ʏ ɛ ɔ ɑ aː eː øː oː), whereas the Spanish vowel inventory only consists of five monophthongs: (i e a o u). A Dutch L2-learner of Spanish will have to learn which vowels of the Dutch vowel inventory do exist in Spanish and which vowels do not. Escudero and Boersma describe the subset problem in acquisition as "the general problem of how the learner can learn on the basis of positive evidence alone that some feature does not exist in the target language". Escudero and

Boersma have designed three tasks to test the nature of the problematicity of MCA. The three tests are briefly discussed below.

The first test aimed to prove that MCA actually exists. In order to test this participants listened to the Spanish vowels /i/ and /e/, embedded in Dutch carrier phrases. Participants thought they were listening to Dutch vowels and they could classify them as /i/, /ɪ/ and /ɛ/. The participants were divided into three groups, based on their level of expertise in Spanish: (1) Dutch only, (2) Beginners, (3) Intermediate, (4) Advanced and (5) Bilingual. All participants used all three Dutch categories to classify the Spanish vowels, which is taken as evidence for the assumption that multiple-category assimilation exists.

In the second test the same vowel stimuli were used, but embedded in Spanish carrier phrases. This time the participants were told that they were listening to Spanish. However, they were asked to listen with their "Dutch ears". This means that the participants had to classify the Spanish vowels as Dutch vowels. The results showed that the participants did classify the sounds differently than in the first task, namely in a more Spanish way. Escudero and Boersma (2002) suggest that this shows that people switch between language modes and are not able to listen with "the ears of the other language mode".

In the third test participants were asked to listen to the same stimuli as in the second task, but now they had to classify the sounds as one of the five Spanish vowels. It was expected that Dutch learners of Spanish would make more mistakes with the front vowels than with the back vowels, due to the MCA. This was exactly what was found. The more proficient the Dutch learners were, the fewer errors they made with the front vowels.

Escudero and Boersma (2002) conclude that MCA exists and that it is problematic for the categorization of L2 vowels. Proficiency plays a major role in acquiring the optimal L2 perception and Escudero and Boersma also state that the learners make use of language modes, between which they can shift without changing their L1 perception.

3.3 Real life: fossilization and an end state

Flege's Speech Learning Model (Flege, 2005) accounts for an accent in a foreign language, as it states that new categories in a second language will never be the same as these categories in a first language, either due to assimilation or to dissimilation. The SLM also predicts an accent in the first language, since assimilation of categories has taken place (Flege, 1987b). On the other hand, the work on two phonological systems is less explicit as to foreign accent, mainly because this works focusses on speech perception rather than on speech production.

Apart from the question which hypothesis is the right one, and besides the fact that research has shown that late bilinguals have more difficulties with learning a second language than early bilinguals (see chapter 2), late bilinguals also often lack the ability to learn a foreign language at a native-like level at all, despite their continuous effort. Selinker (1972) introduced the term *fossilization* for this process. At some point L2-learners arrive in a so called *end state* of their foreign language learning, in which they cannot improve their level of speaking and their level of understanding their second language anymore. However, Birdsong (2009) argues that this end state is not to be thought of in an absolute way. The L2 of a bilingual speaker will still be subject to change over time, with respect to, for example, vocabulary learning, but also with respect to the influence the L1 has on the L2.

Various causes have been proposed for the observed fossilization. A few of these causes are already mentioned before: a decrease of brain plasticity, which was first suggested by Penfield and Roberts (1959) and the completion of the lateralization of the brain, which was proposed by Lenneberg (1967). Apart from these more neurological causes, also more social and psychological causes have been proposed, as it seems that the age of onset of the second language is confounded with many other factors. Flege (2002) provides an overview of research that has investigated the fossilization and the end stage of L2-learning. Many of the proposed factors can be related to the kind of language input. First of all both the quantity and the quality of the L2 input seem to influence the end state in which the L2-learner arrives. Flege describes a study by Stevens (1999) that showed that late bilinguals often receive L2 input that is of less quality than the input the early bilinguals receive. The early, young, bilinguals may be exposed to the L2 spoken by native speakers at school, whereas the late bilinguals may interact in a multicultural environment where they are exposed to several foreign accents in the L2. Other studies described by Flege show that the more the L2-learners are exposed to their second language, the better they learn the second language, despite age differences.

4. NEURAL NETWORKS

In this thesis I use artificial neural networks to gain a better insight in the question whether late bilingual speakers develop two separate phonological systems (one for their first language and one for their second language) or whether their monolingual phonological system is expanded by the acquisition of new L2 phonemes. In this chapter artificial neural networks are explained. In order to understand the networks that I use to answer my research question it is important to understand the biological underpinnings of an artificial neural network. This is briefly explained in section 4.1. Section 4.2 deals with artificial neural networks in general and the comparison is made between a biological neural network and an artificial neural network. Section 4.3 finally focusses on the specific neural networks that are used for this study.

4.1 Neural networks from a biological point of view

Artificial neural networks are based on the animal nervous system. This system consists of the central nervous system and the peripheral nervous system. The central nervous system contains the brain and the spinal cord, whereas the peripheral nervous systems contains the rest of the nerves that are spread through the body. In the nerves several neurons are grouped together. The neurons are the building blocks of the nervous system: the neurons transport information through the entire nervous system. A neuron consists of four parts: the cell body, an axon, a synapse and dendrites. The cell body (the soma) also contains the cell nucleus and controls the cellular function. The axon transports the information that is produced by the cell body to the synapse. In the synapse the information is transferred to another neuron. The neuron receives the information via its dendrites. The sending neuron is called the presynaptic neuron, whereas the receiving neuron is called the postsynaptic neuron (see figure 4.1)

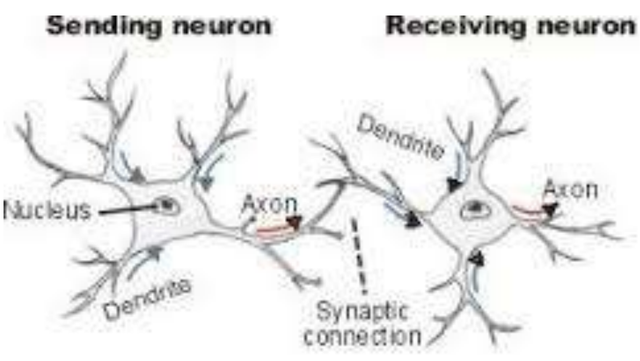


Figure 4.1- A typical structure of a sending and a receiving neuron (Watson, n.d.)

From the cell body to the synapse, information is spread via electric pulses of around 0.07 Volt. If the frequency of the pulses (or the *firing rate*) is high enough, a so-called *neurotransmitter* is released at the synapse. (Note: the voltage of the pulses cannot increase or decrease, only the frequency of the pulses can.) The neurotransmitter is released in the synaptic cleft; the gap between the presynaptic and the postsynaptic neuron. The neurotransmitter is then bound to receptors at the postsynaptic neuron. This causes a change in membrane potential of the postsynaptic neuron. (This is done by transferring Na^+ and K^+ through the cell membrane, via active and passive transport, but the detailed procedure is not relevant for the current study.) The potential of the membrane can change in two ways: excitatory and inhibitory. An excitatory activation causes the activation of the postsynaptic neuron. If the activation is beyond a certain threshold, the action potential is transferred via the postsynaptic neuron as well. On the other hand, an inhibitory activation lowers the chances that the threshold for a new action potential on the postsynaptic neuron is reached. This lowers the chances that the postsynaptic neuron is going to fire (cf. Byrne, (2014) and Seinhorst, 2012).

4.2 Neural networks from a computational point of view

Although highly abstracted, artificial neural networks are based on the biological structure described in the previous section. Exactly this biological validity makes that one can argue that the use of artificial neural networks should be preferred over other models that are often used in Linguistics. In this section the general outlines of an artificial neural network are discussed and important properties of artificial neural networks are mentioned. At the end the reader should have a good basis for understanding the properties of the artificial neural networks that are used for this specific study. In this thesis solely neural networks that can be modelled by the computer program PRAAT (Boersma and Weenink, 2013, version 5.3.65) are discussed, for all the networks that are used in this study are modelled and run in this program.

Figure 4.1 schematically shows an artificial neural network. This network is just a small toy network, but it shows all the important properties of the neural networks that are used in this thesis.

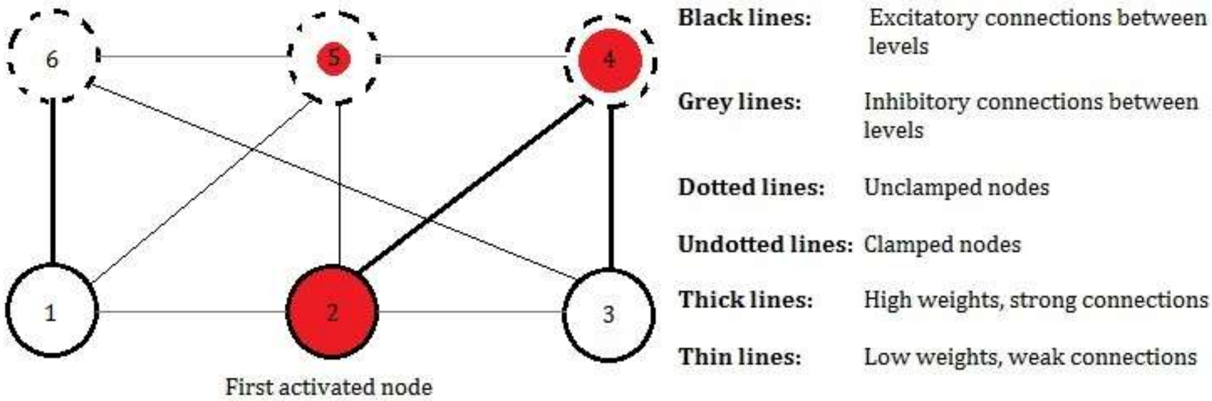


Figure 4.2 – An artificial neural network

As can be observed in figure 4.2, an artificial neural network consists of several nodes. The nodes of a neural network can be compared to the cell bodies in real, biological neurons. The nodes of a neural net are connected via excitatory or inhibitory connections. The excitatory connections can increase the activity of the interconnected nodes, whereas the inhibitory connections can decrease the activity of the interconnected nodes. The amount of activation of a node depends on (1) the amount of activation of the interconnected nodes and (2) the weights of the connections that lead to the node that becomes activated. The weight of the connections can be seen as the strength of the connections. Figure 4.2 also distinguishes between clamped and unclamped nodes. Clamped nodes are the nodes that are kept fixed, whereas the unclamped nodes are the nodes that are not fixed. E.g. the second node of the network has been activated completely (which is a fixed state) and it spreads its activity via the connections to the fourth and the fifth node. The fourth and the fifth node become activated, but the amount of activation depends on certain aspects (such as whether the second node is activated or not, the connection weights between the second and the fourth and the fifth node, the inhibition of the connections to the fifth and the second node, etc.). Because the activation depends on those aspects and because those aspects are variable, the activation of these nodes is not fixed, so, the nodes are unclamped.

In the remaining part of this section more can be read on the activity and the excitation of the nodes (section 4.2.1) and the weights of the connections (section 4.2.2). In these two subsections also the comparison with the biological nervous system is made. Boersma, Benders and Seinhorst (fc.) is used as the main source for this explanation.

4.2.1 Activity and excitation of the nodes

In figure 4.2 the second node was completely activated. To gain a better insight in what this means and what effect this has on the other nodes in the network, imagine the node had an activation of 1.00 (for now the unit of the activation is not important; just assume the activation reaches from 0.00 to 1.00). Activity in the network can be compared to the firing rate of a neuron in the biological nervous system. This means that node two in figure 4.2 is maximally firing. Node two is connected to node four and node five. The activity of 1.00 has to be divided between the connection to the fourth node and the connection to the fifth node. In figure 4.2 the connection between node two and node four is stronger than the connection between node two and node five. One can imagine that the strongest connection has a weight of 0.70 and the weakest connection a weight of 0.30 (these numbers are directly adapted from Boersma, Benders and Seinhorst, fc.). How the weights strengths are exactly determined is explained in the next section. This division causes that 70% of the activation of the second node is transferred to the fourth node and 30% of the activation of the second node is transferred to the fifth node.

In a neural network all the inputs a node receives can be added up to calculate the total excitation of this node. So, for now not taking into account the inhibitory connections that actually play a role as well, the excitation of the fourth node in the network becomes 0 + 0 + 0.70 = 0.70. However, if the third node had been activated as well and if this node had sent an activity of 0.20, the total excitation would have been 0.90. One could smooth the excitations to make sure the excitation of a node cannot exceed 1.00. The excitation of a node can be compared to the change in the membrane potential, which is described in section 4.1.

A node in a neural network that is excited with an excitation higher than 0.00 becomes activated. This node can now spread its activation to other nodes in the network that are connected with this particular node. This spreading of activation can be compared to the firing of the neuron. A neuron will only fire if the firing frequency of the action potential is high enough. The excitation of nodes and the activity spreading continues till all the nodes that could be activated are activated. A connection between two nodes can either be unidirectional or bidirectional. A connection that is unidirectional can only spread the activity in one direction, whereas a bidirectional connection can spread the activity in two directions. The nodes of the neural networks that are used for this study are connected via bidirectional connections.

4.2.2 The weight of the connections

The strengths of the connections between the nodes change, based on whether and how often two nodes are activated together at the same time. It would be appealing to assume that the connection weights change according to Hebb's famous rule: "Cells that fire together, wire together" (Hebb, 1949). However, this would also mean that the connection weights can increase unlimitedly. Boersma, Benders and Seinhorst (fc.) propose several solutions to avoid this unlimited increase. The best solution, which is also the method that is used for the neural nets in this thesis, is called *inoutstar learning*. This means that the weights of the connections depend on the probability that the input node is on, given that an output node is on (which is *instar learning*) and the probability that the output node is on, given that the input node is on (which is *outstar learning*). Because the weight of the connections depends on both these two probabilities the bidirectional character of the network is taken into account. Next to that the inoutstar learning algorithm still causes that the strength of a connection between two activated nodes increases. If only one of the two nodes is activated, the strength of the connection between these nodes decreases and if none of the nodes are activated, the strength of the connection between these nodes does not change. This means that the weight leak is set to zero. (If the weight leak is not set to zero, the connection weights decrease even if the interconnected nodes are not activated, see Boersma, Benders and Seinhorst.) The weights of the connections are updated after all the activity has been spread through the network.

4.3. The BiPhon model

The neural networks that are used in this study are based on earlier constructed neural nets used in Seinhorst (2012), Benders (2013), Chládková (2014) and Boersma, Benders and Seinhorst (fc.). These networks are inspired by Boersma's BiPhon model (2007a). For this reason the BiPhon model is discussed below.

The BiPhon model models phonological and phonetic knowledge of language users. Figure 4.3 on the next page shows the layers of the model and the connections between these layers, as presented in Seinhorst (2012).

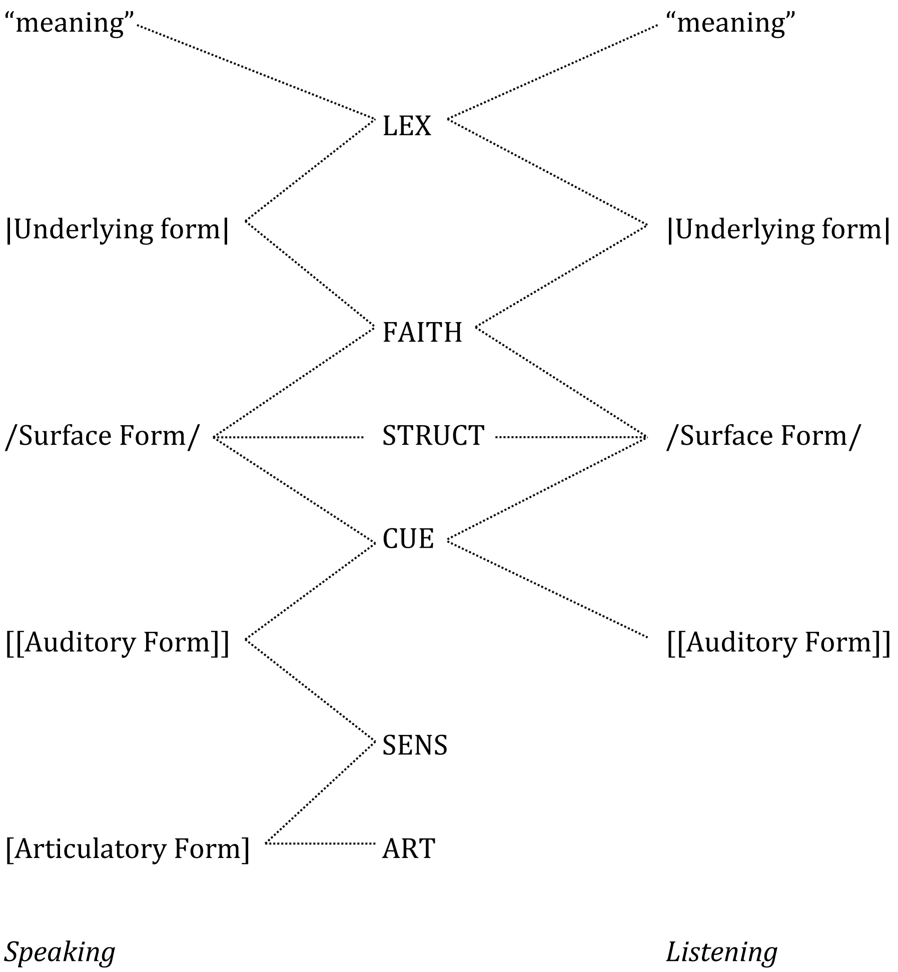


Figure 4.3 – The BiPhon Model

In the model semantic representation is modelled on the 'meaning' layer. The Underlying Form (UF) and the Surface Form (SF) represent the phonological structures of a word. The SF is the layer for the discrete phonological structure of a word, whereas the UF refers to a word stored in the mental lexicon (Escudero, 2005). The Auditory Form (AudF) is the phonetic layer of the network. It represents phonetic aspects such as duration, pitch, formants, etc. As can be seen in figure 4.3, the Articulatory Form (ArtF) is only connected in the speaking mode of the model. This is because the Articulatory Form represents the motoric plan for realising a certain sound.

5. MODALITIES THAT COULD AFFECT THE PHONOLOGY LAYER

The research question of this study is whether late bilingual speakers acquire separate phonological systems, or whether they develop one merged bilingual phonological system. This question is addressed by using neural networks. The SF layer of the neural network is of main interest to answer this question. Boersma, Benders and Seinhorst (fc.) already use neural networks to model a language learner that has learned three categories in her native language. Then she moves to a different area, where a four category language is spoken. Importantly, Boersma, Benders and Seinhorst use a network with only two layers to model this situation, namely an AudF and an SF (adapted from the BiPhon model, see the previous chapter). Boersma, Benders and Seinhorst write that the network is easily able to adapt to the new situation and that the network is able to learn the new language with four categories. The opposite situation (a learner's first language contains four categories, whereas her second language contains only three categories) gave comparable results. Boersma, Benders and Seinhorst conclude that "the network has a high degree of plasticity, adapting itself to changes in the environment as well as to changes in its own structure". However, the network presented by Boersma, Benders and Seinhorst is a very simplistic model. To simulate a situation that comes nearer to reality, many adaptations to the current network are required. In this chapter these adaptations are discussed one by one.

5.1 The auditory form

The network as presented in Boersma, Benders and Seinhorst (fc.), changes from an optimal distribution of three categories at the AudF to an optimal distribution of four categories at the AudF (see figure 5.1a and figure 5.1b). The horizontal axes in figure 5.1 represent the AudF and the vertical axes represent how often a certain sound occurs on the AudF. The peaks are formed according to Gaussian distributions.

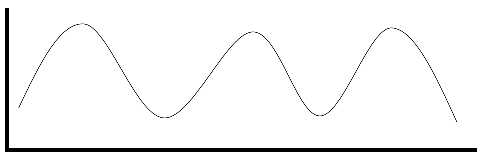


Figure 5.1a – A distribution with three peaks

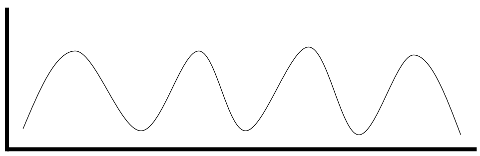


Figure 5.1b – A distribution with four peaks

However, it is not very plausible that the phonemic categories of the learner's L1 all take exactly one third of the AudF, whereas the phonemic categories of the learner's L2 all take exactly one fourth of the AudF. In order model a more realistic situation, the three scenarios that are explained in chapter three of this work (new, similar and subset) should be modelled. For the new scenario peaks should be added to the already known categories, whereas for the subset scenario peaks should be deleted from the language input. For the similar scenario the means of the categories need to change. In chapter six these adaptations are discussed in more detail.

Moreover, in the network in Boersma, Benders and Seinhorst (fc.), only the nodes at one particular part of the AudF are activated for a certain sound. If one wants to model the bilingual acquisition of vowel categories, it would be better to train the model on the F1 and the F2 of these vowels. This can be achieved by activating the AudF at two points for one vowel. The first couple of nodes are activated for the F1 and the second couple of nodes are activated for the F2 (cf. Chládková, 2014). This approach is discussed and explained in chapter 6.

By combining these two adaptations one is able to model a bilingual speaker whose first and second language have different vowel inventories, e.g. a bilingual's first language vowel inventory exists of /a/, /i/ and /u/, whereas this bilingual's second language vowel inventory exists of /a/, /i/, /u/, /e/ and /o/.

5.2 A more realistic language environment

In the current network the second language learner is taught her first language, after which the language environment changes and she is exposed to her second language. After this, the first language is not presented anymore. However, in a more realistic situation the bilingual speaker is also exposed to her first language again, even though she has acquired (the basics of) a new second language now. For this reason it is important to change the language environment of the network every now and then, so that the model is exposed to the first language as well as to the second language.

5.3 The lexicon

As mentioned before, the network in Boersma, Benders and Seinhorst (fc.) contains only two layers: an Auditory form and a Surface Form. However, it could well be possible that another layer, e.g. a lexicon layer, affects the phonology layer as well. The question is how this lexicon layer should be modelled in a neural network.

The bilingual lexicon is a topic that is often addressed within psycholinguistics. Three possibilities for the bilingual lexicon are taken into account: (1) bilingual speakers have two lexicons, one for their first language and one for their second language (cf. Kolers, 1963), (2) bilingual speakers have one single lexicon, which is simply larger than the lexicon of a monolingual speaker (cf. Kirsner, Lalor and Hird, 1993), and (3) bilingual speakers start with one lexicon, but as the words of their first language are always used together and the words of their second language are always used together, the connections between the words of one language strengthen and the connections between words of different languages weaken. In the end this creates two separate lexicons (cf. Paradis, 2004). However, it is not always clear whether early or late bilingual speakers are examined or how terms such as ‘separate lexicons’ or ‘integrated lexicons’ are defined.

Priming experiments are often used to investigate the question whether a bilingual speaker has two separate lexicons or not (cf. Larsen, Fritsch and Grava, 1994; Jiang and Forster, 2001; Duñabeitia et al., 2010). In this kind of research bilingual participants are usually presented with words in either of their two languages (or non-words that could have been words). These words are preceded by primes, which consist of either words in the same language or words in the other language. Participants are asked whether the presented word (not the prime) is a genuine word, or not. If the lexicons are separated, primes in the other language will not affect the decision time of the participants, these studies argue. However, primes in the same language will affect the reaction times. Different results are found and many factors (such as proficiency and language dominance) seem to play a role.

Despite these divergent results, often it is assumed that bilingual speakers have two lexicons that can be activated at the same time. E.g. Dijkstra et al. (1999) and Haigh and Jared (2007) showed that bilingual speakers activate both their lexicons when presented with homophones (words that sound the same in the two languages). In these two studies bilingual speakers performed a lexical decision task. In a lexical decision task participants have to decide whether a presented word is a genuine word or not. In Dijkstra et al. and Haigh and Jared a bilingual lexical decision task was used in which participants had to judge whether the presented word was a genuine word in either of their two languages. E.g. Dutch English bilingual speakers could have to identify whether a word that is presented on the screen is a Dutch word or not. If so, the participants should press ‘yes’, if not (the word was a non-word, or a word in another language, including English) the participants should press ‘no’. Homophones were expected to change the reaction times of the participants. This was found in both studies. For this reason in both studies the conclusion is drawn that bilingual speakers activate both their lexicons while listening to speech. Only at a later moment in time the right word, in the right lexicon, is selected.

Even if bilingual speakers have two lexicons that can be activated simultaneously, the questions whether or not these lexicons are completely separated or whether the words in the different lexicons are still connected is still to be answered. The possible connections between words in the lexicon could be directly between two words, but the words could also be connected via another layer. In by now classic research Potter, So, Von Eckardt and Feldman (1984) state that the two lexicons are connected via an additional concept layer. Over the course of time many researches have argued against this *concept mediation* theory, whereas the current research is rather in favour of this theory (see De Groot (2011) for a detailed overview).

For this study I have chosen to model two separate lexicons that are connected via a phonological layer (see chapter 6 for a detailed explanation). A model like this can account for the double lexical activation in the case of homophones. In the neural networks that are used for this research the nodes of a lexicon layer are connected via inhibitory connections.

Two other options would account for the double activation for homophones as well. One could model one lexicon layer, on which homophones are represented only once. In this case the lexicon layer should be connected to a concept layer, which has separate representations for homophones. The nodes on the lexicon layer that represent the homophones should be connected to different concepts. However, a concept layer negatively influenced the results of the networks, so I have chosen not to include a such a layer (see chapter 6). A second option would be a model with only one lexicon layer, on which all the meanings of a homophone are represented. A concept layer would not be necessary. The only difference between this model and the model I have chosen to use (with complete separate lexicons) is that in the model that is used for this thesis no connections between the two different lexicons are modelled, whereas for the second option described here, inhibitory connections are added between all the nodes of the lexicon.

5.4 Plasticity

In 1890 William James has introduced the term plasticity for the ability of the brain to change (Cotman en Berchtold, 2002). Since then it has been widely held and assumed that over the course of time the brain has to contend with a decreasing plasticity. Due to this decrease the ability of the brain to make new connections and, due to that, the ability to learn, declines. In chapter two of this study it is already mentioned that the a decrease of brain plasticity could be one of the causes for the difficulties late second language learners experience during their second language acquisition. For this study a decreasing brain plasticity is not directly taken into account, but see chapter 9, in which ideas for further research are discussed.

5.5 Sensorimotor constraints

Bilinguals may also face a sensorimotor problem. This would mean that the bilingual speaker has already acquired the fine articulatory movements that belong to her first language. After having acquired these small movements, it could be difficult to learn the new fine muscle movements for the second language. This in turn may cause articulatory problems. This would mean that the late bilingual language learner does not necessarily have another phonological representation than a monolingual speaker, but that she has difficulties pronouncing the new sounds. Sensorimotor constraints are not directly considered in this study either, but again, see chapter 9 for ideas for further research.

PART III: APPLICATION

6. THE DETAILS OF THE NETWORKS

For this study three different neural networks are modelled: (1) a network for the new scenario, (2) a network for the similar scenario, and (3) a network for the subset scenario. In this chapter the details and characteristics that are applicable to all the networks are discussed and explained. After that a closer look is taken at the single networks, one by one. Simple toy languages are used to simulate the different scenarios.

6.1 The general outlines of the networks

The BiPhon model (see chapter 4) has been the inspiration for the layout of the current, and previous, neural networks (e.g. Seinhorst (2012), Benders (2013), Chládkova (2014) and Boersma, Benders and Seinhorst (fc)). The neural networks that are used for this study consist of four layers: (1) the Auditory Form, (2) the Surface Form, (3) the Underlying Form for the first language, and (4) the Underlying Form for the second language (see figure 6.1). As stated in chapter 4, the Underlying Form represents the mental lexicon. In the remaining part of this thesis the word 'lexicon' is used to refer to the Underlying Form.

Excitatory connections are present between the nodes at the Auditory Form and the nodes at the Surface Form, between the nodes at the Surface Form and the L1 lexicon nodes and between the nodes at the Surface Form and the L2 lexicon nodes. Nodes on the same layer are connected via inhibitory connections. Note that these inhibitory connections are present between all the nodes on one layer, so not only between the neighbouring nodes. Excitatory connections are needed to activate the nodes on the next layers, whereas inhibitory connections are needed to make sure that not all the nodes on one layer are activated, but only a limited amount. The values for the inhibition are determined on the basis of the ideal L1 learning situation. This means that the model had to be able to learn the L1 without any mistakes, just like a real child that has to learn its first language.

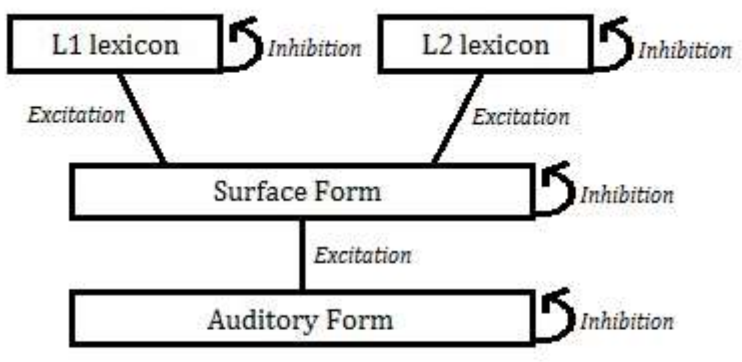


Figure 6.1 – Schematic overview of the used neural networks

The learning phase of the network contains several learning steps (see section 6.2, 6.3 and 6.4 for a detailed explanation of the learning phase per scenario). During one learning step the network is presented with a sound-meaning pair, i.e. sounds on the AudF belong to words on the lexicon layer and during a learning step certain AudF nodes and the corresponding lexicon nodes are activated. The activation is spread through the network and the weights of the connections are updated, following the learning algorithm (*inoutstar* learning, see chapter 4). Due to this learning algorithm patterns emerge on the Surface Form for every sound-meaning pair. The spreading rate of the network is set at 0.01.

The mapping of a single sound to a single word shows that the used networks are only simplifications of reality. However, the current state of the development of the neural networks that are used in phonology does not allow the combination of sequences of sounds in order to form a word.

In the remaining part of this section the different layers and their characteristics during the learning phase are discussed in more detail. This section ends with a brief explanation on the production and the perception of the network, i.e. the network while it speaks and the network while it listens.

6.1.1 The Auditory Form (the AudF)

The Auditory Form consists of forty nodes, following Chládková (2014), which are clamped during the learning phase. The AudF represents a certain auditory continuum. Depending on the scenario that is modelled (new, similar or subset) this auditory continuum is either the frequency spectrum along the basilar membrane in the ear (the spectrum increases from left to right and this makes it possible to model vowels in the network) or the Voice Onset Time. The toy languages that are used to model the three scenarios contain several categories, which are divided over the auditory continuum. Every category is defined according to a Gaussian distribution. For every learning step one or two nodes (depending on whether or not vowels with two formants are modelled) on the AudF are randomly selected and activated, but following the Gaussian distributions for the categories. This means that nodes on the AudF in the middle of the Gaussian distribution are selected more often than nodes on the AudF that lay at the ends of the Gaussian distributions. Due to this distributional learning can take place (see chapter 3). The AudF nodes in the direct surroundings of the selected AudF nodes are also activated according to (another) Gaussian distribution. Then the activation is spread to the SF layer (see next section) via the excitatory connections. Note that the network, just like a real human, does not 'know' to what category a certain presented sound belongs (in the beginning at least).

6.1.2 The Surface Form (SF)

The SF layer is connected with the AudF and the lexicon layers. In all cases the Surface Form contains twenty nodes, again following Chládková (2014). During the learning phase (but also during perception and production, see section 6.1.5 and 6.1.6) the SF nodes are unclamped. By making use of unclamped SF nodes, unsupervised learning can be guaranteed. During unsupervised learning the network itself 'decides' which nodes on the SF should be activated, based on the learning algorithm (*inoutstar* learning, see chapter 4).

6.1.3 The Underlying Form (UF, or lexicon layers)

As stated before, the networks make use of two lexicon layers to represent the mental lexicon of the first language and the mental lexicon of the second language. The lexicon layers are connected with the SF, but not with each other. Before having been exposed to the L2, the L2 lexicon is not yet connected to the rest of the network. As stated before, for one learning step the lexicon nodes that belong to the selected AudF nodes are activated. During the learning phase the nodes of both lexicon layers are clamped and the activation of the lexicon that does not belong to the presented language is set to zero. Four consecutive nodes in the lexicon belong to one word, i.e. the more words a language contains, the more nodes the lexicon layer consists of.

6.1.4 The concept layer: not included in the networks

During the early stages of setting up the network a concept layer was part of the network as well, following psycholinguistic research (e.g. Potter et al., 1984; Kroll et al., 2002; de Groot, 2012 (for an overview)). In these networks the concept level was connected to the two lexicon layers, in order to simulate the idea that the same concept can be expressed by different words in different languages. However, as it turned out, the concept layer was rather negatively influencing the activation in the network. The excitatory connections between the nodes on the concept layer and the nodes on the lexicon layers were influencing the lexicon layer, not only for production, but also for perception. It is to be wondered whether this is a natural situation. For this reason the models in this study do not make use of a concept layer. However, a concept layer would be a good addition in further research (see chapter 9).

6.1.5 The state of the network during production

After the learning phase the neural networks are able to 'speak' their newly learned language. This is called the *production* of the network. In the networks a speaking bilingual is simulated by selecting a word in one of the two lexicons, after which the activation of the lexicon nodes is spread through the rest of the network, first to the Surface Form and then to the Auditory Form. This causes the activation of the nodes on the SF layer that belong to the activated word in the lexicon and the activation of the nodes at the AudF that belong to the activated word in the lexicon. Furthermore, during the production, the lexicon nodes are clamped, whereas the nodes on the Surface Form and the nodes on the Auditory Form are unclamped.

6.1.5 The state of the network during perception

After the learning phase the networks are not only able to speak their new language, they are also able to understand their new language. This is called the *perception* of the network. In order to simulate a listening bilingual speaker, the AudF nodes are clamped and activated. Which AudF nodes are activated depends on the scenario (new, similar or subset). The rest of the nodes in the network are unclamped. The activity of the AudF nodes is spread through the rest of the network, first to the Surface Form and then to the lexicon layers. This causes the activation of the nodes on the SF layer that belong to the activated AudF nodes and the activation of the words in the lexicon layers that belong to the activated AudF nodes. Note that both lexicons can be activated, in order to account for homophones (see chapter 5).

6.2 The network for the new scenario

In this section the layout and the properties of the network that is modelled for the new scenario are discussed. In the new scenario a second language learner has to learn a new sound category that does not exist in her first language (see chapter 3).

The network has the same general layout as is described in section 6.1. The network starts with learning its first language. For the new scenario, the L1 consists of three vowels: the /i/, the /a/ and the /u/, which are the three outer vowels of the vowel chart (see figure 6.2). These three vowels are chosen because languages with only a limited number of vowels often have at least these three vowels (Maddieson, 2013).

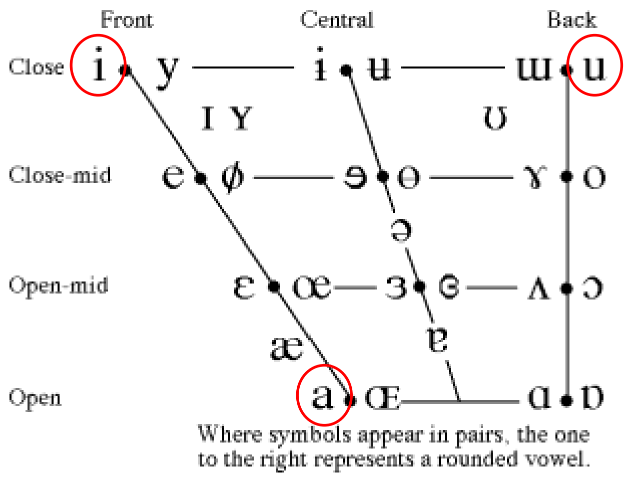


Figure 6.2 – The vowel chart (Ladefoged, 2008)

Following Chládková (2014) I have decided to model the vowels with the F1 and an the F2 only, although the F1 and the F2 are not the only distinctive features for the vowels, as the /i/ and the /u/ on the one hand and the /a/ and the /u/ on the other hand also differ in roundedness. However, Chládková obtained promising results with this approach. The F1 and the F2 are represented on the AudF layer. Chládková is also followed in which AudF nodes are exactly activated. This is based on the ratio between the different formants of the vowels. Table 6.1 shows the mean F1 and F2 values for the three vowels. In the previous section is explained that the categories are formed according to a Gaussian distribution. That means that the values for the F1 and the F2 are not always the ones that are shown in table 6.1, but vary according to the Gaussian distribution. The standard deviation of all the means in table 6.1 is 1.95. To know which node is activated on the AudF one needs to multiply the F1 or the F2 value with the number of AudF nodes. E.g. if the mean F1 and F2 values are selected, the fourth and the 36th node are activated for the /i/: 0.10 * 40 = 4 and 0.90 * 40 = 36. Next to these two nodes also some of the neighbouring nodes are activated, according to the Gaussian function that belongs to AudF layer (see chapter 4).

Vowel	**F1**	**F2**
/i/	0.10	0.90
/a/	0.40	0.75
/u/	0.10	0.60

Table 6.1 – Vowels of the L1 with the means of the corresponding AudF nodes.

As written in the previous section, every vowel belongs to a word on the lexicon layer. As the lexicon layer contains four nodes for every word, the lexicon layer for the L1 consists of twelve nodes. During the learning phase of the neural net, the network is presented with sound-meaning pairs of the language, as explained in the previous section.

Figure 6.3 shows the neural network before any learning steps. One can see that the lexicon for the L2 is not yet connected to the SF layer, so that it cannot influence the acquisition of the first language.

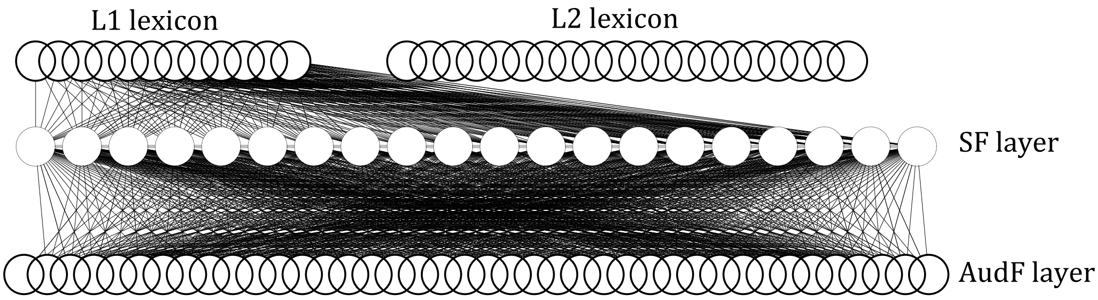


Figure 6.3 – The untrained network for the new scenario

In order to acquire the first language, the network is presented with 50.000 L1 sound-meaning pairs. Although the network has learned its first language after fewer learning steps already, I have chosen for 50.000 sound-meaning pairs to make sure that the network has thoroughly learned the first language, before it is exposed to the second language. In an earlier stage of this study the network was taught the L1 in 10.000 learning steps, after which the network was presented with the L2 sound-meaning pairs. For the new scenario, the results did not differ, but for the similar scenario they did. For this reason in this section only the more 'solid' version of the test, with 50.000 learning steps is presented, whereas for the similar scenario, both versions of the test are discussed.

After 50.000 learning steps the L2 lexicon is connected to the SF, via excitatory connections with a random weight between 0.0 and 0.1. For the L2 the /e/ and the /o/ are added to the vowel inventory. This brings the total number of vowels to five (two vowels more than in the L1). These two new vowels are situated in the middle outsides of the vowel chart. See figure 6.4. An example of a language in which only these five vowels are present is Spanish (Boersma and Escudero, 2002).

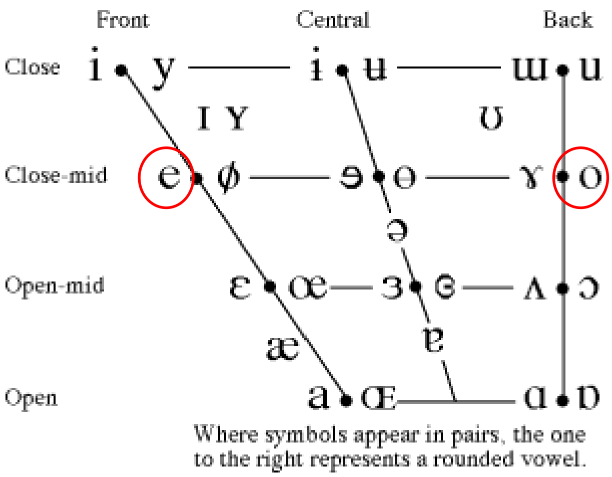


Figure 6.4 – The vowel chart (Ladefoged, 2008)

Again I will follow Chládková (2014) in which nodes are activated on the AudF. The means of the F1 and the F2 can be found in table 6.2. The same calculation as described above for the L1 should be applied to know which nodes on the AudF layer are activated. Again the standard deviation of the means of the F1 and the F2 is 1.95.

Vowel	**F1**	**F2**
/e/	0.25	0.6625
/o/	0.25	0.8375

Table 6.2 – Extra vowels of the L2 and the means of the corresponding F1 and F2 values.

Since the L2 contains five vowels, which all belong to a word in the L2 lexicon, the L2 lexicon consists of fifteen nodes. This can be seen in figure 6.2.

The network is exposed to 5.000 L2 sound-meaning pairs, after which the network is trained on 3000 L1 sound-meaning pairs. Then the network is trained on 10.000 L2 sound-meaning pairs. This is done to see how the phonology layer reacts if the network is trained more thoroughly on its second language. Finally the network is presented with as many L1 sound-meaning pairs as is needed to arrive in a monolingual L1 mode again. Recall from chapter 2 that a monolingual L1 language mode is defined as the state in which the network perceives and produces sounds in an optimal L1 manner. For the networks this may take longer than for real humans.

More than one bilingual speaker needs to be modelled, in order to see whether comparable results are obtained in every training session. For this reason the described approach is repeated seven times, to simulate seven bilingual speakers.

At this point it is also useful to think about what can be expected, considering the two theories that are discussed in the theory section of this thesis (separate phonological systems vs. one merged phonological system). In sections 6.2.1 and 6.2.2 the expectations are discussed.

6.2.1 Expectations separate phonological systems

The first theory about the phonological category creation of late bilingual speakers assumes that bilingual speakers have two different phonological systems. Depending on the language mode, bilingual speakers switch between these two systems. For the neural networks this would mean that, both for production and for perception, a difference should be observed between the way in which the SF layer is activated for the first language and the way in which the SF layer is activated for the second language. After all, if there is no difference between the activation of the SF nodes for the first and the second language, one cannot speak about separate phonological systems. This would also mean that sound-meaning pairs that only exist in the second language should be perceived differently in the first language mode than in the second language mode. It is to be expected that new L2 sounds that are very distinct from the already known L1 sounds are still perceived in an L2 manner if they are presented in the L1 surroundings. (E.g. speakers of languages without clicks are still able to perceive a click as a sound from a different category than the categories that exist in their native language.) On the other hand, new L2 sounds that are less distinct than the already known L1 sounds, are expected to be perceived in an L1 manner in the L1 surroundings, but in an L2 manner in L2 surroundings. It is to be seen how different the new L2 sounds that are used for this scenario are from the already known L1 sounds.

The nodes that are activated for the phonological categories of the first language should not change after or during the acquisition of the second language, because this would lead to Flege's notions of assimilation and dissimilation (see next section). Note that for homophones either the same nodes on the SF layer could be activated for the L2 and for the L1, or different nodes, without rejecting the separate system theory.

6.2.2 Expectations one merged phonological system

If Flege's SLM model (Flege, 2005) were the correct model and a bilingual speaker did not have separate phonological systems for her L1 and L2, but one merged bilingual system, assimilation and/or dissimilation should be observed, as Flege states that the acquisition of new sound categories always occurs either via assimilation or dissimilation. Assimilation would be observed if a new L2 category merges with an already existing L1 category. The same SF nodes should be activated for both categories in the L1 and in the L2. The activated nodes should be different from the nodes that were primarily activated for the category in the L1, as this would show the expansion of the former L1 category by the new L2 category. Dissimilation would be observed if the SF nodes that are activated for an L1 sound-meaning pair change after the acquisition of the second language (without merging with another category as this would be assimilation). Again it is to be seen whether the new L2 categories are similar to the L1 categories or whether the new L2 categories are distinct from the L1. The first option would result in assimilation, whereas the second option would result in dissimilation. Flege does not give very clear conditions for this distinction.

Next to the perception, also the production would change. Not only would the production of the L2 differ from the production of the same language by a native speaker, but the production of the L1 would also change after acquiring the second language, as Flege predicts an L2 accent in the L1.

In chapter 7 of this work the results of the testing are presented and in chapter 8 the results are discussed.

6.3 The network for the similar scenario

In this section the general lay-out and the characteristics of the network for the similar scenario are discussed. As explained in chapter 3, the number of categories does not change in the similar scenario; only the place of the boundary between the categories. In order to test this scenario, two very simple toy languages are constructed. Both languages contain two categories, but the place of the boundary between those two categories is slightly different for both languages. One can imagine that the different VOT boundary between /t/ and /d/ in French and in English is investigated (see chapter 3). Table 6.3 gives an overview of the means of the two categories in both languages. The nodes that are activated on the AudF should be calculated in the same way as is explained in section 6.3. Again the standard deviation of the means of the F1 and the F2 is 1.95. Figure 6.5 shows the distribution graphically.

L1		**L2**	
Mean of category 1	**Mean of category 2**	**Mean of category 1**	**Mean of category 2**
0.45	0.65	0.50	0.70

Table 6.3 – Sound categories in the L1 and the L2 and their corresponding means

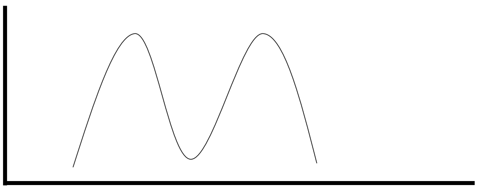


Figure 6.5a – The distribution in the L1

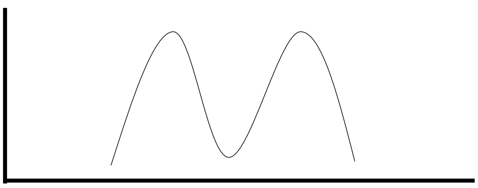


Figure 6.5b – The distribution in the L2

Figure 6.6 shows the untrained network for the similar scenario. Again, the L2 lexicon is already in place, but not yet connected to the SF, as the network starts with learning the L1.

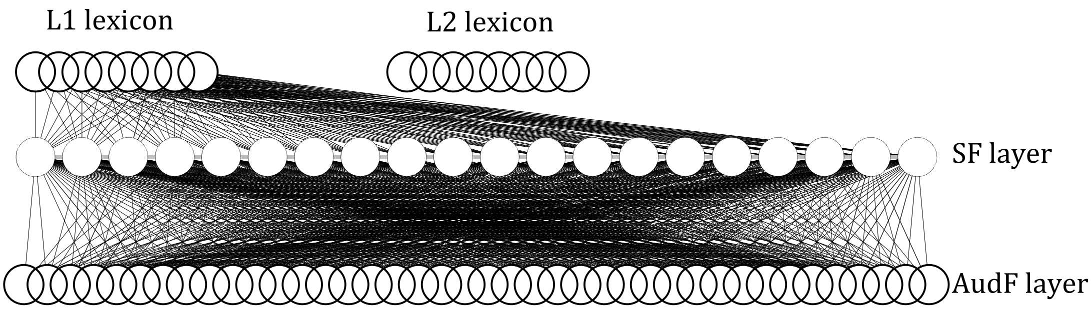


Figure 6.6 – The untrained network for the similar scenario

As already mentioned in the previous section, this scenario has been tested in two ways. In an early phase of testing, the L1 was learned in 10.000 learning steps, after which the network was exposed to the L2 sound-meaning pairs. Also the more solid version, in which the L2 is only presented after 50.000 learning steps has been tested. The differences in results that are obtained with these two different ways of testing are very interesting. For this reason I have decided to present the two ways of testing here and to present the results of both testing sessions in the next chapter.

In the first testing situation, the L2 lexicon is connected to the SF layer after training the network on the L1 for 10.000 times, via excitatory connections with random weights varying from 0.0 to 0.1. After this the network is exposed to 5.000 L2 sound-meaning pairs. Then 5.000 L1 sound-meaning pairs are presented, after which the network is trained on the L2 for 10.000 learning steps. Finally the network is exposed to the first language again and is trained on as many L1 sound-meaning pairs as is necessary to arrive in a full monolingual L1 mode again (see chapter 2 for a detailed explanation on language modes). Just like in the previous section it is to be remarked that this takes longer than one may expect, so again, see chapter 9 for ideas for further research. This training procedure is repeated for seven new networks, again in order to simulate seven bilingual speakers.

In the second testing situation, the L2 lexicon is connected to the SF layer after 50.000 L1 learning steps, again via excitatory connections with random weights varying from 0.0 to 0.1. After that the network is exposed to 5.000 L2 sound-meaning pairs. After these 5.000 learning steps the network is exposed to 10.000 L1 sound-meaning pairs. Finally the network is trained on the L2 for 10.000 learning steps and successively on the L1 for 10.000 learning steps. As the reader may have noticed this procedure is slightly different from the procedure that is followed during the training of the model for the new scenario. I have decided to train the model for the similar scenario this way because best insights were obtained following this approach. Just like before, this procedure is repeated for seven new networks in order to simulate seven bilingual speakers.

Also for this scenario different theories about the bilingual phonological system(s) lead to different expectations on how the network will behave. In the next two sections (6.3.1 and 6.3.2) the different expectations are discussed.

6.3.1 Expectations separate phonological systems

If a bilingual speakers has two separate phonological systems, sounds that lay in between the two different category boundaries in the L1 and the L2, are classified different in an L1 language mode than in an L2 language mode, just like described by Garcia-Sierra, Diehl and Camplin (2009), see chapter 3. Figure 6.7 shows this graphically. The two red lines show the category boundaries in the two languages. One can observe that a sound between these two red lines belongs to the second category in the one language and to the first category in the other language. For production this would mean that the sounds between the two boundaries are pronounced differently, depending on the language environment, whereas for perception this would mean that the sounds between the two boundaries are perceived differently, depending on the language environment, i.e. a boundary shift occurs for both perception and production.

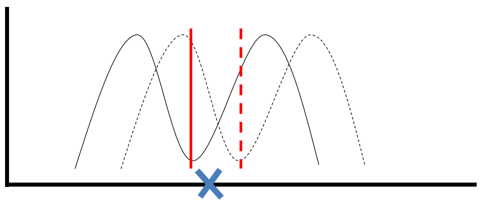


Figure 6.7 – The distribution in the L1 and in the L2

A different activation of the SF nodes for categories in the L1 and in the L2 is not necessarily required for a separate system hypothesis but it would strengthen the hypothesis of separate phonological systems. I.e. if different SF nodes are activated for the first category in the L1 than for the first category in the L2, this may show that the bilinguals have two separate phonological systems. However, if the same nodes are activated for the first category in the L1 and the first category in the L2, this does not necessarily mean that the bilingual speaker does not have separate phonological systems. It is more important to examine whether the place of the boundary between the two categories is different in the L1 and in the L2.

6.3.2 Expectations one merged phonological system

If one assumes bilinguals to have one single bilingual phonological system the first category in the L1 is expected to assimilate with the first category in the L2, just like the second L1 category is expected to assimilate with the second L2 category. Because of that one would expect that, both for production and for perception, the nodes that are activated on the SF are the same for the two categories in the L1 and in the L2, but that the nodes that are activated in the L1 change after the acquisition of the L2, due to assimilation. Besides that the network should show a difference in production of the sounds in the L1 after acquiring the L2.

In chapter 7 of this work the results of the testing are presented and in chapter 8 the results are discussed.

6.4 The network for the subset scenario

The last network that is presented here is a network that models the subset scenario. As explained in chapter 3, the subset scenario can be seen as the opposite of the new scenario. This means that the bilingual's first language contains more categories than her second language.

The network that is used to model this scenario is, just like the scenario itself, the opposite of the network to test the new scenario. The network starts to learn a first language that contains five sound-meaning pairs. The sound categories are exactly the same sound categories as the ones that are summed up in table 6.1 and table 6.2. The untrained network that is used to model the subset scenario can be seen in figure 6.8.

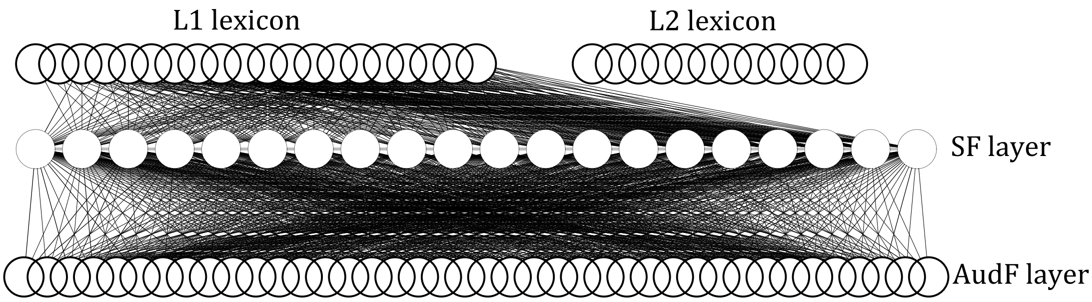


Figure 6.8 – The untrained network for the subset scenario

Whereas the L1 of this learner contains five sound-meaning pairs, the L2 of this learner contains only three sound-meaning pairs. The L2 that is used to model the subset scenario is the same language as the L1 that is used to model the new scenario: the three sound categories that are used are the same sound categories as the ones in table 6.1.

Also this scenario is tested twice. Again, during the first testing phase the network was taught its first language in 10.000 learning steps, and during the second testing phase the network had to learn its first language in 50.000 learning steps. The results for these two ways of testing did not differ. For this reason I have chosen to limit the description of the testing and the presentation of the results in the next chapter to the networks that have learned their first language in 50.000 learning steps.

After 50.000 L1 learning steps the excitatory connections between the SF and the L2 lexicon are added to the network. Also these connections are given a random weight between 0.0 and 0.1 again. Then the network is exposed to 5.000 L2 learning steps. Then the network is trained on the L1 again, for 3000 learning steps. After that the network is presented with 10.000 L2 sound-meaning pairs, after which the network is exposed to at least 5.000 L1 sound-meaning pairs. This procedure is repeated seven times, in order to simulate seven late-bilinguals. One may recognise that this is the same approach as the one that is used for the new scenario.

Just like for the other two scenarios, also here it should be investigated what the network is expected to do, depending on the two theories that are discussed through this entire work. In the next two sections the expectations are discussed. However, one will notice that the less radical differences between the two theories are expected for this scenario.

6.4.1 Expectations separate phonological systems

As for the new scenario all the L2 sounds already exist in the first language, all the L2 sound-meaning pairs have a homophone in the L1. In section 6.2 it was already argued that, both for perception and for production, the SF nodes that are activated in the L1 could either be different from the SF nodes that are activated for the same sounds in the L2, or the same nodes could be activated.

6.4.2 Expectations one merged phonological system

Following the reasoning of Flege's SLM (Flege, 2005), for the merged phonological system the nodes that are activated on the SF should be the same for the L1 and the L2. In the previous two scenarios the activation of the nodes could differ due to assimilation and dissimilation, but as for the subset scenario no new categories need to be learned, this is not to be expected.

If the same SF nodes are activated for the L2 sounds as for the already existing L1 sounds, it will be difficult to argue against one of the two theories. Next to that, and this could be predicted by both theories, the second language learner can have difficulties categorizing speech sounds in the correct sound category, due to the fact that she has 'too much' knowledge of how the speech sounds continuum could be divided as well (see chapter 3 and Escudero and Boersma (2002). In chapter 7 of this work the results of the testing are presented and in chapter 8 the results are discussed.

7. THE RESULTS

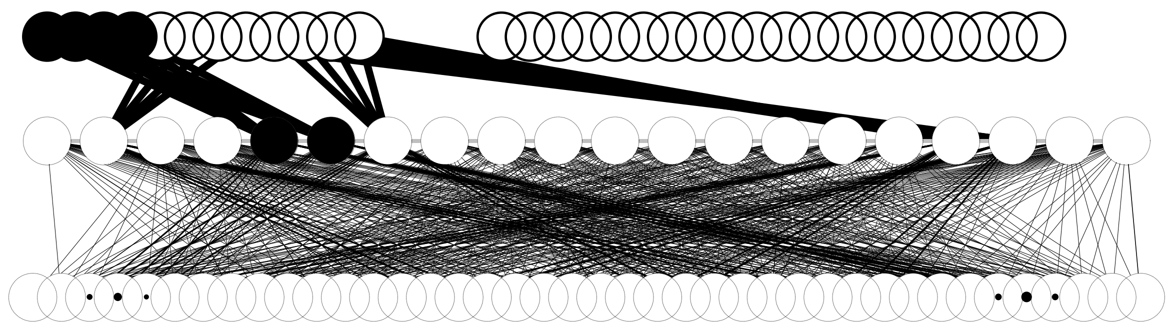
In this chapter the results of the simulations are presented. Every scenario is given a separate section. As the results for the different networks within a situation are very comparable, I have chosen to present the results of one particular instance per scenario, in order to give the best insight in the results.

7.1 The new scenario

In this section the results for the new scenarios are presented. Only the results of the model that has acquired its first language in 50.000 training steps are taken into account. In section 7.1.1 the results for the production of the networks are presented and in section 7.1.2 the results for the perception of the networks are presented.

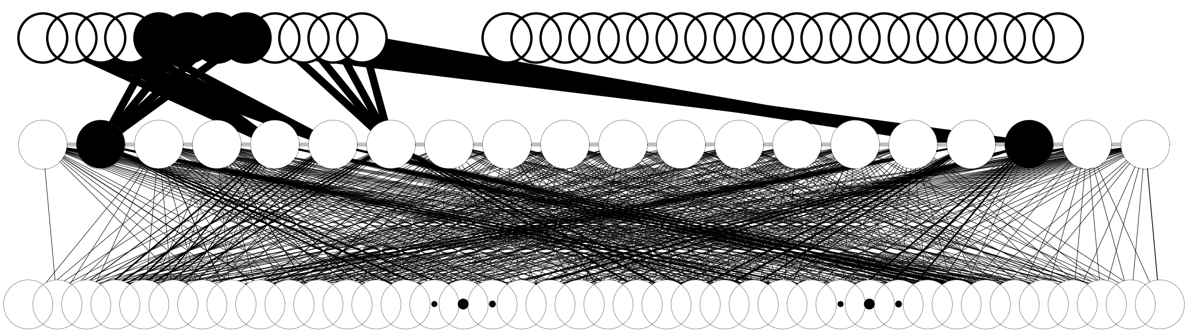
7.1.1 The production of the network for the new scenario

Figures 7.1a-c show a neural network after 50.000 learning steps. The figures show the network in the production phase: the lexicon nodes are clamped, whereas the SF nodes and the AudF nodes are unclamped (see chapter 6). One can see that the networks have successfully learned their first language, as the correct nodes at the AudF are activated after activating the lexicon, e.g. for the 'I' the beginning and the end of the AudF are activated, according to its F1 and its F2. Phonological categories have been formed on the SF layer, i.e. certain nodes on the SF layer belong to certain sound-meaning pairs. Next to that certain connections have been strengthened, whereas other connections have been weakened. Some connections have disappeared. For example, there is no connection left between the first L1 lexicon node and the first SF node. Note that the nodes for the L2 lexicon are already in place, but not yet connected to the SF layer.



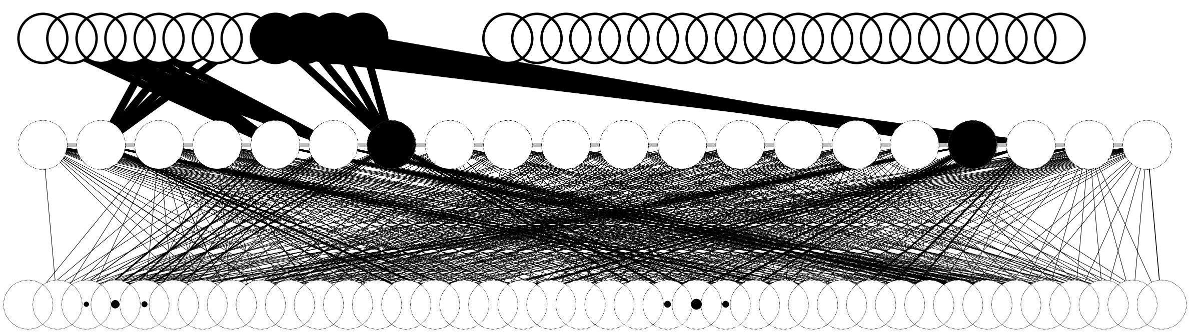
Observations: Two nodes are activated on the SF: 5 and 6.

Figure 7.1a – Production of the ‘I’ in the L1



Observations: Two nodes are activated on the SF: 2 and 18.

Figure 7.1b – Production of the ‘A’ in the L1

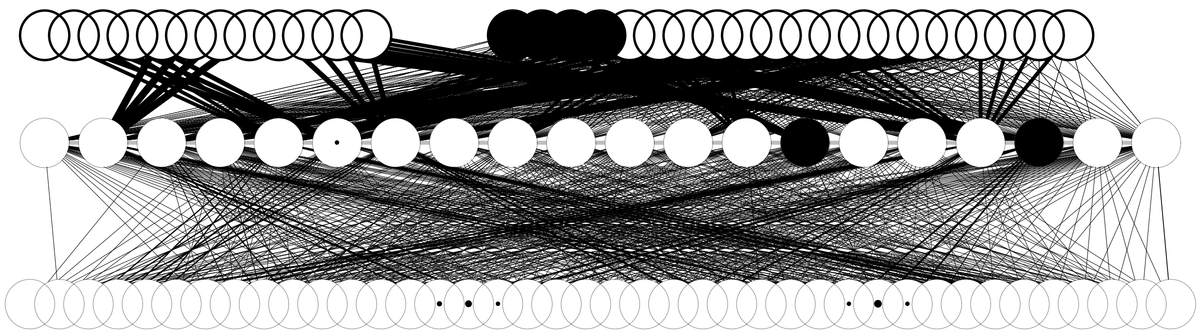


Observations: Two nodes are activated on the SF: 7 and 17.

Figure 7.1c – Production of the ‘U’ in the L1

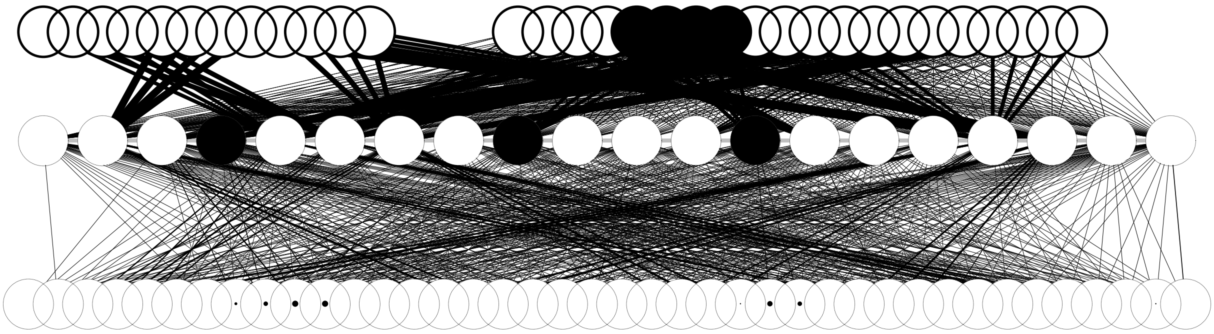
Figures 7.2a-e show the distribution of the SF nodes after training the model on the second language for 5.000 sound-meaning pairs. One may notice that the first word in the L2 lexicon does not correspond to the 'I', as it did in the L1, and that the other sound-meaning pairs have changed position as well. This is done to make sure that the order of the words in the lexicon does not influence the acquisition of the sound categories. However, this is rather an adaptation that was made during the time that the network still contained a concept layer (see chapter 5), so it is not very important to look at now.

Next to that one may observe that now the L2 lexicon layer is connected to the Surface Form. Besides that, some of the connections between the L1 lexicon layer and the Surface Form have become weaker. Recall from chapter four that the connection strength between two nodes decreases when one of the nodes is activated, whereas the other node is not. This can be seen by comparing figures 7.1a and 7.2c: the connection between SF node number 5 and the L1 lexicon layer and between SF node number 6 and the L1 lexicon have become weaker, because, during the L2 acquisition, these two SF nodes are activated together with nodes on the L2 lexicon layer, whereas the L1 lexicon layer is not activated. The subscript '2' is used for words in the L2 that belong to sounds that were already known in the L1, e.g. 'I_2' in the second language belongs to the same sound as 'I' in the first language.



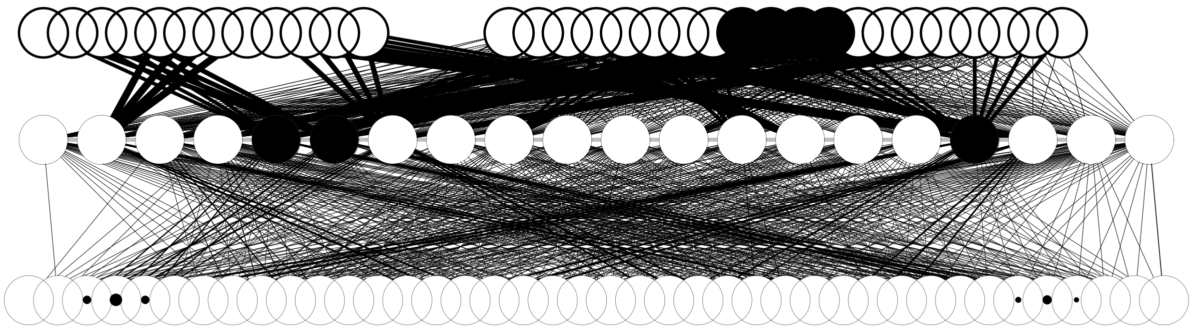
Observations: Two/ three nodes are activated at the SF: (6), 14 and 18. Node 6 is also activated for the 'I' and the 'I_2'. Node 18 is also activated for the 'A' in the L1.

Figure 7.2a – Production of the 'A_2' in the L2



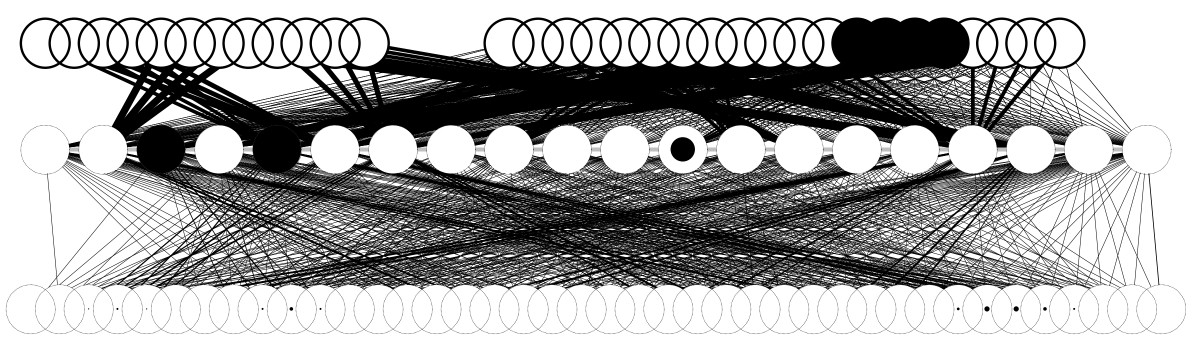
Observations: Three nodes are activated at the SF: 4, 9 and 13. Four nodes are activated for the F1 on the AudF, whereas till now always three nodes per formant were activated at the AudF. This changes after training more thoroughly.

Figure 7.2b – Production of the ‘E_2’ in the L2



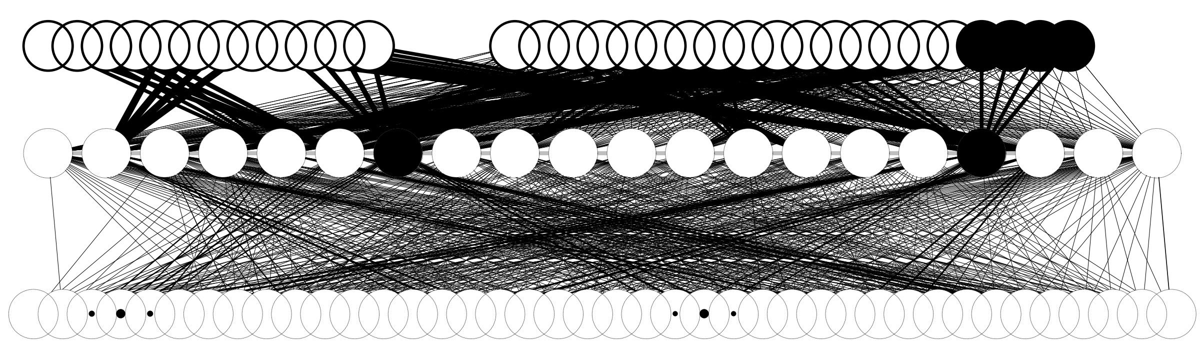
Observations: Three nodes are activated at the SF: 5, 6 and 17. Nodes 5 and 6 are also activated for the ‘I’ in the L1. Node 17 is also activated for the ‘U’ and the ‘U_2’. The /u/ and the /i/ have their first formant in common. Node 6 is also slightly activated for the ‘A_2’.

Figure 7.2c – Production of the ‘I_2’ in the L2



Observations: Two / three nodes are activated at the SF: 3, 5 and (12). At the AudF five nodes are activated for the F2, three nodes are activated around 0.10 and three nodes are activated for the F1. The activation around 0.10 is caused by the fact that the fifth SF node is also activated for the 'I' and the 'I_2'. After training more thoroughly the AudF is only activated for the F1 and for the F2.

Figure 7.2d – Production of the 'O_2' in the L2

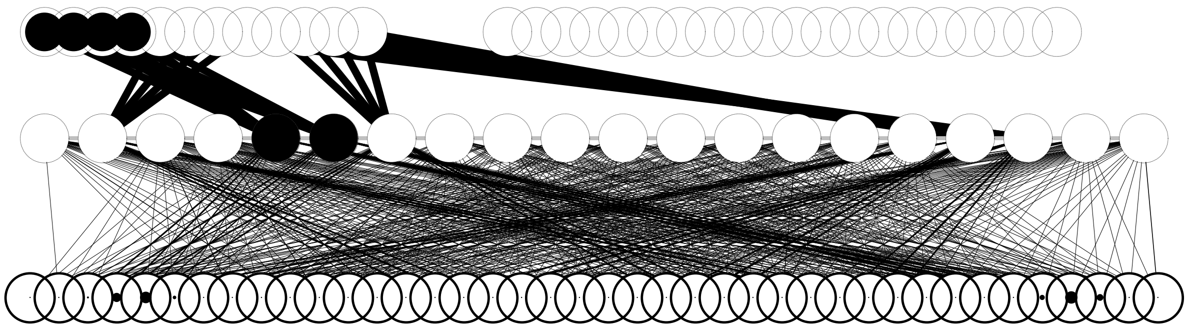


Observations: Two nodes are activated at the SF: 7 and 17. These are exactly the same nodes as were activated for the 'U' in the L1 (see figure 7.1c).

Figure 7.2e – Production of the 'U_2' in L2

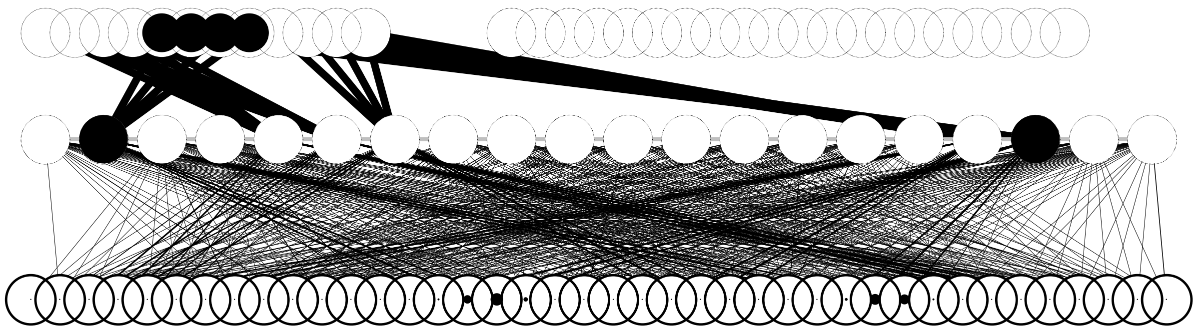
7.1.2 The perception of the network for the new scenario

Figures 7.3a-c and figures 7.4a-e show the perception of the network: the nodes at the AudF are clamped, whereas the nodes at the SF and the nodes on the lexicon layers are unclamped (see chapter 6). Figures 7.3a-c show the perception of the first language, whereas figures 7.4a-e show the perception of the second language. One can see that the networks have successfully acquired the perception, i.e. the correct lexicon nodes are activated for the activation at the AudF and next to that the same phonological categories are used as during the production of the network.



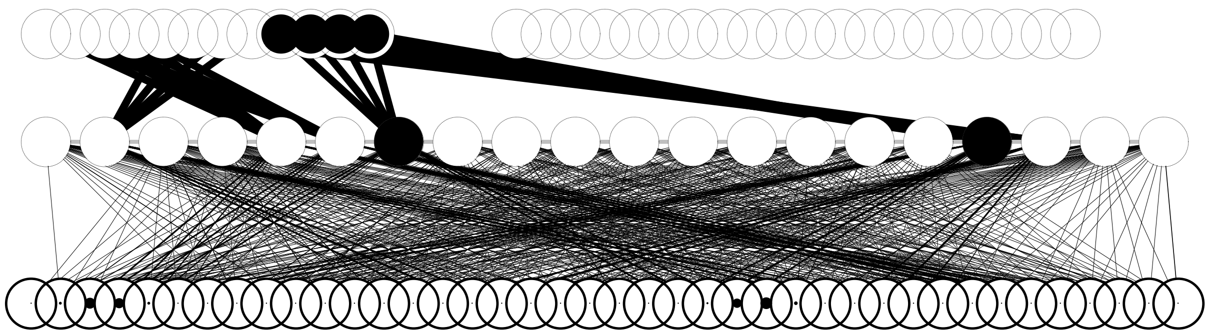
Observations: Two nodes are activated at the SF: 5 and 6. These nodes are the same as during the production of the 'I'.
The activated nodes on the AudF are not exactly the same as in figure 7.1a. Due to distributional learning also sounds near the mean of the F1 and the mean of the F2 of the /i/ are recognised as such.

Figure 7.3a – Perception of the 'I' in the L1



Observations: Two nodes are activated at the SF: 2 and 8. These nodes are the same as during the production of the 'A'.
The activated nodes on the AudF are not exactly the same as in figure 7.1b. Due to distributional learning also sounds near the mean of the F1 and the mean of the F2 of the /a/ are recognised as such.

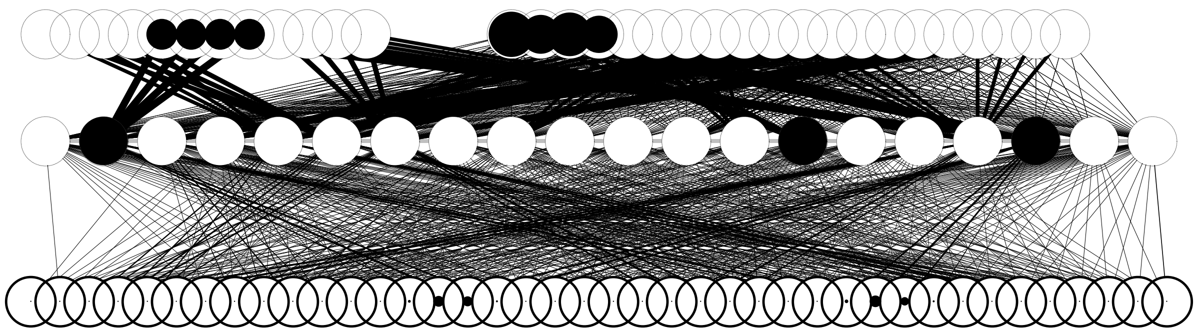
Figure 7.3b – Perception of the 'A' in the L1



Observations: Two nodes are activated at the SF: 7 and 17. These nodes are the same as during the production of the 'U'.
The activated nodes on the AudF are not exactly the same as in figure 7.1c. Due to distributional learning also sounds near the mean of the F1 and the mean of the F2 of the /u/ are recognised as such.

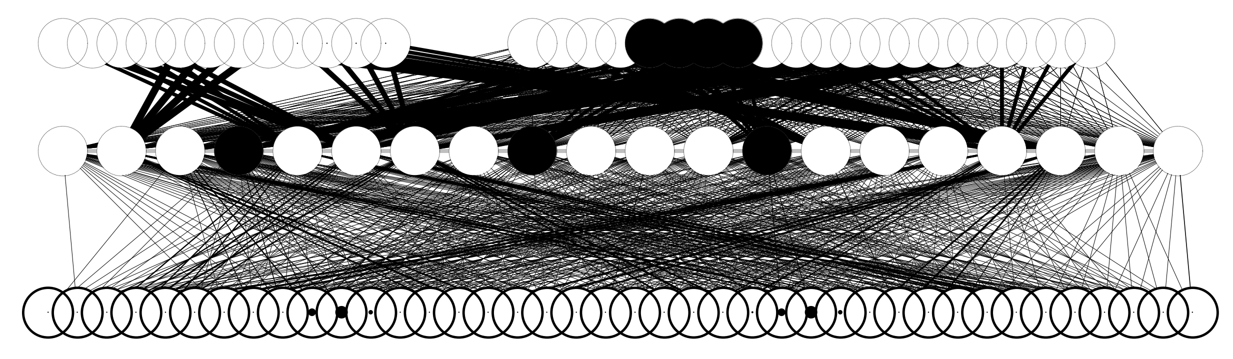
Figure 7.3c – Perception of the 'U' in the L1

The perception of second language is shown in the next five figures. The perception of the categories is acquired, but the influence of the first language is still visable.



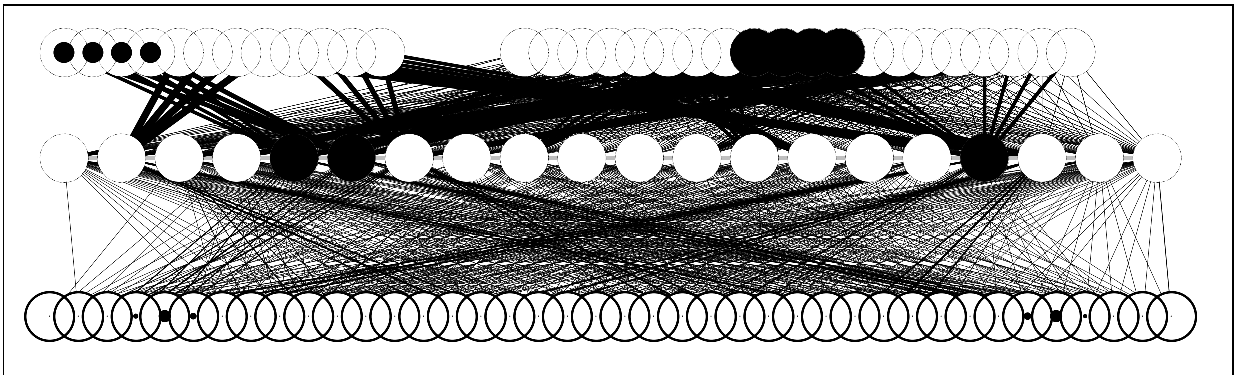
Observations: Three SF nodes are activated: 2, 14 and 18. Node 2 is not activated during the production of the 'A_2' (see figure 7.2a), but this node *is* activated during the production of the 'A' (see figure 7.1b). The L1 lexicon is still activated, even though the network had been trained on the L2 for 5.000 steps. This was predicted for homophones (see chapter 5 and 6).

Figure 7.4a – Perception of the 'A_2' in the L2



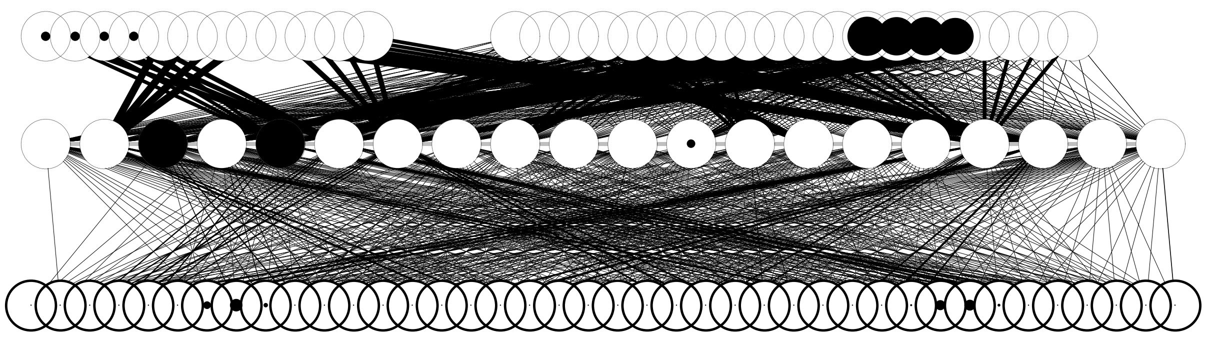
Observations: Three SF nodes are activated: 4, 9 and 13. These are the same nodes as were activated during the production of the 'E'. The perception of this new L2 sound is correctly acquired.

Figure 7.4b – Perception of the 'E' in the L2



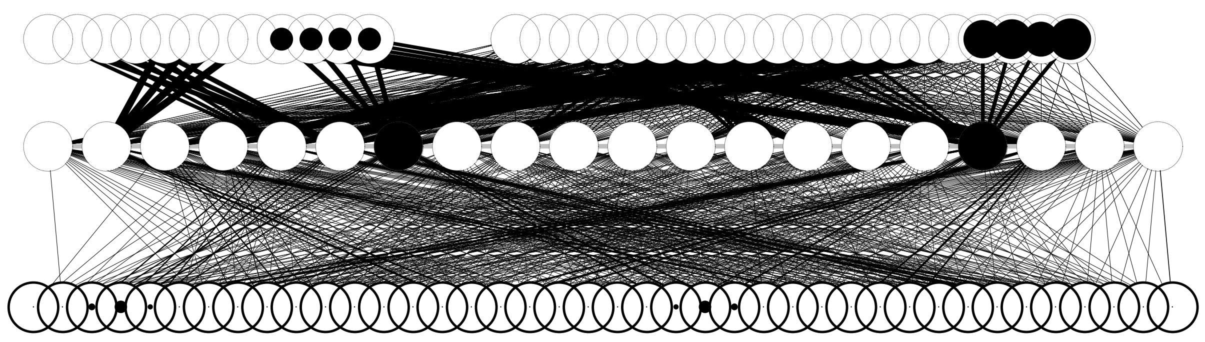
Observations: Three SF nodes are activated: 5, 6 and 17. These are the same nodes as are activated during the perception and the production of the 'I' in the L1. The L1 lexicon is also activated, for the 'I', but not as much as the L2 lexicon is activated for the 'I_2'.

Figure 7.4c – Perception of the 'I_2' in the L2



Observations: Two / three nodes are activated at the SF: 3, 5 and (12). These are the same nodes as were activated during the production. The perception of this new L2 sound is correctly acquired. In addition, the 'I' in the L1 lexicon is slightly activated. For the 'I' and the 'O' the fifth SF node is activated. This causes the activation in the L1 lexicon.

Figure 7.4d – Perception of the 'O' in the L2



Observations

Two nodes are activated at the SF layer: 7 and 17. These are the same nodes as during the production of the 'U_2'. The network has correctly acquired the perception of the /U_2/. In the L1 lexicon the nodes for the 'U' are activated.

Figure 7.4e – Perception of the 'U_2' in L2

It is interesting to investigate how sounds that are present in the L2 but not in the L1 are perceived when the network is in the L1 surroundings. Figures 7.5a-d show that the networks perceive the unknown L2 sounds as sounds in the L1. Note that these figures show a network that has not learned a second language yet. The /e/ is either perceived as /a/, but most often it is perceived as /u/ (figures 7.5a-b), depending on the exact point of activation of the AudF. The /o/

is either perceived as /i/ or as /a/ (figures 7.5c-d), also depending on the exact point of activation of the AudF.

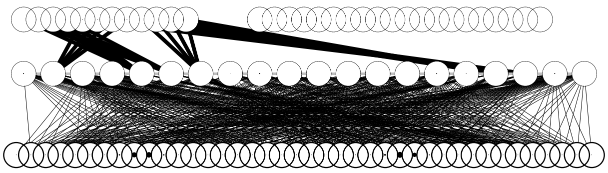


Figure 7.5a - /e/ perceived as /a/

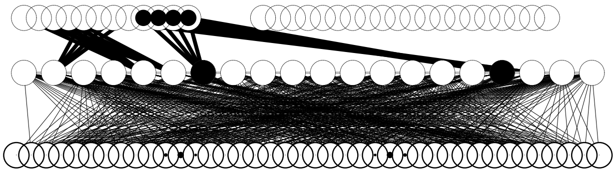


Figure 7.5b - /e/ perceived as /u/

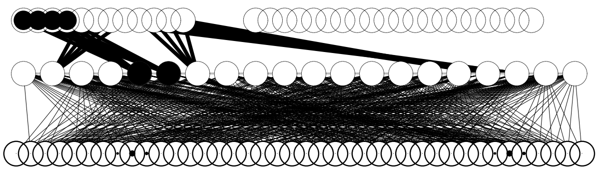


Figure 7.5c - /o/ perceived as /i/

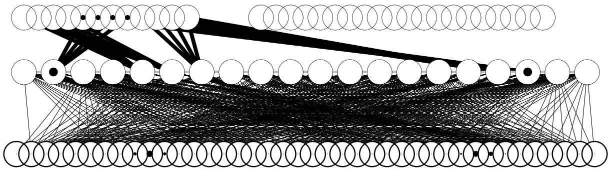


Figure 7.5d - /o/ perceived as /a/

7.1.3 Production and perception after more training

Up until this point stages of the networks have been presented after 50.000 L1 learning steps and 5.000 L2 learning steps. As explained in chapter 6, the networks have been trained on their first and their second language more often, after these initial 55.000 steps. The question here is whether the networks that are more thoroughly trained behave the same for production and perception as the networks that are presented above.

After the initial 50.000 L1 learning steps and the subsequent 5.000 L2 learning steps, the networks switch to the first language surroundings again. Already without having been exposed to any additional L1 sound-meaning pairs, the network is able to produce the L1 words in an L1 manner (so, in the same way as in figures 7.1a-c). However, the SF nodes and the AudF nodes are slightly less activated, as the connections between the L1 lexicon and the SF layer have become weaker due to the exposure to the L2 (see chapter 4 and section 7.1.1). In most cases this effect disappears after presenting the network with another 3000 L1 sound-meaning pairs.

These 3000 L1 sound-meaning pairs were also needed to achieve a successful L1 perception again for the L1 sounds (comparable to figures 7.3a-c). For sounds that occur in *both* languages, *both* lexicon layers are activated. Sounds that only occur in the L2 are sometimes still perceived as L2 sounds and words, although the L1 lexicon is activated as well (see figure 7.6a). On the other hand, sometimes new L2 sounds are perceived as sounds in the L1 and then also the words in the L2 lexicon that belong to that sound are activated (see figure 7.6b for an example). Note that this is different from the results that were presented in figures 7.5a-d.

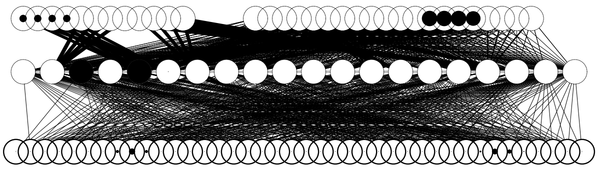


Figure 7.6a – The /o/ is still perceived in an L2 manner after training the network on the L1

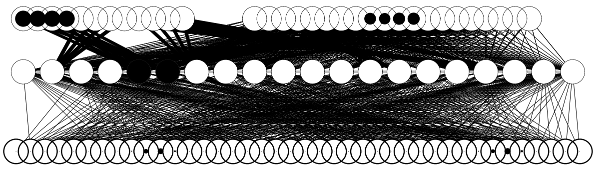


Figure 7.6b – The /o/ is perceived as /i/, both in L1 and in the L2 lexicon

Now the networks switch back to the L2 surroundings. Again, without any additional training on the L2 the networks produce the L2 words in an L2 manner. Now the networks are exposed to 10.000 L2 sound-meaning pairs. After this training the networks are able to perceive the incoming sounds in an L2 manner and the L1 lexicon is less activated than it was before (see figure 7.7 for an example). Once double activation within one lexicon layer could be observed, i.e. more nodes than the nodes for one word were activated: the /i/ activated the nodes in the L2 lexicon for both the ‘I_2’ and the ‘U_2’ and so did the /u/.

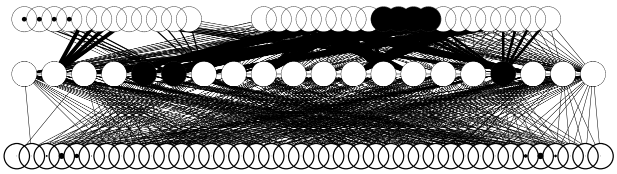


Figure 7.7 – Less activation in the L1 lexicon

Finally the networks are exposed to 5.000 L1 sound-meaning pairs. As is to be expected by now, the networks are able to correctly produce the L1 words already before this training. The networks did not acquire a full L1 perception after the 5.000 learning steps. However, more L1 learning steps did cause that the networks perceived the L1 sounds in a pure L1 manner again, except in a few instances. In these few instances the networks kept activating L2 SF nodes for certain sounds.

As a general remark it is still left to say that for the production, even though the SF nodes were immediately activated in the ‘correct’ language mode, sometimes more *AudF* nodes were activated than before the exposure to the other language (see figure 7.8, in which one can also observe that the SF nodes are less activated). More training on the same language mode let the activation of the additional AudF nodes disappear.

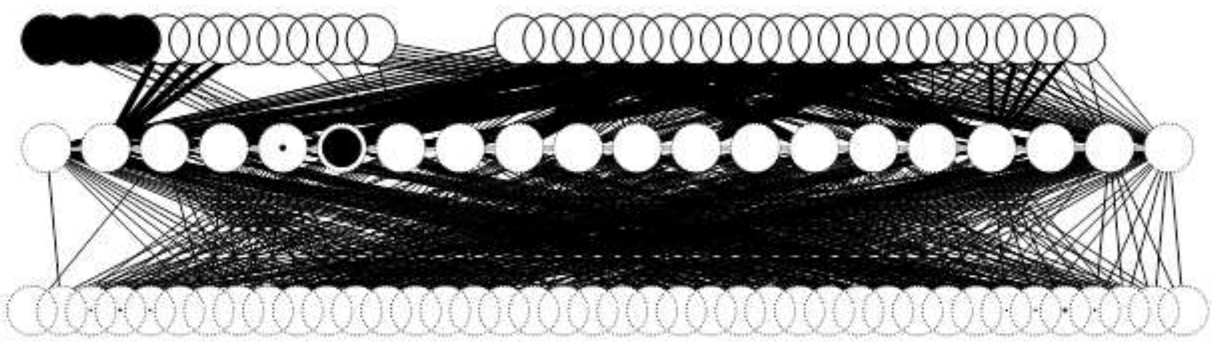


Figure 7.8 – More nodes activated on the AudF and less activation on the SF

7.2 The similar scenario

As has already been stated in the previous chapter, it is interesting and important to present the results of teaching the networks for the similar scenario their first language in 10.000 steps and the results of teaching the networks their first language in 50.000 steps.

Following the same approach as in the previous section, the results of training the networks on the L1 for 50.000 steps are presented first. This is done by looking at the specific example of one trained network, as this example is comparable to the results that were obtained by the other training instances. Noteworthy different results will be added to the specific example.

The main question that needs to be answered for this scenario is whether a boundary shift occurs between the two language environments. Recall from chapter six that sounds between the two boundaries are classified differently in one language than in the other language (figure 6.7 is repeated below).

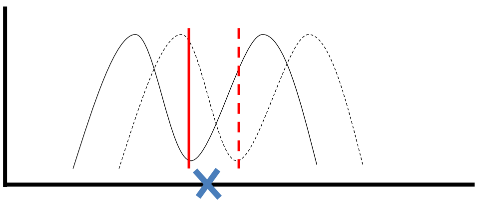
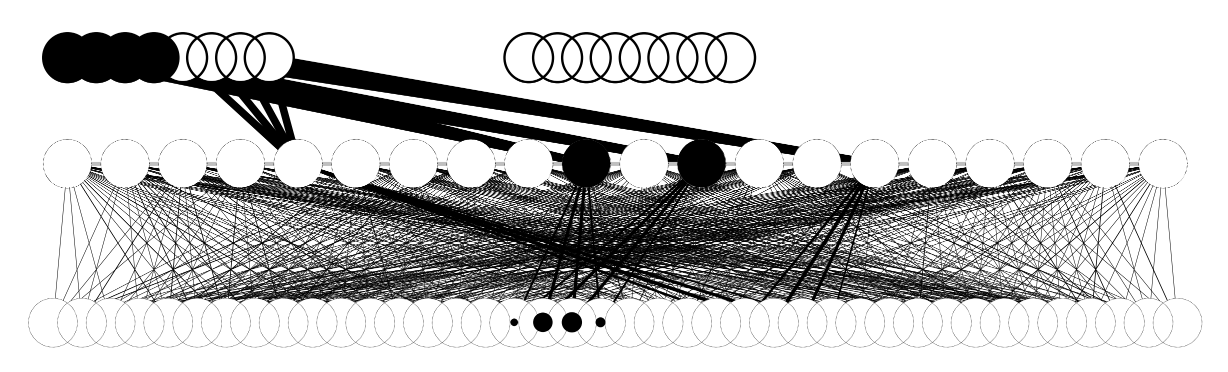


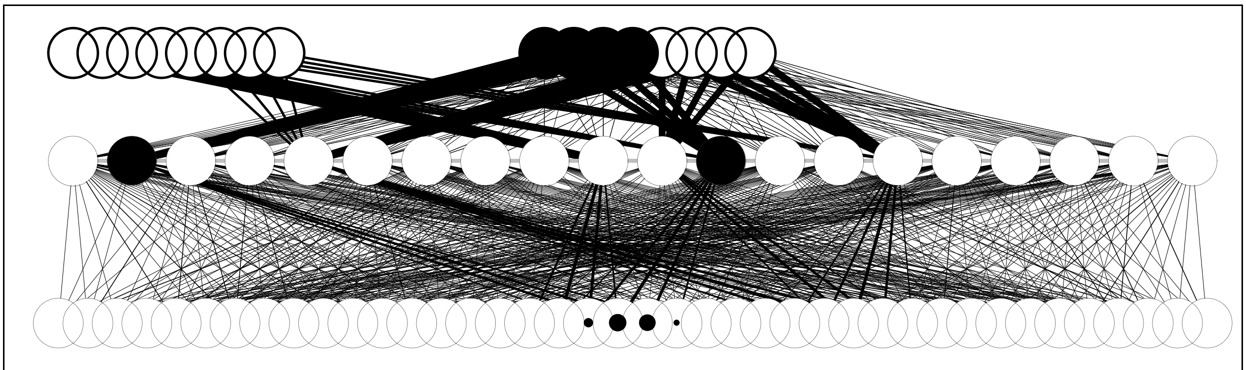
Figure 6.7 – The distribution in the L1 and in the L2

For this reason all three figures 7.9, 7.10 and 7.11 show two stages of the network: first the network that is exposed to the first language for 50.000 training steps and then the network that is trained on the second language for 5.000 times. Figures 7.9 and 7.10 show the network during the production (the lexicon nodes are clamped, the other nodes are unclamped) and figure 7.11 shows the network during perception (only the AudF nodes are clamped, whereas the rest of the nodes are unclamped). By visually presenting the models this way, the reader will get the chance to see whether a boundary shift occurs. This time the AudF is only activated at one point. Recall from chapter 6 that, in order to test the similar scenario, the acquisition of the Voice Onset Time in two languages is modelled, instead of the acquisition of vowels. Considering the production, the networks successfully acquired their first and second language. Considering the perception, the networks successfully acquired their first language, but not necessarily their second language. This is illustrated by the figures.



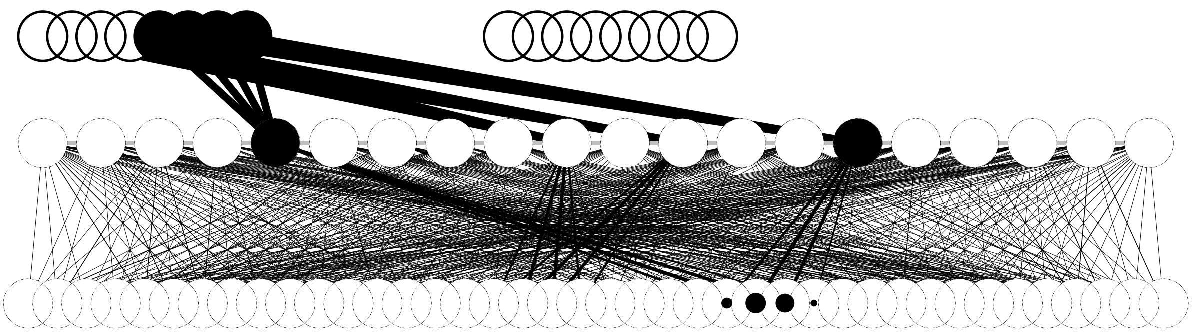
Observations: Two nodes are activated on the SF: 10 and 12.

Figure 7.9a – Production of the first category in L1



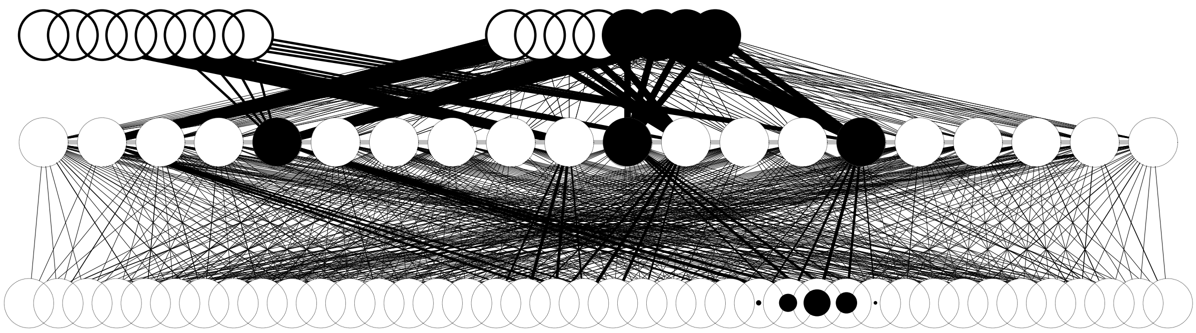
Observations: Two nodes are activated on the SF: 2 and 12. Node 12 is also activated for the first category in the L1. A boundary shift can be observed: the AudF is activated at a different place for the L1 and the L2.

Figure 7.9b – Production of the first category in L2



Observations: Two SF nodes are activated: 5 and 15.

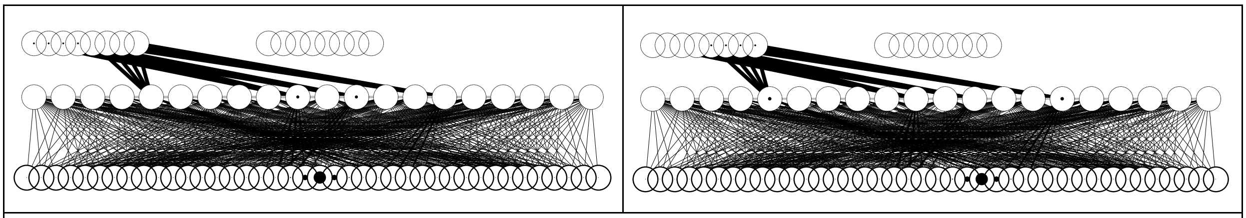
Figure 7.10a – Production of the second category in the L1



Observations: Three nodes are activated on the SF: 5, 11 and 15. Node 5 and node 15 are also activated for the second category in the L1. Again, a boundary shift can be observed, as the AudF is activated at a different place in the L1 than in the L2.

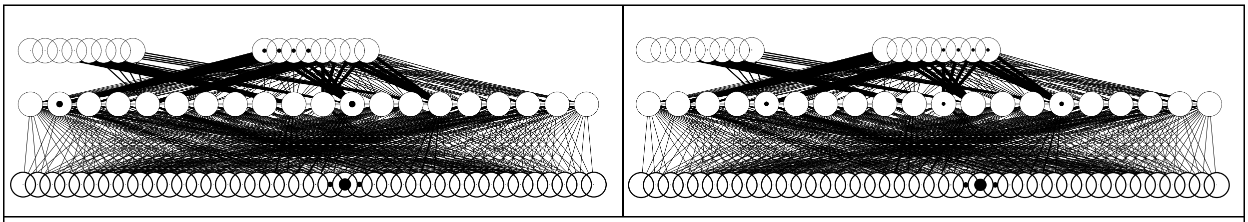
Figure 7.10b – Production of the second category in the L2

For production a boundary shift could be observed. The question is whether a comparable boundary shift can be observed for the perception of the network. Figures 7.11a and 7.11b show the boundary between the first and the second category in the L1 and in the L2. The left networks in the figures show the last AudF activation that is classified as the first category and the second networks in the figures show the first AudF activation that is classified as the second category.



Observations: The nodes that are activated at the AudF in the left picture are not the direct neighbouring nodes of the AudF nodes that are activated in the right picture: there is an area at the AudF that is not classified as one of the two categories.

Figure 7.11a – Boundary between the two categories in the L1, perception



Observations: There is no area at the AudF that is not classified as one of the two categories. Due to this a boundary shift can be observed if one compares the left images of figures 7.11a and 7.11b, but no boundary shift is present between the right images of these figures. However, note that different SF nodes are activated in the L1 than in the L2, so there still is a difference in perception between the L1 and the L2. Next to that it is to be remarked that one instance of the networks did show a boundary shift in both instances.

Figure 7.11b – Boundary between the two categories in the L2, perception

7.2.3 Production and perception after more training

After these 50.000 learning steps in the L1 and the 5.000 learning steps in the L2, the network is exposed to the L1 again. Some interesting and important observations are worth presenting here. Before exposing the network to L1 sound-meaning pairs, the production of the L1 words has changed to L2 production on the AudF layer, although the AudF nodes are less activated than in the L2 (see figure 7.12 in comparison to figure 7.10b). Note that the SF nodes are still activated according to the L1 production. This changes after presenting the network with L1 sound-meaning pairs again.

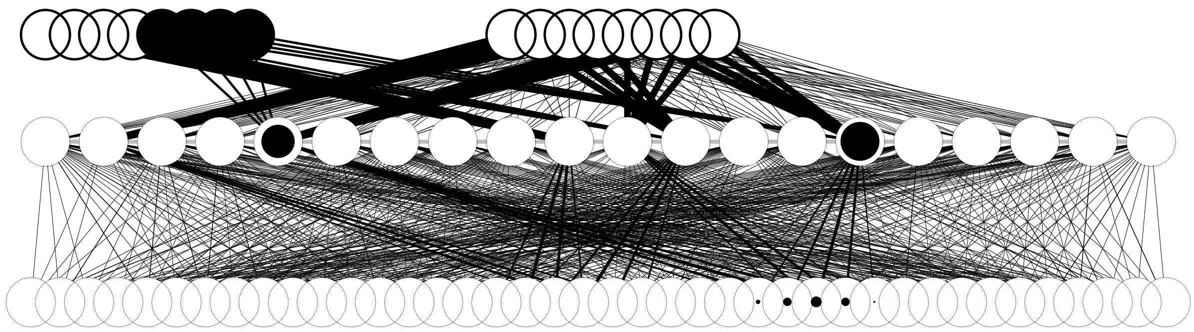


Figure 7.12 – Production first category L1

In the current examples the nodes that were activated on the SF layer for the first and second category in the L1 were different than the SF nodes that were activated for the first and the second category in the L2. This does not apply to all networks. In a very few cases exactly the same SF nodes are activated for this first category in the L1 and the first category in the L2 on the one hand and the second category in the L1 and the second category in the L2 on the other hand.

After training the network on the L1 again, for the perception the nodes that are activated on the SF layer follow this pattern: L1 – L1 L2 – L2 – L1 – L1 L2 – L2. This means that by moving over the AudF first the L1 SF nodes are activated, then a combination of the L1 and L2, then the L2 SF nodes, followed by the L1, the combination of L1 L2 and finally the L2. So, now there are boundary shifts, but within one language environment! Figures 7.13a-f illustrate this.

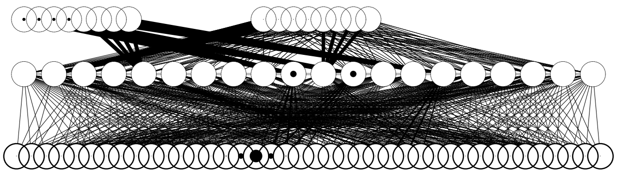


Figure 7.13a – L1: SF nodes 10 and 12

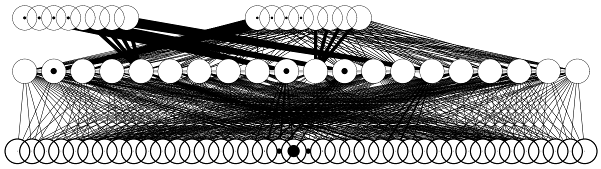


Figure 7.13b – L1 L2: SF nodes 2, 10 and 12

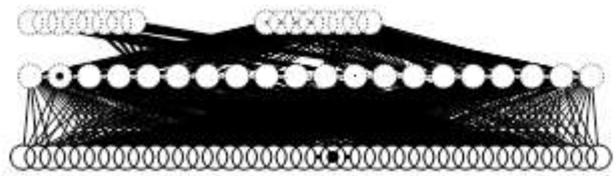


Figure 7.13c – L2: SF nodes 2 (and 12)

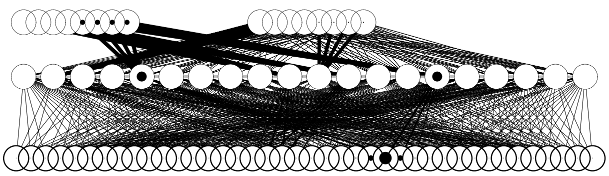


Figure 7.13d – L1: SF nodes 5 and 15

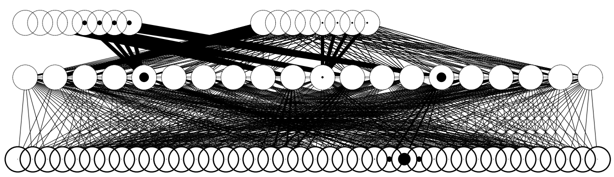


Figure 7.13e – L1 L2: SF nodes 5, (11) and 15

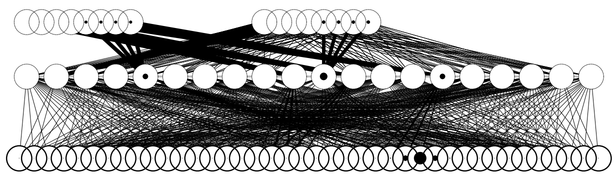


Figure 7.13f – L2: SF nodes 5, 11 and 15

The activation pattern is exactly following the distribution that is presented in chapter six, and copied here for matters of clarification:

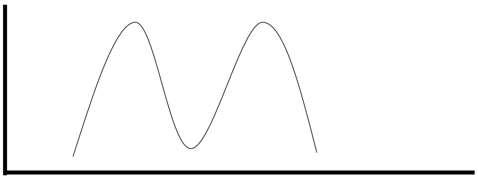


Figure 6.4a – The distribution in the L1

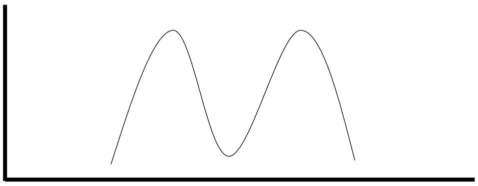


Figure 6.4b – The distribution in the L2

This distribution stays like this, even if the network has been trained on the L1 for many steps. The different lexicons are activated according to the same distribution.

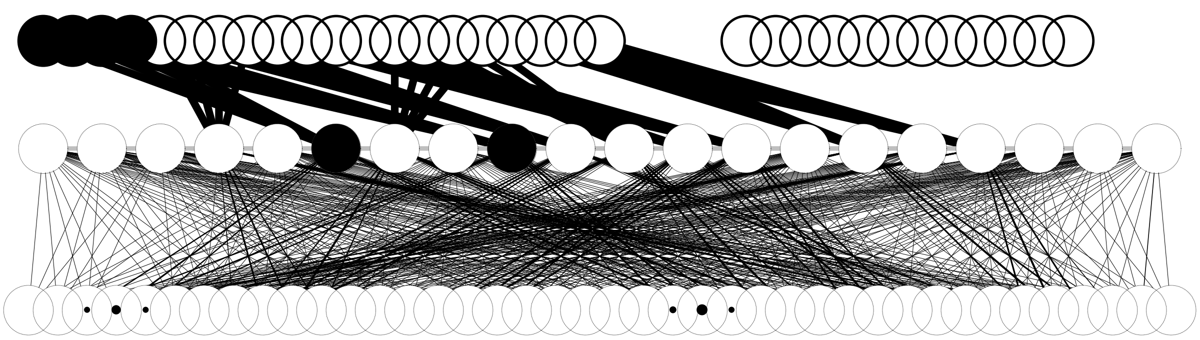
Now a closer look is taken at the networks that are trained on the first language for 10.000 learning steps, before they are presented with their second language. The results are very comparable to the results described above. However, there is one very important difference considering the boundary shift. The networks that learn their first language in 50.000 learning steps did not show a boundary shift for the second category (see figure 7.11a-b). However, the networks that were taught their first language in 10.000 steps did show this boundary shift. Next to that this boundary shift was more 'direct' than the boundary shift for the other networks: the networks that learned their L1 in 10.000 learning steps classified a sound either as category 1 or as category 2, without any area on the AudF that was not classified as any category (see figure 7.11a).

7.3 The subset scenario

The results that are obtained for the subset scenario are very comparable to the results obtained in the new scenario. Just as for the new scenario only the results for the networks that have learned their first language in 50.000 steps are presented, as these results are very comparable to the results for the networks that acquired their first language in 10.000 steps. Besides, as the results of the subset scenario are very comparable to the results of the new scenario, the figures in this section only contain networks in the production phase.

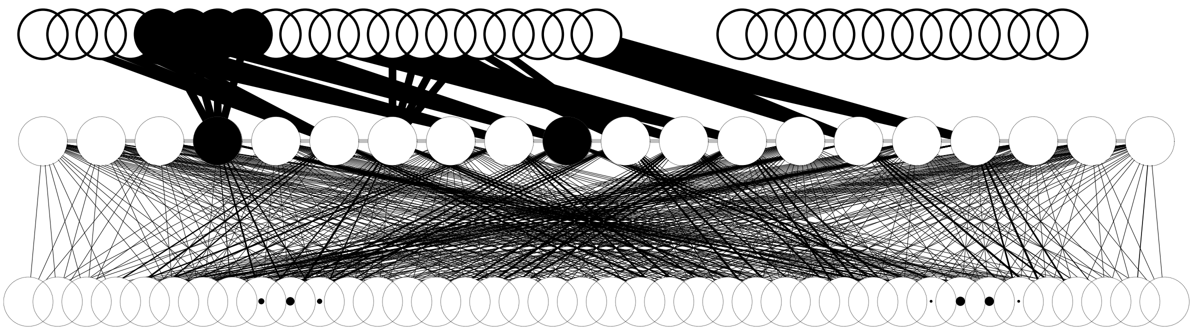
7.3.1 The production of the network for the subset scenario

After training the network on the L1 for 50.000 steps, the network has successfully learned its first language. Every sound-meaning pair receives its own distribution on the SF layer. In all cases a sound-meaning pair belongs to two or three nodes on the SF layer. In most cases different SF nodes are activated for different sound-meaning pairs. Figures 7.14a-e show images of the activation of one particular network.



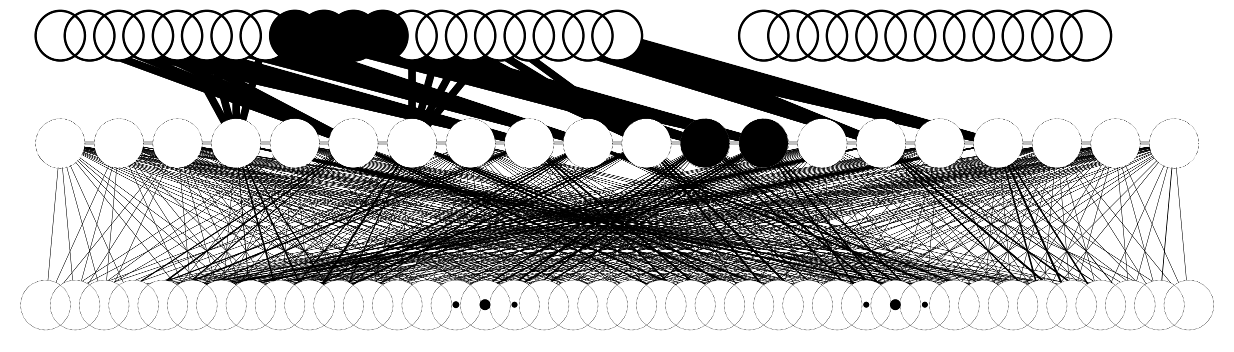
Observations: Two SF nodes are activated: 6 and 9.

Figure 7.14a – Production of the 'U' in the L1



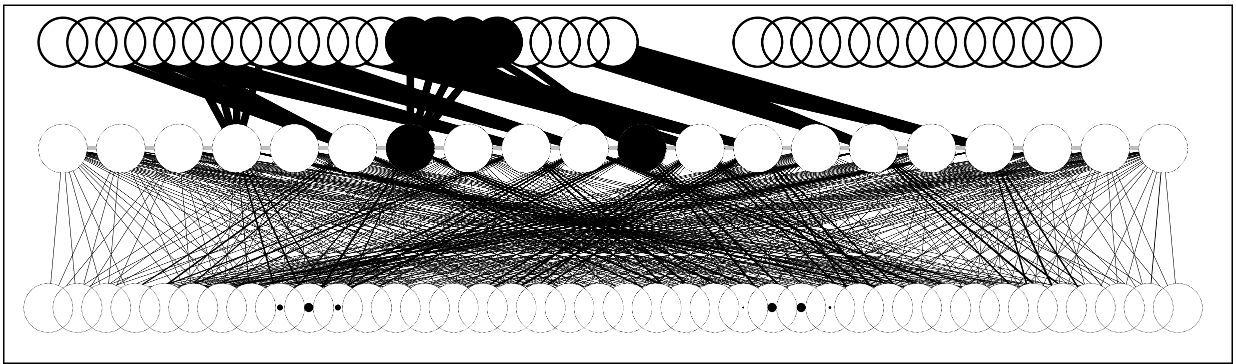
Observations: Two SF nodes are activated: 4 and 10. Four AudF nodes are activated for the F2. This is comparable to the production of the 'O' in the new scenario. In the new scenario the production of the 'O' was acquired for the second language.

Figure 7.14b – Production of the 'O' in the L1



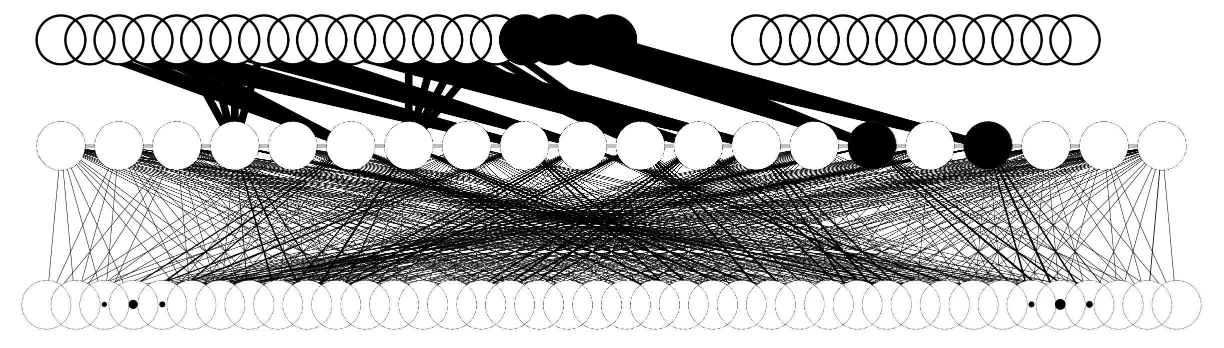
Observations: Two SF nodes are activated: 12 and 13.

Figure 7.14c – Production of the 'A' in the L1



Observations: Two SF nodes are activated: 7 and 11. Four AudF nodes are activated for the F2. This is comparable to the production of the 'E' in the new scenario. In the new scenario the production of the 'E' was acquired for the second language.

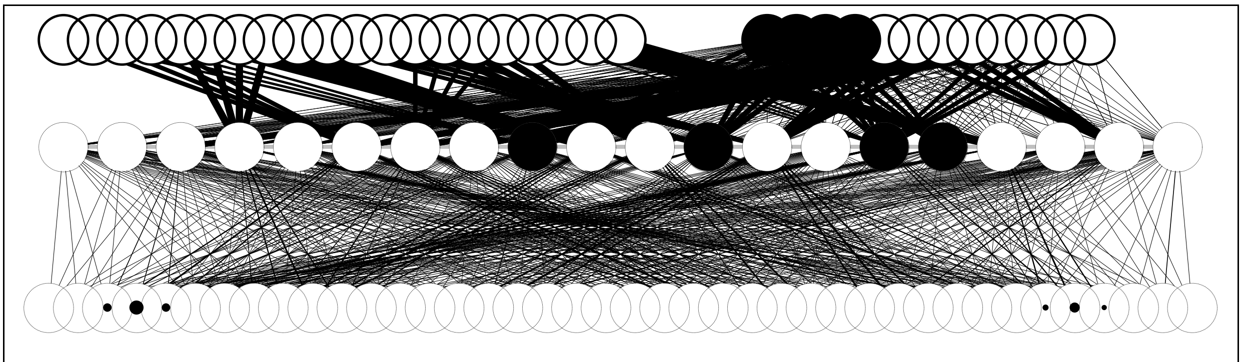
Figure 7.14d – Production of the 'E' in the L1



Observations: Two SF nodes are activated: 15 and 17.

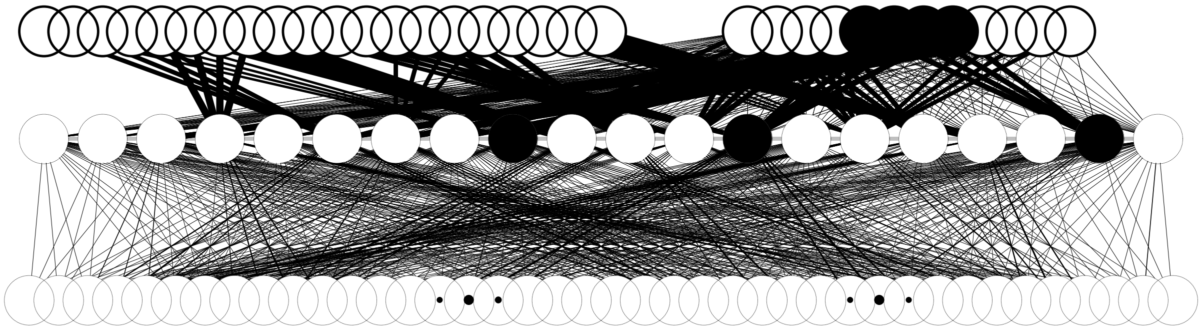
Figure 7.14e – Production of the 'I' in the L1

After this, the network is exposed to the second language for the first time. The second language contains two vowels less than the network's first language. Although the second language does not contain any sounds that do not exist in the first language, the SF nodes are not necessarily copied. Figures 7.15a-c show the productions of the network for the second language.



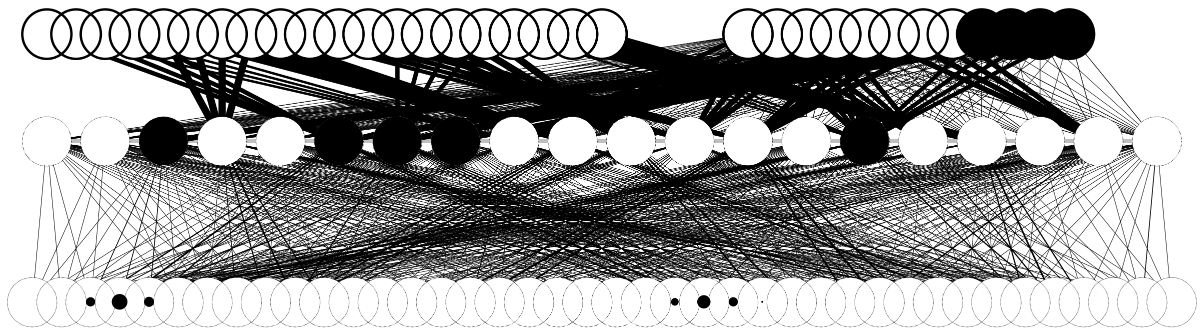
Observations: Four SF nodes are activated: 9, 12, 15 and 16. Node 15 is activated for the 'I' in the L1 as well.

Figure 7.15a – Production of the 'I_2' in the L2



Observations: Three SF nodes are activated: 9, 13 and 19. Node 9 is also activated for the 'I_2'. Node 13 is also activated for the 'A' in the L1.

Figure 7.15b – Production of the 'A_2' in the L2



Observations: Five SF nodes are activated: 3, 6, 7, 8 and 15. It is very exceptional that this many nodes are activated. Node 15 is also activated for the 'I_2'. Node 6 and node 9 are also activated for the 'U' in the L1.
Four AudF nodes are activated for the F2. After more training this changes to three nodes.

Figure 7.15c – Production of the 'U_2' in the L2

In figures 7.15a-c, for the sound categories that the L1 and the L2 have in common, more nodes are activated at the SF in the L2 than in the L1. However, in a few instances the SF nodes that are activated for these sound categories in the L2 are the same as for these sound categories in the L1. Besides, in most cases only one additional SF node is activated for a sound category in the L2 than for this sound category in the L1, in contradiction to what can be seen in the example presented in this section. However, in all cases at least one SF node is shared between sounds that occur in the first and in the second language (see chapter 8 for a detailed explanation on why this happens).

In the previous two scenarios, production immediately switched back after changing the language environment. In the majority of cases, this happened in the subset scenario as well. However, sometimes one SF node was left out of the production after shifting between the language environments. In chapter 8 it is explained why this happens.

7.3.2 The perception of the network for the subset scenario

Considering the perception not such a clear distinction between the first and the second language can be made. After training the model on the second language for 5.000 times, the networks perceived sounds in an L2 manner, even if the networks have been trained on the first language again, i.e. the networks forget their first language (but only for perception!).

Double activation in one lexicon layer is observed for this scenario as well. In some cases the 'U' and the 'I', which have their F1 in common, are both activated if one of the two sounds is presented to the network.

The networks perceived L1 sounds that are not present in the L2 as L2 sounds that are nearest to the sounds that do not exist in the L2. For example, the /e/ is perceived as /u/. The F2 of these two sounds is comparable. For the /e/, both the SF layer and the lexicon layer were activated in the same way as for the /u/ in the L2.

8. DISCUSSION

In the previous chapter the results for the different networks were presented. In this chapter the results are discussed. In order to be able to draw a conclusion it is important to compare the bilingual results with monolingual results. This is done in this chapter as well.

8.1 The new scenario

In the new scenario the bilingual's first language contains three sound-meaning pairs, whereas her second language consists of five sound-meaning pairs. In order to be able to compare the bilingual's second language situation with a monolingual situation, the results for the L2 in the new scenario can be compared with the results for the L1 in the subset scenario, as these languages are the same. For every network in the new scenario the total amount of SF nodes that are activated for the L2 categories is calculated and for every network in the subset scenario the total amount of SF nodes that are activated for the L1 categories is calculated. The comparison of the two scenarios reveals a significant difference between the language in the new scenario and the language in the subset scenario. For the subset scenario significantly fewer SF nodes are activated than for the new scenario ($U(12) = 1.0$, $p < .05$), i.e. for the same language fewer SF nodes are activated when this language is learned as a first language than when this language is learned as a second language. For the bilingual speaker this could mean that the speaker is able to link a specific L2 sound to a specific L2 word in the lexicon, but that the bilingual speaker has a different phonological system for the same language than a monolingual speaker. However, see the next section for an explanation on why certain SF nodes are activated.

In the remaining part of this section a closer look is taken at the production and the perception of the networks.

8.1.1 Production

For the discussion of the production of the new scenario it is important to look at the SF layer and at the AudF layer of the networks, as the nodes on these layers are unclamped. The results in chapter 7 showed a clear difference between the SF nodes that are activated for the sound categories in the L1 and the SF nodes that are activated for the sound categories in the L2. Very likely this is inherent to the layout of the networks. Figures 8.1a-d show the stages of the L1 and the L2 acquisition and in this figure it is made clear why different SF nodes are activated for the same sounds in the L1 and in the L2. Figure 8.1a shows the acquisition of a sound-meaning pair in the first language. One can see that two SF nodes are activated for this sound-meaning pair and that certain excitatory connections are strengthened. Of course, in the real neural networks more than four connections are present, but this is only a very schematic drawing. In figure 8.1b the network has just started acquiring its second language. One can see that the network has to learn an L2 sound-meaning pair that contains the same sound as the network already acquired for its L1. Because of that, the nodes on the AudF layer for that particular sound spread their activity to the SF via the already existing connections. Figure 8.1c shows that, because of this, also the connections between the lexicon layer and these SF sounds strengthen. On the other hand, because the L1 lexicon is not activated anymore, the connections between the L1 and the SF weaken (see chapter 6). Figure 8.1d shows that the activation of the L2 lexicon layer causes the activation of another SF node. Now one can imagine why the network is able to switch between the production of the L1 and the L2 without any training: after exposing the network to the L2 the connections between the L1 lexicon and the SF are weaker than before the L2 acquisition, but they have not disappeared. By presenting the network with L1 sound-meaning pairs, these connections strengthen again. Sometimes an SF node that is activated in the L1 surroundings, is not activated in the L2 surroundings. A possible explanation would be that the L1 lexicon causes most of the activation of this SF node. As the L1 lexicon is not activated in the L2 surroundings, it cannot activate this SF node anymore.

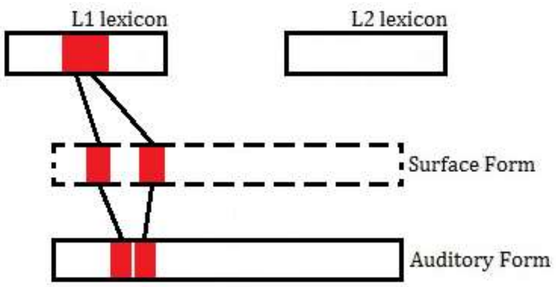


Figure 8.1a – The L1 acquisition

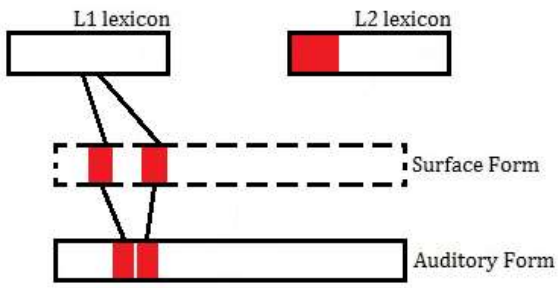


Figure 8.1b – Start of the L2 acquisition

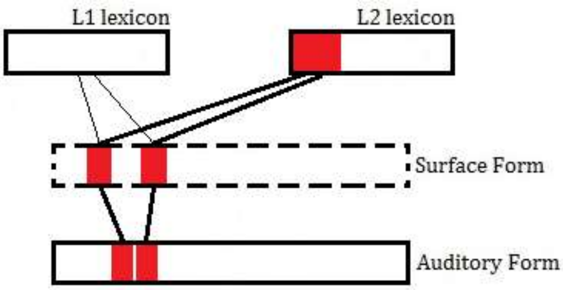


Figure 8.1c – Connections between SF and L1 lexicon weaken and connections between SF and L2 lexicon strengthen

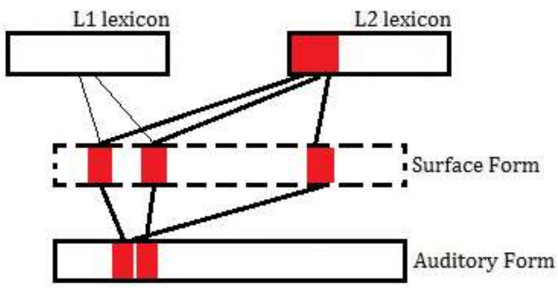


Figure 8.1d – Extra node activated on SF

One may argue that separate lexicon layers are affecting the results too much. However, for reasons that are explained in chapter 5 I have chosen to model the networks with two separate lexicon layers. In chapter 9 ideas for further research are presented and the lexicon layer is discussed again.

Now a closer look is taken at the activation of the AudF. The figures in chapter 7 showed that for the production of the ‘E’ and the ‘O’ more nodes were activated at the AudF than for the production of the other words. At first sight this may look like an L2 accent, as these are the two sound-meaning pairs that did not exist in the L1. However, comparing these results to the L1 of the subset scenario reveals that this is the case for the ‘E’ and the ‘O’ in the subset scenario as well.

Next to that one may expect that other nodes are activated at the AudF during the production of the L2 in the new scenario than during the production of the L1 in the subset scenario, as that would mean the bilingual speaker has an accent in her second language. However, exactly the same AudF nodes were activated. Only in some sporadic cases the AudF in the new scenario was activated at other points than expected (recall figure 7.2d), but this changed after more training. This shows the importance of the amount of input.

The results of the production in the new scenario show that the network has two separate phonological systems for its two languages. However, the layout of the lexicon layer plays an important role in this outcome. Next to that the amount of input is very important for which nodes are exactly activated on the AudF layer.

8.1.2 Perception

As was stated in the previous chapter, the networks perceive sounds differently in the L1 languages mode than in the L2 language mode. However, the amount of input is important. If the networks have had no, or not enough input in one of their two languages, they perceive the sounds as they would do in the other language. This is caused by the same phenomenon as is described in figures 8.1a-d. By training a network on the one language, the connections between the AudF and the SF nodes for this language strengthen, whereas the connections between the AudF and the SF nodes for the other languages weaken.

Sounds that are not present in the first language, but are part of the second language, are either perceived as L1 words or as L2 words. This depends on the formants of the sounds. If one of the formants of the new L2 sounds is near to the formant of a known L1 sound, the L2 sound is perceived as this L1 sound. On the other hand, if none of the two formants of the new L2 sounds is near to the formants of a known L1 sound, the L2 sound is perceived in an L2 manner (see chapter 7). It is important to realise that this does not happen in the L2 surroundings, i.e. in the L2 surroundings the network perceives the new L2 sounds as separate sounds. Due to this one cannot argue that assimilation has taken place. Dissimilation does not take place either, as the activation of the SF layer for the L1 does not change after acquiring the second language. Instead, also the results for perception lead to the conclusion that the network has acquired two separate phonological systems: one for the first language and one for the second language. However, it is to be remarked that also for perception the choice for two separate lexicon layers plays an important role.

Another important observation is the activation of both lexicon layers. As could be expected both the lexicons are activated if a sound is perceived that exists in both languages. However, if the network is more thoroughly trained on the one language, the lexicon in the other language is less activated, because the connections between this lexicon layer and the SF layer weaken (see chapter 6). So, also here the input plays an important role. Sometimes other nodes than the expected nodes are slightly activated in the other lexicon as well. During the training and testing phase this has happened thirteen times. In nine occasions there were no SF nodes in common. This leads to the conclusion that apparently the connections between the SF nodes and the activated lexicon nodes do weight a bit more than the other connections. However, as these SF nodes are not activated in the other language, they do not normally activate the lexicon layer in this language.

8.1.3 General discussion

The networks for the new scenario do not reveal many results that could be explained by Flege's notions of assimilation and dissimilation. For assimilation a new L2 category should have merged with an already existing L1 category. This did not happen. One explanation could be that the new categories are not near enough to an already existing L1 category. In this case dissimilation should be observed. However, dissimilation did not play a role either, as the activation of the L1 SF nodes did not change after the acquisition of the L2 (only the weight of certain connections between the L1 lexicon and the SF increased, but see figures 8.1a-d).

On the other hand, it *is* true that the network had one single bilingual system during some parts of the training. For example, after exposing the network to the L2, the network needed some training on the L1 before it perceived sounds in an L1 manner again. Then there is the observation of the activation of more nodes on the AudF layer directly after changing the language environment. Also this changed after receiving more input. In general it can be said that both these observations should be interpreted as switching to the right language mode rather than as a merged phonological system or as a permanent accent due to assimilation or dissimilation.

8.2 The similar scenario

In this section the results that are obtained for the similar scenario are discussed. The results of the similar scenario are compared to another test, in which the networks had to learn the second language of the similar scenario as their first language. For every network in the similar scenario the total amount of SF nodes that are activated for the L2 categories has been calculated and for every network in the new test the total amount of SF nodes that are activated for the L1 categories has been calculated. In contradiction to the results that were obtained for the new scenario no significant differences in the activation of the SF nodes could be observed ($t(12) = 1.7$, $p = 0.12$). It has to be remarked that this may have to do with the amount of data, as in the similar scenario fewer phonological categories have to be acquired than in the new scenario. Purely looking at the means a difference *can* be observed. The mean amount of SF nodes that is activated for the language as a first language is 4.86, whereas the mean amount of SF nodes that is activated for this language as a second language is 6.14. If this difference is significant when more data are taken into account, the use of separate lexicon layers can be an explanation for this difference.

In the remaining part of this section the production of the network for the similar scenario is being looked at first, after which the perception of the network is discussed.

8.2.1 Production

For the production of the network a clear boundary shift could be observed. Again this can be explained by the choice for two separate lexicon layers (see figure 8.1a-d). The fact that an overlap can be observed between the activated SF nodes for the first category in the L1 and the activated SF nodes for the first category in the L2, and between the activated SF nodes of the second category in the L1 and the activated SF nodes of the second category in the L2, shows that the sounds are very comparable: otherwise different SF nodes had been activated.

Another interesting observation is that, after training the network on the second language, the L1 was produced according to the boundary shift on the AudF layer, and besides that more nodes were activated than before teaching the model the second language. This would mean that the network has an L2 accent in its first language. However, after training the network on the L1 again this accent disappeared. Again this stresses the importance of the input. Next to that the fact that the L2 accent disappears after training the network on the L1 again is not in favour of Flege's SLM (Flege, 2005). Instead it shows that it is rather a matter of changing back to the right language mode (see chapter 2). Overall, production does reveal two separate phonological systems. Again it is to be remarked that this is caused by the choice for two separate lexicon layers (see previous section and chapter 9).

8.2.2 Perception

The results in chapter 7 already revealed the importance of the input, also for the similar scenario, by showing that the boundary shift for perception was different in the situation in which the network was trained only 10.000 times on the L1, but not when the network was trained on the L1 for 50.000 times.

The results also showed that the input was very important for the end state of the SF layer of the network. In neither theories (separate systems or a merged system) the observed L1 – L1 L2 – L2 – L1 – L1 L2 – L2 distribution is expected. One could argue that the speaker has one phonological system, because the way the SF nodes are activated does not change within the different language environments. On the other hand one could argue as well that the speaker has two different phonological systems, because the observed distribution is not the distribution one would expect if either assimilation or dissimilation had taken place. For assimilation the end state should contain two categories, whereas for dissimilation the end state should reveal four categories. The distribution may be nearer to what one would expect for dissimilation than to what one would expect for assimilation, but still the L1 L2 combination cannot be explained.

This L1 – L1 L2 – L2 – L1 – L1 L2 – L2 activation pattern is likely to occur if the new and old categories lay near to each other and only differ in one feature. I.e. such an activation pattern was not likely to occur for the new scenario, as the new vowel categories in the new scenario are more distinct from the already known vowel categories than the new first category from the already known first category in the similar scenario. Next to that the vowels differ from each other on the first and/or the second formant, whereas the categories in the similar scenario only differ from each other on the VOT. See chapter 9 for future research on the L1 – L1 L2 – L2 – L1 – L1 L2 – L2 activation pattern.

8.3 The subset scenario

In this section the results of the subset scenario are discussed. Again this is done by looking at production and perception separately, but only after a general remark considering the activation of the SF nodes.

The activation of the SF nodes of the second language in the subset scenario has been compared to the activation of the SF nodes of the first language in the new scenario, as these are the same languages. For every network in the subset scenario the total amount of SF nodes that are activated for the L2 categories has been calculated and for every network in the new scenario the total number of SF nodes that are activated for the L1 categories has been calculated. This showed that more SF nodes are activated (both for perception and for production) for the language if it is the network's second language than if it is the network's second language ($U(13) = 0.0$, $p < .05$). This could mean that a speaker's phonological representation is different for the same language depending on whether the language is acquired as a first or as a second language. However, note that the separate lexicon layers seem to play an important role again.

8.3.1 Production

Again the network acquires two separate phonological systems for the two languages during the production. Again this can be explained by the choice for two separate lexicon layers (see section 8.1).

8.3.2 Perception

Contradictory to the previous two scenarios the networks for the subset scenario were not able to change back to their L1 system anymore after having been exposed to their second language. (But note: this was only the case for perception.) After training on the L2 the networks had acquired a different phonological systems, which stayed like that independently from the language environment. Very likely this is caused by the amount of input. Presumably the networks have not received enough L1 input in order to be able to keep their ability to perceive their first language in an L1 manner.

Sounds that do exist in the L1 of the networks but do not exist in the L2 of the networks are perceived as L2 sounds in the L2. So, the networks perceive the same sounds differently depending on the language environment. Because the networks were not able to switch back to their first language anymore, it is not possible to say whether this means that the network has developed two separate phonological systems.

8.3.3 General discussion

Because the networks forget the perception of their first language, it is difficult to say whether the results are in favour of Flege's SLM or rather in favour of the separate systems theory. The fact that different SF nodes are activated for production may lead to the assumption that the network has two separate systems. However, this is caused by the choice for separate lexicon layers.

PART IV: CONCLUSIONS AND FURTHER RESEARCH

9. CONCLUSIONS AND FURTHER RESEARCH

In the previous chapter the results of this thesis are discussed. Based on these results the conclusion can be drawn that the amount of input plays a very important role in acquiring a second language. The comparisons between the SF nodes that were activated for the L1 and the L2 showed that the phonological system for the second language was often (for the new and the subset scenario, but not for the similar scenario) different than the phonological system for this same language but then acquired as a first language. For the production all the models acquired separate phonological systems. Already in the previous chapter is stated that this is inherent to the layout of the model. This does not necessarily mean that the results are biologically not plausible. If bilingual speakers have separate lexicons that behave like the lexicons that are modelled in the networks in this study, the results *are* biologically plausible. However, the question how the lexicon(s) of bilingual speakers should be modelled is a research question on its own.

The results for perception were less clear-cut. In the new scenario the network developed separate systems for perception as well, provided that the network had received enough language input. The similar network showed a combination of separate phonological systems and a combined phonological system by learning an L1 – L1 L2 – L2 – L1 – L1 L2 – L2 distribution for the SF layer. The subset scenario did not shift back to the first language situation after learning a second language. However, presumably this is caused by a lack of L1 input. Of course the networks that are used here are only a simplification of reality. For this reason ideas for further research are discussed here.

9.1 Further research

The ideas for further research are divided into three sections. In the first section is focussed on changes that can be made considering the input to the networks. The next section discusses changes to the layers of the network and finally additional changes are taken into account.

9.1.1 Changes to the layers

Although I have chosen to model two separate lexicons, according to current psycholinguistic research (see chapter 5), a new network could be modelled with one lexicon layer. One could also try to do the same experiments, but then without a lexicon layer. Next to that one could examine a situation in which the nodes of the lexicon of the one language are unclamped during the acquisition of the other language, e.g. the nodes of the L1 lexicon layer would be unclamped during the acquisition of the L2. These three adaptations will reveal more insight in what happens if the separate lexicon layers do not influence the activity in the network.

Next to that it would be good to experiment with connecting the lexicon layer to the SF layer only later during the language acquisition. This would model the acquisition of a new language better, as only after a while the new sounds are connected to words in the lexicon.

Besides this a concept layer should be added to the network. As I explained earlier in this thesis I decided to leave the concept layer out of the network, because it interfered with the lexicon layer. However, of course a network with a concept layer is a better representation of reality.

9.1.2 Changes to the input

It would be interesting to experiment with the amount of input. Especially for the subset scenario the amount of input could possibly change the inability to change back to the L1 surroundings. For the similar scenario it would be interesting to explore how much input is needed to obtain results comparable to the results that have been obtained for the networks that had acquired their L1 in 10.000 learning steps, and how much input is needed to obtain results comparable to the results that have been obtained for the networks that had acquired their L2 in 50.000 learning steps.

Additionally if would be interesting to look at what happens if the difference between the sound categories is larger or smaller. For the similar scenario a larger difference between the category boundaries could reveal another activation pattern at the SF than the L1 – L1 L2 – L2 – L1 – L1 L2 – L2 pattern. For the other scenarios it would be interesting to see what results are obtained by teaching the network different vowels, or changing the composition of sounds in the first and in the second language. E.g. in this study, in the new scenario, the vowels that were to be learned in addition, were quite distinct from each other. In chapter 3 the example was given of a Spanish L1 speaker that had to learn English as her L2 and therefore had to learn the difference between the /i/ and the /ɪ/, a distinction that is not known in Spanish. This difference may be more difficult to acquire, as the difference may be smaller than the distinction between, for example, the /i/ and the /e/. A next study could look at these smaller differences.

9.1.3 Additional changes

Of course, also linking single sounds to words in the lexicon is not a very good representation of reality. However, connecting several sounds in order to make words is not possible with the current state of the neural networks. It would be a very interesting research.

As already stated in chapter 5 of this thesis it could be possible that the speaker is affected by sensorimotor constraints. For now I have not included this option in the model, but this would be an interesting addition. It could result in a better explanation for the bilingual accent. Seinhorst (2012) uses an articulary node in his neural networks. This accounts for the fact that certain sounds are easier to pronounce than other sounds. However, more articulatory nodes with inhibitory connections to the Auditory Form would be an option to model sensorimotor constraints.

Next to sensorimotor constraints, also a decrease of brain plasticity may influence the ability to learn a second language. In order to investigate the increase of brain plasticity one needs to decide on how brain plasticity declines over time. Figures 9.1a-c give examples of graphs that can be followed by modelling declining brain plasticity.

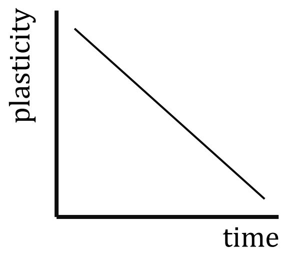


Figure 9.1a –
Linear decrease

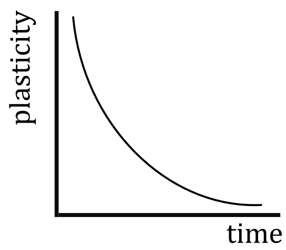


Figure 9.1b –
Exponential decrease

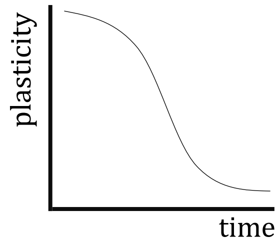


Figure 9.1c –
Sigmoid function

The current neural networks are not able to switch quickly between language environments. In order to have an optimal language perception, the networks need quite some learning steps in the target language. In reality bilingual speakers are able to switch easier and faster between two language environments. In future networks one or more parameters that help the networks with this switch could be included.

In general can be said that many adaptations to the neural networks will increase their biological validity. Of course, the ultimate goal is to develop a network that exactly models the human brain. A computational neural network of the human brain would be extremely valuable for research in many scientific fields.

REFERENCES

Benders, T. (2013). *Nature's distributional-learning experiment. Infants' input. Infants' perception. Computational Modelling.* PhD dissertation, University of Amsterdam.

Birdsong, D. (2009). Age and the end state of second language acquisition. In: W.C. Ritchie and T.K. Bhatia (eds.), *The New Handbook of Second Language Acquisition,* 401-424, United Kingdom: Emerald Group Publishing Limited.

Birdsong, D. (2005). Interpreting age effects in second language acquisition. In: J.F. Kroll & A.M.B. de Groot (Eds.) *Handbook of bilingualism: Psycholinguistic approaches,* pp. 109-127. New York: Oxford University Press.

Birdsong, D. (2006). Age and second language acquisition and processing. A selective overview. *Language Learning,* 56:9-48.

Birdsong, D. & Molis, M. (2001). On the evidence for maturational constraints in second-language acquisition. *Journal of Memory of Language,* 44:235-249.

Boersma, P. (2007a). *Some listener-oriented accounts of h-aspiré in French*. Lingua 117:1989–2054.

Boersma, P., Benders, T. and Seinhorst, K. (fc.). Neural network models for phonology and phonetics.

Boersma, P. and Weenink, D. (2013). *Praat: doing phonetics by computer.* http://www.praat.org.

Boomershine, A. (2013). The Perception of English Vowels by Monolingual, Bilingual, and Heritage Speakers of Spanish and English. *Selected Proceedings of the 15th Hispanic Linguistics Synposium,* ed. Chad Howe et al, 103-118. Somerville, MA: Cascadilla Proceedings Project.

Byrne, J.H. (2014). *Introduction to Neurons and Neuronal Networks.* Online source. Accessed 19th of May, 2014. Available at: http://neuroscience.uth.tmc.edu/s1/introduction.html.

Chládková, K. (2014). *Finding phonological features in perception.* PhD dissertation, University of Amsterdam.

Cotman, C.W. and Berchtold, N.C. (2002). Exercise: a behavioural intervention to enhance brain health and plasticity. *Trends in Neurosciences*, Vol. 25, no. 6;295-301.

De Groot, A.M.B. (2011). *Language and cognition in bilinguals and multilinguals.* Psychology Press: New York.

DeKeyser, R.M. (2000). The robustness of critical period effects in second language acquisition. *Studies in Second Language Acquisition,* 22:499-433.

Dijkstra, T., Grainger, J., & Van Heuven, W.J.B. (1999). Recognition of cognates and interlingual homographs: The neglected role of phonology. *Journal of Memory and Language,* 41:496-518.

Duñabeitiaa, J.A., Dimitropouloua, M., Uribe-Etxebarriac, O. et al. (2010). Electrophysiological correlates of the masked translation priming effect with highly proficient simultaneous bilinguals. *Brain research,* 1359: 142-154.

Escudero, P. (2005). *Linguistic Perception and Second Language Acquisition. Explaining the attainment of optimal phonological categorization.* PhD dissertation, University of Utrecht.

Escudero, P. and Boersma, P. (2002). The Subset Problem in L2 Perceptual Development: Multiple-Category Assimilation by Dutch Learners of Spanish. Available at: http://www.fon.hum.uva.nl/paul/p2b/papers/bu26.pdf

Finkbeiner, M., Forster, K.I., Nicol, J., and Nakamura, K. (2004). The role of polysemy in masked semantic and translation priming. *Journal of Memory and Language*, 51:47-65.

Flege J.E. (1987). The production of "new" and "similar" phones in a foreign language: Evidence for the effect of equivalence classification. *Journal of Phonetics*, 15:47-65.

Flege, J.E. (1995). Second-language speech learning: theory, findings, and problems. Strange, W. (eds.), *Speech Perception and Linguistic Experience*. Timonium, 233-277, MD; York Press, Flege, J.E. (2002). Interactions between the native and Second-language phonetic systems. In: P. Burmeister, T. Piske and A. Rohde (Eds.). *An Integrated View of Language Development: Papers in Honor of Henning Wode*, 217-244. Trier: Wissenschaftlicher Verlag.

Flege, J.E. (2003). Assessing constraints on second-language segmental production and perception. In: A. Meyer and N. Schiller (Eds.). *Phonetics and Phonology in Language Comprehension and Production, Differences and Similarities.* 319-355, Berlin: Mouton de Gruyter.

Flege, J.E., Bohn, O.S. and Jang, S. (1997). Effects of experience on non-native speakers' production and perception of English vowels. *Journal of Phonetics*, 25:437-470.

Flege, J.E., Schirru, C. and MacKay, I.R.A. (2003). Interaction between the native and second language phonetic subsystems. *Speech Communication*, 40:467-491.

Fox, R.A., Flege, J.E. and Munro, M.J. (1995). The perception of English and Spanish vowels by native English and Spanish listeners: A multidimensional scaling analysis. *J. Acoust. Soc. Am.*, 97(4):2540-2551.

Garcia-Sierra, A., Diehl, R.L. and Champlin, C. (2009). Testing the double phonemic boundary in bilinguals. *Speech communication*, 51:369-378.

Grosjean, F. (1988). Exploring the recognition of guest words in bilingual speech. *Language and Cognitive Processes,* Vol. 3, no. 3;233-274.

Haigh, C.A. and Jared, D. (2007). The activation of phonological representation by bilinguals while reading silently: Evidence from interlingual homophones. *Journal of Experimental Psychology: Learning, Memory and Cognition,* 33:623-644.

Hebb, D. (1949). *The Organization of Behavior.* New York: Wiley & Sons.

Hyltenstam K. & N. Abrahamsson (2003). Maturational constraints in SLA. In C. Doughty & M. Long (eds.), *Handbook of Second Language Acquisition*, 104-129. Cambridge, MA: Blackwell.

Kirsner, K., Lalor, E., and Hird, K. (1993). The Bilingual Lexicon. Exercise, Meaning and Morphology. In R. Schreuder and B. Weltens (eds.), *The bilingual lexicon,* p. 215-248. Amsterdam/Philadelphia: John Benjamins.

Kolers, P. A. (1963). Interlingual Word Associations. *Journal of Verbal Learning and Behaviour*, 2:291-300.

Kroll, J.F., Michael, E., Tokowicz, C.M. and Dufour, R. (2002). The development of lexical fluency in a second language. *Second Language Research*, Vol. 18, nr. 2:137-171.

Ladefoged, P. (1996). *Elements of Acoustic Phonetics. Second Edition.* Chicago: The University of Chicago Press.

Ladefoged, P. (2008). *Vowels.* Online source. Accessed at 29th of May, 2014. Available at: http://www.phonetics.ucla.edu/course/chapter1/vowels.html.

Larsen, J.D., Fritsch, T. and Grava, S. (1994). A semantic priming test of bilingual language storage and the compound vs the coordinate bilingual distinction with Latvian-English bilinguals. *Perceptual and Motor Skills*, 79:459-466.

Lee, T. (2011). Grammatical knowledge of Korean heritage speakers: Early vs. late bilinguals. In: *Linguistic Approaches to Bilingualism,* 95: 149-174.

Lehtonen, M., Hultén, A., Rodríguez-Fornells, A., Cunillera, T., Tuomainen, J. et al. (2012). Differences in word recognition between early bilinguals and monolinguals: Behavioral and ERP evidence. *Neuropsychologia,* 50:1362-1371.

Lenneberg, E.H. (1967). *Biological foundation of language.* New York: John Wiley & Sons.

Ian Maddieson. (2013). Vowel Quality Inventories. In: Dryer, Matthew S. & Haspelmath, Martin (eds.) *The World Atlas of Language Structures Online.* Leipzig: Max Planck Institute for Evolutionary Anthropology. (Available online at http://wals.info/chapter/2, Accessed on 2the 27th of May, 2014.)

Jiang, N. and Forster, K.I. (2001). Cross-Language Priming Asymmetries in Lexical Decision and Episodic Recognition. *Journal of Memory and Language*, 44:32-51.

Maye, J., Werker, F. and Gerken, L. (2002). Infant sensitivity to distributional information can affect phonetic discrimination. *Cognition*, 82:B101-B111.

Paradis, M. (2004). *A Neurolinguistic Theory of Bilingualism.* Amsterdam/Philadelphia: John Benjamins Publishing Company.

Patkowski, M. (1980). The sensitive period for the acquisition of syntax in a second language. *Language Learning*, 30, 449-472.

Patkowski, M. (1990). Age and accent in a second language: A reply to James Emil Flege, *Applied Linguistics*, 11, 73-89.

Penfield, W., and Roberts, L. (1959). *Speech and brain mechanisms.* Princeton, NJ: Princeton University Press.

Piske, T., Flege, J.E., MacKay, I.R.A. and Meador, D. (2002). The production of English vowels by early and late Italian-English bilinguals, *Phonetica*, 59:49-71.

Popper, K. (1959). *The Logic of Scientific Discovery.* Hutchinson & Co.

Potter, M.C., So, K., Von Eckhardt, B. and Feldman, L.B. (1984). Lexical and Conceptual Representation in Beginning and Proficient Bilinguals. *Journal of verbal learning and verbal behaviour*, 23:23-38.

Scovel, T. (1988). *A Time to Speak. A Psycholinguistic Inquiry Into the Critical Period for Human Speech*. Cambridge, MA: Newbury House Publishers.

Seinhorst, K. (2012). *The evolution of auditory dispersion in symmetric neural nets.* M.A. thesis, University of Amsterdam.

Selinker (1972). Interlanguage. *International Review of Applied Linguistics*, 10:219-231.

Soares, C. and Grosjean, F. (1984). Bilinguals in a monolingual and bilingual speech mode: The effect on lexical access. *Memory and Cognition,* Vol.12, no. 4;380-386.

Van der Stelt, J., Koopmans-van Beinum, F. (2000). Klankperceptie en klankproductie in het eerste levensjaar. In S. Gillis & A. Schaerlaekens (eds.), *Kindertaalverwerving. Een handboek voor het Nederlands*, 93-130. Groningen, the Netherlands: Martinus Nijhoff.

Watson, R. (n.d.). *Neurons.* Online source. Accessed the 19th of May, 2014. Available at: https://www.math.ucdavis.edu/~watson/spiking/

Appendix A – Script for the new scenario

```
# Praat script for the new scenario, based on an earlier network by Boersma (2013)

form growingLexicon
	word Foreground_colour Yellow
	word Background_colour Maroon
	word Button_colour Olive
	word Font Times
	natural Font_size 35
	real number_of_vowels_l1 3
	real number_of_vowels_l2 5
endform

demo.foregroundColour$ = foreground_colour$
demo.backgroundColour$ = background_colour$
demo.buttonColour$ = button_colour$
demo.font$ = font$
demo.fontSize = font_size

#
# Properties of the network
#

spreadingRate = 0.01
actMin = 0.0
actMax = 1.0
actLeak = 1.0
learning_rate = 0.001
weightMin = -1.0
weightMax = 1.0
weightLeak = 0.0
xMin = 0.0
xMax = 10.0
yMin = 0.0
yMax = 10.0
instar = 0.5
outstar = 0.5
initialWeightMin = 0.0
initialWeightMax = 0.1
inhibitionAtSF = -0.5
inhibitionAtPF = -0.5
inhibitionAtAudF = -0.35
artConnStrength = 1.5
weightOfCentrArtConn = -0.25

pfL1.numberOfNodes = 4 * number_of_vowels_l1
pfL2.numberOfNodes = 4 * number_of_vowels_l2
sf.numberOfNodes = 20
audf.numberOfNodes = 40
artf.numberOfNodes = 0

pf.y = 7.0
sf.y = 5.0
audf.y = 2.0
artf.y = 0.5

artf.offsetNode = 0
audf.offsetNode = artf.offsetNode+artf.numberOfNodes
sf.offsetNode = audf.offsetNode+audf.numberOfNodes
```

```
pfL1.offsetNode = sf.offsetNode + sf.numberOfNodes
pfL2.offsetNode = pfL1.offsetNode + pfL1.numberOfNodes

peak_sharpness = 2
auditory_sharpness = 50
stdevOfAmbient = (audf.numberOfNodes - 1) / peak_sharpness / 10
auditory_spreading = (audf.numberOfNodes - 1) / auditory_sharpness
numberOfTimesOfActivitySpreading = 100
inputExaggeration = 0.25

f1_strength = 1
weightNormalization = 0
l2_step = 50000
use_alternations = 1

vowel1_audf1_mean = 0.10*audf.numberOfNodes
vowel1_audf2_mean = 0.90*audf.numberOfNodes
vowel2_audf1_mean = 0.40*audf.numberOfNodes
vowel2_audf2_mean = 0.75*audf.numberOfNodes
vowel3_audf1_mean = 0.10*audf.numberOfNodes
vowel3_audf2_mean = 0.60*audf.numberOfNodes
vowel4_audf1_mean = 0.25*audf.numberOfNodes
vowel4_audf2_mean = 0.8375*audf.numberOfNodes
vowel5_audf1_mean = 0.25*audf.numberOfNodes
vowel5_audf2_mean = 0.6625*audf.numberOfNodes


procedure createNetwork
	network = Create empty Network... BilingualSpeaker spreadingRate
	... linear actMin actMax actLeak learning_rate weightMin weightMax
	... weightLeak xMin xMax yMin yMax

	Set instar... instar
	Set outstar... outstar

	for i to artf.numberOfNodes
		Add node... xMin+(xMax-xMin)/artf.numberOfNodes*(i-0.5) artf.y 0 yes
	endfor

	for i to audf.numberOfNodes
		a = artConnStrength / (0.5 - 0.5 * audf.numberOfNodes ^ 2)
		b = (1 + audf.numberOfNodes) / 2
		Add connection... artf.offsetNode+1 audf.offsetNode+i weightOfCentrArtConn+a*(i-
		…b)^2 0
	endfor

	for i to audf.numberOfNodes
		Add node... xMin+(xMax-xMin)/audf.numberOfNodes*(i-0.5) audf.y 0 yes
	endfor

	for i to sf.numberOfNodes
		Add node... xMin+(xMax-xMin)/sf.numberOfNodes*(i-0.5) sf.y 0 no
	endfor

	for i to pfL1.numberOfNodes
		Add node... xMin+i/4 pf.y 0 yes
	endfor

	for i to pfL2.numberOfNodes
		Add node... xMin+4+i/4 pf.y 0 yes
```

```
	endfor

#
# Excitatory connections
#

	for i to audf.numberOfNodes
		for j to sf.numberOfNodes
			Add connection... audf.offsetNode+i sf.offsetNode+j
			…randomUniform(initialWeightMin,initialWeightMax) 1.0
		endfor
	endfor

	for i to sf.numberOfNodes
		for j to pfL1.numberOfNodes
			Add connection... sf.offsetNode+i pfL1.offsetNode+j
			…randomUniform(initialWeightMin,initialWeightMax) 1.0
		endfor
	endfor


#
# Inhibitory connections
#

	for i to artf.numberOfNodes
		for j to audf.numberOfNodes
			Add connection... artf.offsetNode+i audf.offsetNode+j
			…randomUniform(initialWeightMin,initialWeightMax) 0.0
		endfor
	endfor

	for .i to audf.numberOfNodes - 1
		for .j from .i + 1 to audf.numberOfNodes
			Add connection... audf.offsetNode+.i audf.offsetNode+.j inhibitionAtAudF 0.0
		endfor
	endfor


	for i to sf.numberOfNodes - 1
		for j from i+1 to sf.numberOfNodes
			Add connection... sf.offsetNode+i sf.offsetNode+j inhibitionAtSF 0.0
		endfor
	endfor

	for i to pfL1.numberOfNodes - 1
		for j from i+1 to pfL1.numberOfNodes
			Add connection... pfL1.offsetNode+i pfL1.offsetNode+j inhibitionAtPF 0.0
		endfor
	endfor

	for i to pfL2.numberOfNodes - 1
		for j from i+1 to pfL2.numberOfNodes
			Add connection... pfL2.offsetNode+i pfL2.offsetNode+j inhibitionAtPF 0.0
		endfor
	endfor


	inputDistribution = Create Matrix... inputDistribution 0.5 audf.numberOfNodes+0.5
	…audf.numberOfNodes 1.0 1 1 1 1 1 1 0.0
```

```
	distanceMatrix = Create Matrix... distance 0.5 audf.numberOfNodes+0.5
	...audf.numberOfNodes+1 1.0 0.5 1 1 1 1 1 0.0

endproc

label NETWORK
call createNetwork
step = 0
pfNode = 1
audNode = 1
l1 = 1
l2 = 0
repeat
	call demo.erase
	call demo.centredTitle BilingualSpeaker
	demo.textY += 13
	demo Select inner viewport... 20 80 20 80
	select network
	demo Draw... yes
	demo Text... xMin right artf.y half [[ArtF]]
	demo Text... xMin right audf.y half [[AudF]]
	demo Text... xMin right sf.y half /SF/
	demo Text... xMin right pf.y half Lexicons
	select inputDistribution
	demo Magenta
	demo Line width... 3
	for i to number_of_vowels_l2
		demo Draw rows... 0 0 i-0.5 i+0.5 0 step/3
	endfor
	demo 'demo.foregroundColour$'
	demo Line width... 2


	demo Select inner viewport... 0 100 0 100
	demo Axes... 0 100 0 100
	demo Text... 50 centre 10 half After step 'step'.
	call demo.button 88 98 50 1000↑
	call demo.button 88 98 40 100↑
	call demo.button 88 98 30 5000↑
	call demo.button 88 98 20 1↑
	call demo.button 88 98 10 new
	call demo.button 2 12 10 set...

	while demoWaitForInput ( )
		if demoInput ("a")
			select network
			Zero activities... 0 0

			if l1 = 1

				for i to pfL1.numberOfNodes
					Set clamping... pfL1.offsetNode+i no
				endfor


				for i to pfL2.numberOfNodes
					Set clamping... pfL2.offsetNode+i no
				endfor

				audNode = max (1, min (round (audNode), audf.numberOfNodes)) + 1
```

```
		if audNode > audf.numberOfNodes
			audNode = 1
		endif

		whichVowel = randomInteger (1, number_of_vowels_l2)
		vowel.f1 = vowel'whichVowel'_audf1_mean
		vowel.f2 = vowel'whichVowel'_audf2_mean

		audNode1 = randomGauss (vowel.f1, stdevOfAmbient/2)
		audNode2 = randomGauss (vowel.f2, stdevOfAmbient/2)

		for i to audf.numberOfNodes
			Set clamping... audf.offsetNode+i yes
			Set activity... audf.offsetNode+i
			... f1_strength * exp (-0.5 * (i - audNode1) ^ 2 /
			…auditory_spreading ^ 2) * inputExaggeration +
			... exp (-0.5 * (i - audNode2) ^ 2 / auditory_spreading ^ 2) *
			…inputExaggeration
		endfor

		Spread activities... numberOfTimesOfActivitySpreading*5
		goto NETWORK_NEXT

	elsif l2 = 1

		for i to pfL2.numberOfNodes
			Set clamping... pfL2.offsetNode+i no
		endfor

		for i to pfL1.numberOfNodes
			Set clamping... pfL1.offsetNode+i no
		endfor

		audNode = max (1, min (round (audNode), audf.numberOfNodes)) + 1

		if audNode > audf.numberOfNodes
			audNode = 1
		endif

		whichVowel = randomInteger (1, number_of_vowels_l2)
		vowel.f1 = vowel'whichVowel'_audf1_mean
		vowel.f2 = vowel'whichVowel'_audf2_mean

		audNode1 = randomGauss (vowel.f1, stdevOfAmbient/2)
		audNode2 = randomGauss (vowel.f2, stdevOfAmbient/2)

		for i to audf.numberOfNodes
			Set clamping... audf.offsetNode+i yes
			Set activity... audf.offsetNode+i
			... f1_strength * exp (-0.5 * (i - audNode1) ^ 2 /
			…auditory_spreading ^ 2) * inputExaggeration +
			... exp (-0.5 * (i - audNode2) ^ 2 / auditory_spreading ^ 2) *
			…inputExaggeration
		endfor

		Spread activities... numberOfTimesOfActivitySpreading*5
		goto NETWORK_NEXT
	endif
```

```
elsif demoInput ("12345")
	select network
	Zero activities... 0 0
	pfNode = index ("12345", demoKey$ ())

	if l1 = 1
		if pfNode <= number_of_vowels_l1
			for .i to audf.numberOfNodes
				Set clamping... audf.offsetNode+.i no
			endfor

			for .i to pfL1.numberOfNodes
				Set activity... pfL1.offsetNode+.i 0
				Set clamping... pfL1.offsetNode+.i yes
			endfor

			for i to pfL2.numberOfNodes
				Set clamping... pfL2.offsetNode+i yes
				Set activity... pfL2.offsetNode+i 0
			endfor

			for .i to pfL1.numberOfNodes/number_of_vowels_l1
				.k = (pfNode-
				…1)*(pfL1.numberOfNodes/number_of_vowels_l1)
				…+ .i
				Set activity... pfL1.offsetNode+.k 1
			endfor

			Spread activities... numberOfTimesOfActivitySpreading
		endif
		goto NETWORK_NEXT

	elsif l2 = 1
		if pfNode <= number_of_vowels_l2
			for .i to audf.numberOfNodes
				Set clamping... audf.offsetNode+.i no
			endfor

			for .i to pfL2.numberOfNodes
				Set activity... pfL2.offsetNode+.i 0
				Set clamping... pfL2.offsetNode+.i yes
			endfor

			for i to pfL1.numberOfNodes
				Set clamping... pfL1.offsetNode+i yes
				Set activity... pfL1.offsetNode+i 0
			endfor

			for .i to pfL2.numberOfNodes/number_of_vowels_l2
				.k = (pfNode-
				…1)*(pfL2.numberOfNodes/number_of_vowels_l2)
				…+ .i
				Set activity... pfL2.offsetNode+.k 1
			endfor

			Spread activities... numberOfTimesOfActivitySpreading
		endif
	endif
	goto NETWORK_NEXT
```

```
elsif demoClickedIn (2, 12, 10-4, 10+4) or demoInput ("z")   ; set...
	beginPause ("Settings")
		boolean ("l1", 0)
		boolean ("l2", 0)
	clicked = endPause ("Cancel", "Set", 2)

	if clicked = 2
		if l1 = 1
			vowel1_audf1_mean = 0.10*audf.numberOfNodes
			vowel1_audf2_mean = 0.90*audf.numberOfNodes
			vowel2_audf1_mean = 0.40*audf.numberOfNodes
			vowel2_audf2_mean = 0.75*audf.numberOfNodes
			vowel3_audf1_mean = 0.10*audf.numberOfNodes
			vowel3_audf2_mean = 0.60*audf.numberOfNodes
			writeInfoLine: "l1 aan"
		elsif l2 = 1
			vowel1_audf1_mean = 0.40*audf.numberOfNodes
			vowel1_audf2_mean = 0.75*audf.numberOfNodes
			vowel2_audf1_mean = 0.25*audf.numberOfNodes
			vowel2_audf2_mean = 0.6625*audf.numberOfNodes
			vowel3_audf1_mean = 0.10*audf.numberOfNodes
			vowel3_audf2_mean = 0.90*audf.numberOfNodes
			vowel4_audf1_mean = 0.25*audf.numberOfNodes
			vowel4_audf2_mean = 0.8375*audf.numberOfNodes
			vowel5_audf1_mean = 0.10*audf.numberOfNodes
			vowel5_audf2_mean = 0.60*audf.numberOfNodes
			writeInfoLine: "l2 aan"
		endif

	endif


	goto NETWORK_NEXT
endif

numberOfSteps =
... if demoClickedIn (88, 98, 20-4, 20+4) or demoInput ("↑") then 1 else
... if demoClickedIn (88, 98, 30-4, 30+4) or demoInput ("v") then 5000 else
... if demoClickedIn (88, 98, 40-4, 40+4) or demoInput ("h") then 100 else
... if demoClickedIn (88, 98, 50-4, 50+4) or demoInput ("d") then 1000 else
... if demoInput ("4") then 10000 else 0 fi fi fi fi fi

if numberOfSteps <> 0
	select network
	for ministep to abs (numberOfSteps)

		if step = l2_step
			select network

### Add Excitatory connections between the new lexicon layer and sf ###

			for .i to sf.numberOfNodes
				for .j to pfL2.numberOfNodes
					Add connection... sf.offsetNode+.i
					…pfL2.offsetNode+.j
					…randomUniform
					…(initialWeightMin,initialWeightMax) 1.0
				endfor
			endfor
```

```
		for .i to sf.numberOfNodes
			for .j to pfL2.numberOfNodes
				Add connection... sf.offsetNode+.i
				…pfL2.offsetNode+.j
				…randomUniform
				…(initialWeightMin,initialWeightMax) 1.0
			endfor
		endfor

	endif

	step += 1
	Zero activities... 0 0

#### L1 surroundings ####

	if l1 = 1
		whichVowel = randomInteger (1, number_of_vowels_l1)

		vowel.f1 = vowel'whichVowel'_audf1_mean
		vowel.f2 = vowel'whichVowel'_audf2_mean

		audNode1 = randomGauss (vowel.f1, stdevOfAmbient/2)
		audNode2 = randomGauss (vowel.f2, stdevOfAmbient/2)

		for i from 1 to audf.numberOfNodes
			Set clamping... audf.offsetNode+i yes
			Set activity... audf.offsetNode+i
			... f1_strength * exp (-0.5 * (i - audNode1) ^ 2 /
			…auditory_spreading ^ 2) * inputExaggeration +
			... exp (-0.5 * (i - audNode2) ^ 2 / auditory_spreading ^
			…2) * inputExaggeration
		endfor

		for i from 1 to pfL2.numberOfNodes
			Set clamping... pfL2.offsetNode+i yes
			Set activity... pfL2.offsetNode+i 0
		endfor

		for i from 1 to pfL1.numberOfNodes
			Set clamping... pfL1.offsetNode+i yes
			Set activity... pfL1.offsetNode+i 0
		endfor

		if whichVowel <= number_of_vowels_l1
			for i to pfL1.numberOfNodes/number_of_vowels_l1
				Set activity... pfL1.offsetNode+(whichVowel-1)
				…*(pfL1.numberOfNodes/
				…number_of_vowels_l1)+i 1
			endfor
		endif

		if whichVowel <= number_of_vowels_l1
			for i to pfL1.numberOfNodes/number_of_vowels_l1
				Set activity... pfL1.offsetNode+(whichVowel-1)
				…*(pfL1.numberOfNodes/
				…number_of_vowels_l1)+i 1
			endfor
		endif
```

```
                    Spread activities... numberOfTimesOfActivitySpreading
                    Update weights

        #### L2 surroundings ####

                elsif l2 = 1
                    whichVowel = randomInteger (1, number_of_vowels_l2)

                    vowel.f1 = vowel'whichVowel'_audf1_mean
                    vowel.f2 = vowel'whichVowel'_audf2_mean

                    audNode1 = randomGauss (vowel.f1, stdevOfAmbient/2)
                    audNode2 = randomGauss (vowel.f2, stdevOfAmbient/2)

                    for i from 1 to audf.numberOfNodes
                        Set clamping... audf.offsetNode+i yes
                        Set activity... audf.offsetNode+i
                        ... f1_strength * exp (-0.5 * (i - audNode1) ^ 2 /
                        …auditory_spreading ^ 2) * inputExaggeration +
                        ... exp (-0.5 * (i - audNode2) ^ 2 / auditory_spreading ^
                        …2) * inputExaggeration
                    endfor

                    for i from 1 to pfL1.numberOfNodes
                        Set clamping... pfL1.offsetNode+i yes
                        Set activity... pfL1.offsetNode+i 0
                    endfor

                    for i from 1 to pfL2.numberOfNodes
                        Set clamping... pfL2.offsetNode+i yes
                        Set activity... pfL2.offsetNode+i 0
                    endfor

                    if whichVowel <= number_of_vowels_l2
                        for i to pfL2.numberOfNodes/number_of_vowels_l2
                            Set activity... pfL2.offsetNode+(whichVowel-1)
                            …*(pfL2.numberOfNodes/
                            …number_of_vowels_l2)+i 1
                        endfor
                    endif

                    Spread activities... numberOfTimesOfActivitySpreading
                    Update weights

                endif

                select inputDistribution
                Formula... self + f1_strength * (exp (-0.5 * (col - audNode1) ^ 2 /
                …auditory_spreading ^ 2)) * inputExaggeration
                ... + (exp (-0.5 * (col - audNode2) ^ 2 / auditory_spreading ^ 2)) *
                …inputExaggeration
                select network

        endfor
        goto NETWORK_NEXT
    endif
    goto NETWORK_END demoInput ("h←• →")
endwhile
label NETWORK_NEXT
```

```
until 0
label NETWORK_END
select network
plus inputDistribution
plus distanceMatrix
Remove

include demo.praatinclude
```

Appendix B – Script for the similar scenario

```
# Praat script for the similar scenario, based on an earlier script by Boersma (2013)

form growingLexicon
	word Foreground_colour Yellow
	word Background_colour Maroon
	word Button_colour Olive
	word Font Times
	natural Font_size 35
	real number_of_categories_l1 2
	real number_of_categories_l2 2
endform

demo.foregroundColour$ = foreground_colour$
demo.backgroundColour$ = background_colour$
demo.buttonColour$ = button_colour$
demo.font$ = font$
demo.fontSize = font_size

#
# Properties of the network
#

spreadingRate = 0.01
actMin = 0.0
actMax = 1.0
actLeak = 1.0
learning_rate = 0.001
weightMin = -1.0
weightMax = 1.0
weightLeak = 0.0
xMin = 0.0
xMax = 10.0
yMin = 0.0
yMax = 10.0
instar = 0.5
outstar = 0.5
initialWeightMin = 0.0
initialWeightMax = 0.1
inhibitionAtSF = -0.5
inhibitionAtPF = -0.5
inhibitionAtAudF = -0.35
artConnStrength = 1.5
weightOfCentrArtConn = -0.25

pfL1.numberOfNodes = 4 * number_of_categories_l1
pfL2.numberOfNodes = 4 * number_of_categories_l2
```

```
sf.numberOfNodes = 20
audf.numberOfNodes = 40
artf.numberOfNodes = 0

pf.y = 7.0
sf.y = 5.0
audf.y = 2.0
artf.y = 0.5

artf.offsetNode = 0
audf.offsetNode = artf.offsetNode+artf.numberOfNodes
sf.offsetNode = audf.offsetNode+audf.numberOfNodes
pfL1.offsetNode = sf.offsetNode + sf.numberOfNodes
pfL2.offsetNode = pfL1.offsetNode + pfL1.numberOfNodes

peak_sharpness = 4
auditory_sharpness = 50
stdevOfCategory = (audf.numberOfNodes - 1) / peak_sharpness / 10
auditory_spreading = (audf.numberOfNodes - 1) / auditory_sharpness
numberOfTimesOfActivitySpreading = 100
inputExaggeration = 0.5

f1_strength = 1
weightNormalization = 0
l2_step = 50000
use_alternations = 1

mean_of_category_1 = 0.45
mean_of_category_2 = 0.65

procedure createNetwork
	network = Create empty Network... BilingualSpeakerVOT spreadingRate
	... linear actMin actMax actLeak learning_rate weightMin weightMax
	... weightLeak xMin xMax yMin yMax

	Set instar... instar
	Set outstar... outstar

	for i to artf.numberOfNodes
		Add node... xMin+(xMax-xMin)/artf.numberOfNodes*(i-0.5) artf.y 0 yes
	endfor

	for i to audf.numberOfNodes
		a = artConnStrength / (0.5 - 0.5 * audf.numberOfNodes ^ 2)
		b = (1 + audf.numberOfNodes) / 2
		Add connection... artf.offsetNode+1 audf.offsetNode+i weightOfCentrArtConn+a*(i-
		…b)^2 0
	endfor

	for i to audf.numberOfNodes
		Add node... xMin+(xMax-xMin)/audf.numberOfNodes*(i-0.5) audf.y 0 yes
	endfor

	for i to sf.numberOfNodes
		Add node... xMin+(xMax-xMin)/sf.numberOfNodes*(i-0.5) sf.y 0 no
	endfor

	for i to pfL1.numberOfNodes
		Add node... xMin+i/4 pf.y 0 yes
	endfor
```

```
	for i to pfL2.numberOfNodes
		Add node... xMin+4+i/4 pf.y 0 yes
	Endfor

#
# Excitatory connections
#

	for i to audf.numberOfNodes
		for j to sf.numberOfNodes
			Add connection... audf.offsetNode+i sf.offsetNode+j
			…randomUniform(initialWeightMin,initialWeightMax) 1.0
		endfor
	endfor

	for i to sf.numberOfNodes
		for j to pfL1.numberOfNodes
			Add connection... sf.offsetNode+i pfL1.offsetNode+j
			…randomUniform(initialWeightMin,initialWeightMax) 1.0
		endfor
	endfor

#
# Inhibitory connections
#

	for i to artf.numberOfNodes
		for j to audf.numberOfNodes
			Add connection... artf.offsetNode+i audf.offsetNode+j 0 0.0
		endfor
	endfor

	for .i to audf.numberOfNodes - 1
		for .j from .i + 1 to audf.numberOfNodes
			Add connection... audf.offsetNode+.i audf.offsetNode+.j inhibitionAtAudF 0.0
		endfor
	endfor
	for i to sf.numberOfNodes - 1
		for j from i+1 to sf.numberOfNodes
			Add connection... sf.offsetNode+i sf.offsetNode+j inhibitionAtSF 0.0
		endfor
	endfor

	for i to pfL1.numberOfNodes - 1
		for j from i+1 to pfL1.numberOfNodes
			Add connection... pfL1.offsetNode+i pfL1.offsetNode+j inhibitionAtPF 0.0
		endfor
	endfor

	for i to pfL2.numberOfNodes - 1
		for j from i+1 to pfL2.numberOfNodes
			Add connection... pfL2.offsetNode+i pfL2.offsetNode+j inhibitionAtPF 0.0
		endfor
	endfor

	inputDistribution = Create Matrix... inputDistribution 0.5 audf.numberOfNodes+0.5
	…audf.numberOfNodes 1.0 1 1 1 1 1 1 0.0
	distanceMatrix = Create Matrix... distance 0.5 audf.numberOfNodes+0.5
	…audf.numberOfNodes+1 1.0 0.5 1 1 1 1 1 0.0
```

```
endproc

label NETWORK
call createNetwork
step = 0
pfNode = 1
audNode = 1
l1 = 1
l2 = 0
repeat
	call demo.erase
	call demo.centredTitle BilingualSpeakerVOT
	demo.textY += 13
	demo Select inner viewport... 20 80 20 80
	select network
	demo Draw... yes
	demo Text... xMin right artf.y half [[ArtF]]
	demo Text... xMin right audf.y half [[AudF]]
	demo Text... xMin right sf.y half /SF/
	demo Text... xMin right pf.y half Lexicons
	select inputDistribution
	demo Magenta
	demo Line width... 3
	for i to number_of_categories_l2
		demo Draw rows... 0 0 i-0.5 i+0.5 0 step/3
	endfor
	demo 'demo.foregroundColour$'
	demo Line width... 2


	demo Select inner viewport... 0 100 0 100
	demo Axes... 0 100 0 100
	demo Text... 50 centre 10 half After step 'step'.
	call demo.button 88 98 50 1000↑
	call demo.button 88 98 40 100↑
	call demo.button 88 98 30 5000↑
	call demo.button 88 98 20 1↑
	call demo.button 88 98 10 new
	call demo.button 2 12 10 set...

	while demoWaitForInput ( )
		if demoInput ("a")
			select network
			Zero activities... 0 0

			if l1 = 1

				for i to pfL1.numberOfNodes
					Set clamping... pfL1.offsetNode+i no
				endfor

				for i to pfL2.numberOfNodes
					Set clamping... pfL2.offsetNode+i no
				endfor

				audNode = max (1, min (round (audNode), audf.numberOfNodes)) + 1

				if audNode > audf.numberOfNodes
					audNode = 1
```

```
		endif

		for .i to audf.numberOfNodes
			Set clamping... audf.offsetNode+.i yes
			Set activity... audf.offsetNode+.i   exp (-0.5 * (.i - audNode) ^ 2
			…/ auditory_spreading ^ 2) * inputExaggeration
		endfor

		Spread activities... numberOfTimesOfActivitySpreading
		goto NETWORK_NEXT

	elsif l2 = 1

		for i to pfL2.numberOfNodes
			Set clamping... pfL2.offsetNode+i no
		endfor

		for i to pfL1.numberOfNodes
			Set clamping... pfL1.offsetNode+i no
		endfor

		audNode = max (1, min (round (audNode), audf.numberOfNodes)) + 1

		if audNode > audf.numberOfNodes
			audNode = 1
		endif

		for .i to audf.numberOfNodes
			Set clamping... audf.offsetNode+.i yes
			Set activity... audf.offsetNode+.i   exp (-0.5 * (.i - audNode) ^ 2
			…/ auditory_spreading ^ 2) * inputExaggeration
		endfor

		Spread activities... numberOfTimesOfActivitySpreading
		goto NETWORK_NEXT
	endif

elsif demoInput ("12")
	select network
	Zero activities... 0 0
	pfNode = index ("12", demoKey$ ())

	if l1 = 1
		if pfNode <= number_of_categories_l1
			for .i to audf.numberOfNodes
				Set clamping... audf.offsetNode+.i no
			endfor

			for .i to pfL1.numberOfNodes
				Set activity... pfL1.offsetNode+.i 0
				Set clamping... pfL1.offsetNode+.i yes
			endfor

			for i to pfL2.numberOfNodes
				Set clamping... pfL2.offsetNode+i yes
				Set activity... pfL2.offsetNode+i 0
			endfor

			for .i to pfL1.numberOfNodes/number_of_categories_l1
```

```
                .k = (pfNode-
                …1)*(pfL1.numberOfNodes/number_of_categories_l1)
                …+ .i
                Set activity... pfL1.offsetNode+.k 1
            endfor

            Spread activities... numberOfTimesOfActivitySpreading
        endif
        goto NETWORK_NEXT

    elsif l2 = 1
        if pfNode <= number_of_categories_l2
            for .i to audf.numberOfNodes
                Set clamping... audf.offsetNode+.i no
            endfor

            for .i to pfL2.numberOfNodes
                Set activity... pfL2.offsetNode+.i 0
                Set clamping... pfL2.offsetNode+.i yes
            endfor

            for i to pfL1.numberOfNodes
                Set clamping... pfL1.offsetNode+i yes
                Set activity... pfL1.offsetNode+i 0
            endfor

            for .i to pfL2.numberOfNodes/number_of_categories_l2
                .k = (pfNode-
                …1)*(pfL2.numberOfNodes/number_of_categories_l2)
                …+ .i
                Set activity... pfL2.offsetNode+.k 1
            endfor

            Spread activities... numberOfTimesOfActivitySpreading
        endif
    endif
    goto NETWORK_NEXT

elsif demoClickedIn (2, 12, 10-4, 10+4) or demoInput ("z")   ; set...
    beginPause ("Settings")
        boolean ("l1", 0)
        boolean ("l2", 0)
    clicked = endPause ("Cancel", "Set", 2)

    if clicked = 2
        if l1 = 1
            mean_of_category_1 = 0.45
            mean_of_category_2 = 0.65
            writeInfoLine: "l1 aan"
        elsif l2 = 1
            mean_of_category_1 = 0.50
            mean_of_category_2 = 0.70
            writeInfoLine: "l2 aan"
        endif

    endif

    goto NETWORK_NEXT
endif
```

```
numberOfSteps =
... if demoClickedIn (88, 98, 20-4, 20+4) or demoInput ("↑") then 1 else
... if demoClickedIn (88, 98, 30-4, 30+4) or demoInput ("v") then 5000 else
... if demoClickedIn (88, 98, 40-4, 40+4) or demoInput ("h") then 100 else
... if demoClickedIn (88, 98, 50-4, 50+4) or demoInput ("d") then 1000 else
... if demoInput ("4") then 10000 else 0 fi fi fi fi fi

if numberOfSteps <> 0
	select network
	for ministep to abs (numberOfSteps)

		if step = l2_step
			select network

### Add Excitatory connections between the new lexicon layer and sf ###

			for .i to sf.numberOfNodes
				for .j to pfL2.numberOfNodes
					Add connection... sf.offsetNode+.i
					…pfL2.offsetNode+.j
					…randomUniform
					…(initialWeightMin,initialWeightMax) 1.0
				endfor
			endfor


			for .i to sf.numberOfNodes
				for .j to pfL2.numberOfNodes
					Add connection... sf.offsetNode+.i
					…pfL2.offsetNode+.j
					…randomUniform
					…(initialWeightMin,initialWeightMax) 1.0
				endfor
			endfor

		step += 1
		Zero activities... 0 0

	#### L1 surroundings ####

		if l1 = 1
			category = randomInteger (1, number_of_categories_l1)
			meanOfCategory = 1 + mean_of_category_'category' *
			…(audf.numberOfNodes - 1)
			audNode = randomGauss (meanOfCategory,
			…stdevOfCategory)

			for i to audf.numberOfNodes
				Set clamping... audf.offsetNode+i yes
				Set activity... audf.offsetNode+i   exp (-0.5 * (i -
				…audNode) ^ 2 / auditory_spreading ^ 2) *
				…inputExaggeration
			endfor

			for i from 1 to pfL2.numberOfNodes
				Set clamping... pfL2.offsetNode+i yes
			endfor

			for i from 1 to pfL1.numberOfNodes
				Set clamping... pfL1.offsetNode+i yes
```

```
			Set activity... pfL1.offsetNode+i 0
		endfor

		for i to pfL1.numberOfNodes/number_of_categories_l1
			Set activity... pfL1.offsetNode+(category-
			…1)*(pfL1.numberOfNodes/number_of_categories_l1)
			…+i 1
		endfor

		for i to pfL1.numberOfNodes/number_of_categories_l1

			Set activity... pfL1.offsetNode+(category-
			…1)*(pfL1.numberOfNodes/number_of_categories_l1)
			…+i 1
		endfor

		Spread activities... numberOfTimesOfActivitySpreading
		Update weights

#### L2 surroundings ####

	elsif l2 = 1
		category = randomInteger (1, number_of_categories_l1)
		meanOfCategory = 1 + mean_of_category_'category' *
		…(audf.numberOfNodes - 1)
		audNode = randomGauss (meanOfCategory,
		…stdevOfCategory)

		for i to audf.numberOfNodes
			Set clamping... audf.offsetNode+i yes
			Set activity... audf.offsetNode+i   exp (-0.5 * (i -
			…audNode) ^ 2 / auditory_spreading ^ 2) *
			…inputExaggeration
		endfor

		for i from 1 to pfL1.numberOfNodes
			Set clamping... pfL1.offsetNode+i yes
		endfor

		for i from 1 to pfL2.numberOfNodes
			Set clamping... pfL2.offsetNode+i yes
			Set activity... pfL2.offsetNode+i 0
		endfor

		for i to pfL2.numberOfNodes/number_of_categories_l1

			Set activity... pfL2.offsetNode+(category-
			…1)*(pfL2.numberOfNodes/number_of_categories_l2)
			…+i 1
		endfor

		for i to pfL2.numberOfNodes/number_of_categories_l2
			Set activity... pfL2.offsetNode+(category-
			…1)*(pfL2.numberOfNodes/number_of_categories_l2)
			…+i 1
		endfor

		Spread activities... numberOfTimesOfActivitySpreading
		Update weights
```

```
                endif

                select inputDistribution
                Formula... self + exp (-0.5 * (col - audNode) ^ 2 / auditory_spreading ^
                …2) * inputExaggeration
                select network

            endfor
            goto NETWORK_NEXT
        endif
        goto NETWORK_END demoInput ("h←• →")
    endwhile
    label NETWORK_NEXT
until 0
label NETWORK_END
select network
plus inputDistribution
plus distanceMatrix
Remove

include demo.praatinclude
```

Appendix C – Script for the subset scenario

```
# Praat script for the subset scenario, based on an earlier script by Boersma (2013)

form growingLexicon
	word Foreground_colour Yellow
	word Background_colour Maroon
	word Button_colour Olive
	word Font Times
	natural Font_size 35
	real number_of_vowels_l1 5
	real number_of_vowels_l2 3
endform

demo.foregroundColour$ = foreground_colour$
demo.backgroundColour$ = background_colour$
demo.buttonColour$ = button_colour$
demo.font$ = font$
demo.fontSize = font_size

#
# Properties of the network
#

spreadingRate = 0.01
actMin = 0.0
actMax = 1.0
actLeak = 1.0
learning_rate = 0.001
weightMin = -1.0
weightMax = 1.0
weightLeak = 0.0
xMin = 0.0
xMax = 10.0
yMin = 0.0
yMax = 10.0
instar = 0.5
outstar = 0.5
initialWeightMin = 0.0
initialWeightMax = 0.1
inhibitionAtSF = -0.5
inhibitionAtPF = -0.5
inhibitionAtAudF = -0.35
artConnStrength = 1.5
weightOfCentrArtConn = -0.25

pfL1.numberOfNodes = 4 * number_of_vowels_l1
pfL2.numberOfNodes = 4 * number_of_vowels_l2
sf.numberOfNodes = 20
audf.numberOfNodes = 40
artf.numberOfNodes = 0

pf.y = 7.0
sf.y = 5.0
audf.y = 2.0
artf.y = 0.5

artf.offsetNode = 0
audf.offsetNode = artf.offsetNode+artf.numberOfNodes
```

```
sf.offsetNode = audf.offsetNode+audf.numberOfNodes
pfL1.offsetNode = sf.offsetNode + sf.numberOfNodes
pfL2.offsetNode = pfL1.offsetNode + pfL1.numberOfNodes

peak_sharpness = 2
auditory_sharpness = 50
stdevOfAmbient = (audf.numberOfNodes - 1) / peak_sharpness / 10
auditory_spreading = (audf.numberOfNodes - 1) / auditory_sharpness
numberOfTimesOfActivitySpreading = 100
inputExaggeration = 0.25 ; veranderd

f1_strength = 1
weightNormalization = 0
l2_step = 50000
use_alternations = 1

vowel5_audf1_mean = 0.10*audf.numberOfNodes
vowel5_audf2_mean = 0.90*audf.numberOfNodes
vowel3_audf1_mean = 0.40*audf.numberOfNodes
vowel3_audf2_mean = 0.75*audf.numberOfNodes
vowel1_audf1_mean = 0.10*audf.numberOfNodes
vowel1_audf2_mean = 0.60*audf.numberOfNodes
vowel2_audf1_mean = 0.25*audf.numberOfNodes
vowel2_audf2_mean = 0.8375*audf.numberOfNodes
vowel4_audf1_mean = 0.25*audf.numberOfNodes
vowel4_audf2_mean = 0.6625*audf.numberOfNodes


procedure createNetwork
	network = Create empty Network... BilingualSpeaker spreadingRate
	... linear actMin actMax actLeak learning_rate weightMin weightMax
	... weightLeak xMin xMax yMin yMax

	Set instar... instar
	Set outstar... outstar

	for i to artf.numberOfNodes
		Add node... xMin+(xMax-xMin)/artf.numberOfNodes*(i-0.5) artf.y 0 yes
	endfor

	for i to audf.numberOfNodes
		a = artConnStrength / (0.5 - 0.5 * audf.numberOfNodes ^ 2)
		b = (1 + audf.numberOfNodes) / 2
		Add connection... artf.offsetNode+1 audf.offsetNode+i weightOfCentrArtConn+a*(i-
		…b)^2 0
	endfor

	for i to audf.numberOfNodes
		Add node... xMin+(xMax-xMin)/audf.numberOfNodes*(i-0.5) audf.y 0 yes
	endfor

	for i to sf.numberOfNodes
		Add node... xMin+(xMax-xMin)/sf.numberOfNodes*(i-0.5) sf.y 0 no
	endfor

	for i to pfL1.numberOfNodes
		Add node... xMin+i/4 pf.y 0 yes
	endfor
```

```
	for i to pfL2.numberOfNodes
		Add node... xMin+6+i/4 pf.y 0 yes
	endfor

#
# Excitatory connections
#

	for i to audf.numberOfNodes
		for j to sf.numberOfNodes
			Add connection... audf.offsetNode+i sf.offsetNode+j
			…randomUniform(initialWeightMin,initialWeightMax) 1.0
		endfor
	endfor

	for i to sf.numberOfNodes
		for j to pfL1.numberOfNodes
			Add connection... sf.offsetNode+i pfL1.offsetNode+j
			…randomUniform(initialWeightMin,initialWeightMax) 1.0
		endfor
	endfor

#
# Inhibitory connections
#

	for i to artf.numberOfNodes
		for j to audf.numberOfNodes
			Add connection... artf.offsetNode+i audf.offsetNode+j
			…randomUniform(initialWeightMin,initialWeightMax) 0.0
		endfor
	endfor

	for .i to audf.numberOfNodes - 1
		for .j from .i + 1 to audf.numberOfNodes
			Add connection... audf.offsetNode+.i audf.offsetNode+.j inhibitionAtAudF 0.0
		endfor
	endfor


	for i to sf.numberOfNodes - 1
		for j from i+1 to sf.numberOfNodes
			Add connection... sf.offsetNode+i sf.offsetNode+j inhibitionAtSF 0.0
		endfor
	endfor

	for i to pfL1.numberOfNodes - 1
		for j from i+1 to pfL1.numberOfNodes
			Add connection... pfL1.offsetNode+i pfL1.offsetNode+j inhibitionAtPF 0.0
		endfor
	endfor

	for i to pfL2.numberOfNodes - 1
		for j from i+1 to pfL2.numberOfNodes
			Add connection... pfL2.offsetNode+i pfL2.offsetNode+j inhibitionAtPF 0.0
		endfor
	endfor

	inputDistribution = Create Matrix... inputDistribution 0.5 audf.numberOfNodes+0.5
	…audf.numberOfNodes 1.0 1 1 1 1 1 1 0.0
```

```
	distanceMatrix = Create Matrix... distance 0.5 audf.numberOfNodes+0.5
	…audf.numberOfNodes+1 1.0 0.5 1 1 1 1 1 0.0

endproc

label NETWORK
call createNetwork
step = 0
pfNode = 1
audNode = 1
l1 = 1
l2 = 0
repeat
	call demo.erase
	call demo.centredTitle BilingualSpeaker
	demo.textY += 13
	demo Select inner viewport... 20 80 20 80
	select network
	demo Draw... yes
	demo Text... xMin right artf.y half [[ArtF]]
	demo Text... xMin right audf.y half [[AudF]]
	demo Text... xMin right sf.y half /SF/
	demo Text... xMin right pf.y half Lexicons
	select inputDistribution
	demo Magenta
	demo Line width... 3
	for i to number_of_vowels_l2
	demo Draw rows... 0 0 i-0.5 i+0.5 0 step/3
	endfor
	demo 'demo.foregroundColour$'
	demo Line width... 2


	demo Select inner viewport... 0 100 0 100
	demo Axes... 0 100 0 100
	demo Text... 50 centre 10 half After step 'step'.
	call demo.button 88 98 50 1000↑
	call demo.button 88 98 40 100↑
	call demo.button 88 98 30 5000↑
	call demo.button 88 98 20 1↑
	call demo.button 88 98 10 new
	call demo.button 2 12 10 set...

	while demoWaitForInput ( )
		if demoInput ("a")
			select network
			Zero activities... 0 0

			if l1 = 1

				for i to pfL1.numberOfNodes
					Set clamping... pfL1.offsetNode+i no
				endfor

				for i to pfL2.numberOfNodes
					Set clamping... pfL2.offsetNode+i no
				endfor

				audNode = max (1, min (round (audNode), audf.numberOfNodes)) + 1
```

```
            if audNode > audf.numberOfNodes
                audNode = 1
            endif

            whichVowel = randomInteger (1, number_of_vowels_l1)
            vowel.f1 = vowel'whichVowel'_audf1_mean
            vowel.f2 = vowel'whichVowel'_audf2_mean

            audNode1 = randomGauss (vowel.f1, stdevOfAmbient/2)
            audNode2 = randomGauss (vowel.f2, stdevOfAmbient/2)

            for i to audf.numberOfNodes
                Set clamping... audf.offsetNode+i yes
                Set activity... audf.offsetNode+i
                ... f1_strength * exp (-0.5 * (i - audNode1) ^ 2 /
                …auditory_spreading ^ 2) * inputExaggeration +
                ...exp (-0.5 * (i - audNode2) ^ 2 / auditory_spreading ^ 2) *
                …inputExaggeration
            endfor

            Spread activities... numberOfTimesOfActivitySpreading*5
            goto NETWORK_NEXT

        elsif l2 = 1

            for i to pfL2.numberOfNodes
                Set clamping... pfL2.offsetNode+i no
            endfor

            for i to pfL1.numberOfNodes
                Set clamping... pfL1.offsetNode+i no
            endfor

            audNode = max (1, min (round (audNode), audf.numberOfNodes)) + 1

            if audNode > audf.numberOfNodes
                audNode = 1
            endif

            whichVowel = randomInteger (1, number_of_vowels_l1)
            vowel.f1 = vowel'whichVowel'_audf1_mean
            vowel.f2 = vowel'whichVowel'_audf2_mean

            audNode1 = randomGauss (vowel.f1, stdevOfAmbient/2)
            audNode2 = randomGauss (vowel.f2, stdevOfAmbient/2)

            for i to audf.numberOfNodes
                Set clamping... audf.offsetNode+i yes
                Set activity... audf.offsetNode+i
                ... f1_strength * exp (-0.5 * (i - audNode1) ^ 2 /
                …auditory_spreading ^ 2) * inputExaggeration +
                ... exp (-0.5 * (i - audNode2) ^ 2 / auditory_spreading ^ 2) *
                …inputExaggeration
            endfor

            Spread activities... numberOfTimesOfActivitySpreading*5
            goto NETWORK_NEXT
        endif

    elsif demoInput ("12345")
```

```
		select network
		Zero activities... 0 0
		pfNode = index ("12345", demoKey$ ())

		if l1 = 1
			if pfNode <= number_of_vowels_l1
				for .i to audf.numberOfNodes
					Set clamping... audf.offsetNode+.i no
				endfor

				for .i to pfL1.numberOfNodes
					Set activity... pfL1.offsetNode+.i 0
					Set clamping... pfL1.offsetNode+.i yes
				endfor

				for i to pfL2.numberOfNodes
					Set clamping... pfL2.offsetNode+i yes
					Set activity... pfL2.offsetNode+i 0
				endfor

				for .i to pfL1.numberOfNodes/number_of_vowels_l1
					.k = (pfNode-1)
					…*(pfL1.numberOfNodes/number_of_vowels_l1) + .i
					Set activity... pfL1.offsetNode+.k 1
				endfor

				Spread activities... numberOfTimesOfActivitySpreading
			endif
			goto NETWORK_NEXT

		elsif l2 = 1
			if pfNode <= number_of_vowels_l2

				for .i to audf.numberOfNodes
					Set clamping... audf.offsetNode+.i no
				endfor

				for .i to pfL2.numberOfNodes
					Set activity... pfL2.offsetNode+.i 0
					Set clamping... pfL2.offsetNode+.i yes
				endfor

				for i to pfL1.numberOfNodes
					Set clamping... pfL1.offsetNode+i yes
					Set activity... pfL1.offsetNode+i 0
				endfor


				for .i to pfL2.numberOfNodes/number_of_vowels_l2
					.k = (pfNode-1)
					…*(pfL2.numberOfNodes/number_of_vowels_l2) + .i
					Set activity... pfL2.offsetNode+.k 1
				endfor

				Spread activities... numberOfTimesOfActivitySpreading
			endif
		endif
		goto NETWORK_NEXT

	elsif demoClickedIn (2, 12, 10-4, 10+4) or demoInput ("z")   ; set...
```

```
			beginPause ("Settings")
				boolean ("l1", 0)
				boolean ("l2", 0)
			clicked = endPause ("Cancel", "Set", 2)

			if clicked = 2
				if l1 = 1
					vowel5_audf1_mean = 0.10*audf.numberOfNodes
					vowel5_audf2_mean = 0.90*audf.numberOfNodes
					vowel3_audf1_mean = 0.40*audf.numberOfNodes
					vowel3_audf2_mean = 0.75*audf.numberOfNodes
					vowel1_audf1_mean = 0.10*audf.numberOfNodes
					vowel1_audf2_mean = 0.60*audf.numberOfNodes
					vowel2_audf1_mean = 0.25*audf.numberOfNodes
					vowel2_audf2_mean = 0.8375*audf.numberOfNodes
					vowel4_audf1_mean = 0.25*audf.numberOfNodes
					vowel4_audf2_mean = 0.6625*audf.numberOfNodes
					writeInfoLine: "l1 aan"
				elsif l2 = 1
					vowel1_audf1_mean = 0.10*audf.numberOfNodes
					vowel1_audf2_mean = 0.90*audf.numberOfNodes
					vowel2_audf1_mean = 0.40*audf.numberOfNodes
					vowel2_audf2_mean = 0.75*audf.numberOfNodes
					vowel3_audf1_mean = 0.10*audf.numberOfNodes
					vowel3_audf2_mean = 0.60*audf.numberOfNodes

					writeInfoLine: "l2 aan"
				endif

			endif


			goto NETWORK_NEXT
		endif

		numberOfSteps =
		... if demoClickedIn (88, 98, 20-4, 20+4) or demoInput ("↑") then 1 else
		... if demoClickedIn (88, 98, 30-4, 30+4) or demoInput ("v") then 5000 else
		... if demoClickedIn (88, 98, 40-4, 40+4) or demoInput ("h") then 100 else
		... if demoClickedIn (88, 98, 50-4, 50+4) or demoInput ("d") then 1000 else
		... if demoInput ("4") then 10000 else 0 fi fi fi fi fi

		if numberOfSteps <> 0
			select network
			for ministep to abs (numberOfSteps)

				if step = l2_step
					select network

### Add Excitatory connections between the new lexicon layer and sf ###

					for .i to sf.numberOfNodes
						for .j to pfL2.numberOfNodes
							Add connection... sf.offsetNode+.i
							…pfL2.offsetNode+.j
							…randomUniform
							…(initialWeightMin,initialWeightMax) 1.0
						endfor
					endfor
```

```
				for .i to sf.numberOfNodes
					for .j to pfL2.numberOfNodes
						Add connection... sf.offsetNode+.i
						…pfL2.offsetNode+.j
						…randomUniform
						…(initialWeightMin,initialWeightMax) 1.0
					endfor
				endfor

			endif

			step += 1
			Zero activities... 0 0

		#### L1 surroundings ####

			if l1 = 1
				whichVowel = randomInteger (1, number_of_vowels_l1)

				vowel.f1 = vowel'whichVowel'_audf1_mean
				vowel.f2 = vowel'whichVowel'_audf2_mean

				audNode1 = randomGauss (vowel.f1, stdevOfAmbient/2)
				audNode2 = randomGauss (vowel.f2, stdevOfAmbient/2)

				for i from 1 to audf.numberOfNodes
					Set clamping... audf.offsetNode+i yes
					Set activity... audf.offsetNode+i
					... f1_strength * exp (-0.5 * (i - audNode1) ^ 2 /
					…auditory_spreading ^ 2) * inputExaggeration +
					... exp (-0.5 * (i - audNode2) ^ 2 / auditory_spreading ^
					…2) * inputExaggeration
				endfor

				for i from 1 to pfL2.numberOfNodes
					Set clamping... pfL2.offsetNode+i yes
					Set activity... pfL2.offsetNode+i 0
				endfor

				for i from 1 to pfL1.numberOfNodes
					Set clamping... pfL1.offsetNode+i yes
					Set activity... pfL1.offsetNode+i 0
				endfor

				if whichVowel <= number_of_vowels_l1
					for i to pfL1.numberOfNodes/number_of_vowels_l1
						Set activity... pfL1.offsetNode+(whichVowel-1)
						…*(pfL1.numberOfNodes/
						…number_of_vowels_l1)+i 1
					endfor
				endif

				if whichVowel <= number_of_vowels_l1
					for i to pfL1.numberOfNodes/number_of_vowels_l1
						Set activity... pfL1.offsetNode+(whichVowel-1)
						…*(pfL1.numberOfNodes/
						…number_of_vowels_l1)+i 1
					endfor
				endif
```

```
		Spread activities... numberOfTimesOfActivitySpreading
		Update weights

#### L2 surroundings ####

	elsif l2 = 1
		whichVowel = randomInteger (1, number_of_vowels_l2)

		vowel.f1 = vowel'whichVowel'_audf1_mean
		vowel.f2 = vowel'whichVowel'_audf2_mean

		audNode1 = randomGauss (vowel.f1, stdevOfAmbient/2)
		audNode2 = randomGauss (vowel.f2, stdevOfAmbient/2)

		for i from 1 to audf.numberOfNodes
			Set clamping... audf.offsetNode+i yes
			Set activity... audf.offsetNode+i
			... f1_strength * exp (-0.5 * (i - audNode1) ^ 2 /
			…auditory_spreading ^ 2) * inputExaggeration +
			... exp (-0.5 * (i - audNode2) ^ 2 / auditory_spreading ^
			…2) * inputExaggeration
		endfor

		for i from 1 to pfL1.numberOfNodes
			Set clamping... pfL1.offsetNode+i yes
			Set activity... pfL1.offsetNode+i 0
		endfor

		for i from 1 to pfL2.numberOfNodes
			Set clamping... pfL2.offsetNode+i yes
			Set activity... pfL2.offsetNode+i 0
		endfor

		if whichVowel <= number_of_vowels_l2
			for i to pfL2.numberOfNodes/number_of_vowels_l2
				Set activity... pfL2.offsetNode+(whichVowel-1)
				…*(pfL2.numberOfNodes/
				…number_of_vowels_l2)+i 1
			endfor
		endif

		if whichVowel <= number_of_vowels_l2
			for i to pfL2.numberOfNodes/number_of_vowels_l2
				Set activity... pfL2.offsetNode+(whichVowel-1)
				…*(pfL2.numberOfNodes/
				…number_of_vowels_l2)+i 1
			endfor
		endif

		Spread activities... numberOfTimesOfActivitySpreading
		Update weights

	endif

	select inputDistribution
	Formula... self + f1_strength * (exp (-0.5 * (col - audNode1) ^ 2 /
	…auditory_spreading ^ 2)) * inputExaggeration
	... + (exp (-0.5 * (col - audNode2) ^ 2 / auditory_spreading ^ 2)) *
	…inputExaggeration
	select network
```

```
                        endfor
                        goto NETWORK_NEXT
                endif
                goto NETWORK_END demoInput ("h←• →")
        endwhile
        label NETWORK_NEXT
until 0
label NETWORK_END
select network
plus inputDistribution
plus distanceMatrix
Remove

include demo.praatinclude
```