Speech Synthesis by Artificial Neural Networks (AI / Speech processing / Signal processing)

Christos P. Yiakoumettis

Department of Informatics
University of Sussex, UK
(Email: c.yiakoumettis@sussex.ac.uk)

Abstract

This research was accomplished during my studies at University of Sussex under supervision of Dr Si Wu. The whole project was focused on speech synthesis by artificial neural networks. The objective of the whole research is to implement an improvement of the speech synthesis process. A different approach on some speech synthesis procedures was introduced by neural networks. That approach was independent from any language. It was able to learn and follow the rules of the language but it wasn't based on them. The main difference of this approach compared with the previous ones is that this one produces speech directly. The network will be able to produce directly speech instead of classifying sound segments. A multi layer perceptron with two layers will be designed and trained. The main input of the network was a phoneme paired with a stress value. The whole approach was presented in English language. A text to acoustic English dictionary was used for the transformation of the plain text to a sequence of phonemes. Unfortunately the performance of the network wasn't the expected one. The network was trained but it was impossible to generalise its performance for bigger vocabulary.

Keywords: Feed-forward neural network, multi-layer perceptron, phoneme, robotic speech, spectrogram, speech, speech synthesis, synthesis, synthesizer

1. Introduction

Humans can produce and recognise directly speech. Somehow one is the opposite process of the other. In addition there are methodologies on speech recognition process in which the input is real speech, sampled every few milliseconds and the output is whole words. An attempt of inverting that process like humans do will be introduced for producing speech.

The evolution of personal computers has made possible the analysis of big amount of information in real time. The complexity and the difficulty to understand and build a mathematical model of human speech either for input or for output has led us to look for alternative ways to manipulate it. The best second approach can be accomplished by artificial neural networks [Zurada J. (1992)], [Vlahabas I. (2002)].

The results of many years of research led us today to talk about speech technology. Speech technology is the field of science that is focused on speech input and speech output on machines. The above, are also known as speech recognition and speech synthesis.

The aim of this research is focused only on speech synthesis, presenting an alternative way of speech synthesis. Using dynamical structures of artificial intelligence field an artificial neural network was designed and trained to produce speech. The input of the system will be plain text in acoustic representation and the output was directly speech. The speech was presented by pictures which was the results of spectral analysis of the human speech samples.

2. Speech and Speech Synthesis

2.1 Speech Synthesis

Speech synthesis or text to speech is the process of converting the written language to spoken. It is used by machines in order to communication with us like human do.

The extraction of speech from plain text is a complex process and it consists of two phases [Lemmetty S. (1999)]. The first one is the text analysis, in which the plain text in transformed to an acoustic representation. This step is accomplished by natural language processing unit or a dynamic structure like neural networks. In the last steps of the speech synthesis process the speech waveform is generated by the acoustic data and prosodic information from the previous steps. The speech is produced by a low level synthesizer [Lemmetty S. (1999)].

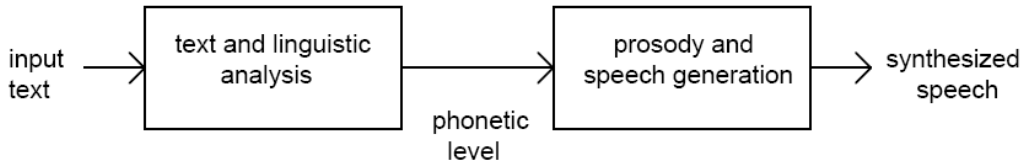

Figure 2.2.1. *Speech synthesis overview block diagram*

There are three different general approaches to build a speech synthesis engine. The first approach is using pre-recorded words from a data base to synthesise phrases used basically for specific applications with small vocabulary. This methodology will not be discussed deeply as it is far away from the expectations of this paper. Emphasis will be on general purpose speech synthesis approaches, which have unlimited vocabulary.

2.2 The sound representation

The main idea of this approach is based on the physical specifications of the sound and how it changes during time. A sound can be a very complex mixture of different simple sounds. Each simple sound changes smoothly during time. Moreover the mixed sound changes smoothly on the space of frequencies and time. It is known that the voice of an adult male speaker is in the range of 300Hz up to 3 kHz [Yiannakoudakis E. (1987)]. Moreover, intelligible speech can be distinguished from a bandwidth as low as 2.2 kHz [Pearson P.], [The American Radio Relay League (1991)]. Spectrograms were used to visualise that space. Using spectrograms, it is possible to describe the sound output with a few vectors, considering the fact that the amplitude changes smoothly against the frequencies and the time. Only the changes matter and it always changes smoothly. In that way the dimensionality of the problem is reduced and it is possible to be produced by a neural network.

3. The artificial Neural Network

3.1 The speech data

The input of the network will be the acoustic representation of the plain text and it will be represented by phonemes. Thirty nine phonemes were used for English pronunciations and one addition for the space between the words. Each phoneme is paired by an integer that represents the stress of the phoneme. Consonants phonemes don't have stress but vowels phonemes may have either two types of stress or none. The input of the network was the previous phoneme, the current one and the next one. All the phonemes were paired with a stress value.

The output of the network will be 121 different vectors on the three dimensional space of frequencies and amplitude against the time, one additional vector that represents the time duration of the output segment and one for describing the distribution of the output vectors. Each time the network will produce different segments of speech. The duration of each output was different and it was depending on the type of the input phonemes. Moreover the distribution of the vectors varied on space of frequency, amplitude and time. Vowel phonemes last more than consonants phonemes.

The output vectors will present a two dimensional matrix that represents a speech segment on a three dimensional space of frequencies and amplitude against time. For the consonants phonemes there were placed more vectors on the frequency axis and less on time axis. The opposite happened on vowels.

For the creation of the data a vocabulary of the most common English language word were used. A set of 1000 most common words were used for training purpose and the

next 500 ones for testing purpose. The sound samples were created by a speech synthesis engine of Microsoft's corporation.

3.2 The artificial Neural Network

The artificial neural network had six input vectors, three phonemes and their stress. The output was 123 vectors. The first 121 vectors were representing the produced sound segment. One output vector was used to represent the number of vectors on time axis. The last vector was used to represent the duration of the produced sound segment in milliseconds.

The training process was accomplished by the back propagation algorithm. The input data were normalized and rescaled in the range of [0, 1]. Principle component analysis (PCA) was impossible as the dimensionality of the output is much bigger than the dimensionality of the input. Besides, the dimensionality of the output is too big, and all the input vectors are required. The whole process was very slow even with learning rate 10-10. The network couldn't be trained even after 35.000 iterations. Bigger learning rate was leading to increment of the error during the training process. In order to train the network, it had to be splitted into two sub networks.

The two sub networks were connected individually to the six input neurons. Each sub network had the same number of hidden neurons but different number of outputs. The outputs were splitted into the two sub networks. The first sub network was producing the first 61 output vectors and the next one the remaining output vectors and the additional time and time axis distribution of vectors output. A block diagram of the change can be found on the next figure 3.1.1.

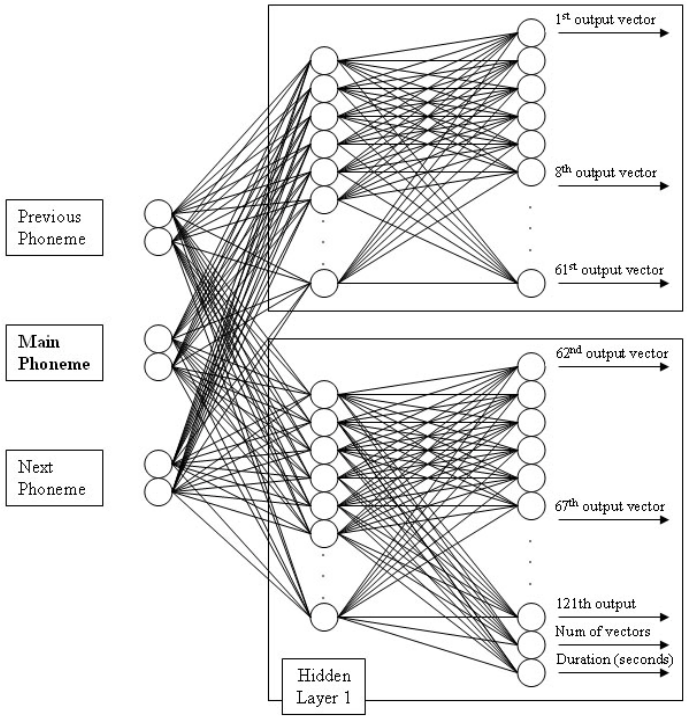

Figure 3.1.1. *Block diagram of sub networks*

The training process was accomplished in each individual sub network. In the end of the process the output vectors gathered all together again. Training each individual sub network the whole process completed much faster. After 46 for the first sub network and 47 for the second one, iteration the sub networks were trained with fault tolerance smaller than 10-23. The learning rate was set to 10-4 value. The training error of each sub network is plotted in the next figure 3.1.2.

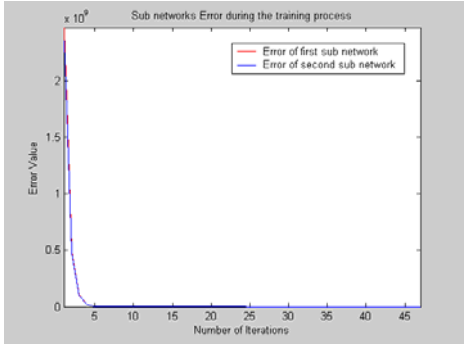

Figure 3.1.2. *Training Error of each sub network individually*

3.3 Training process and issues

Dividing the whole network structure into two sub networks the training process was possible to be completed. The training process was completed successfully with training error smaller than 10^{-22}. The training set had the thousands most common English words and the testing data set had the next 500 most common words. The inputs couldn't be randomized as each input represents a phoneme inside a word. The sequence of the phonemes matters in the word's pronunciation.

Unfortunately the testing and the validation process came to decline the aims and the objectives of this research. The network was unable to be generalised for more input data. It seems that it was unable to learn the patterns of the input data set. The generalization error was in very high values, close to $2.4778 * 10^{12}$. It was impossible to reduce the generalization error. Different networks structures were tested from 6 hidden neurons up to 150 with an incensement step of 2 neurons. The generalization error was always close the initial value of it. In the next figures the required iterations of each network (figure 3.3.1) and the generalization error is presented for different number of hidden neurons (figure 3.3.2).

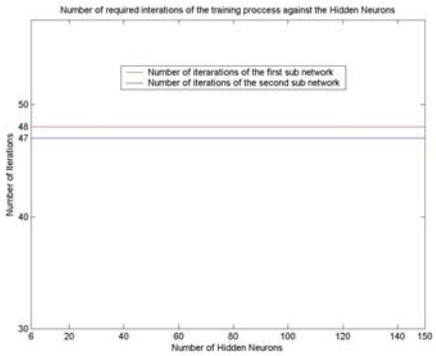

Figure 3.3.1. *Number of iterations per sub network for different number of hidden neurons*

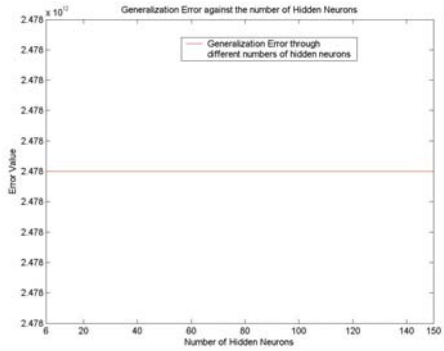

Figure 3.3.2. *Generalization error of validation data for different number of hidden neurons*

In order to increase the speech of the analysis process and the extraction of the statistic data the whole process run parallel in three computers. All the data were gathered in the end of the process and reconstructed. The results are plotting in the above figures. Different numbers of learning were used but still the generalization error couldn't be decreased.

An approach of improving the training process was ventured. The learning rate and the fault tolerance were decreased. The back propagation algorithm was unable to train the network more. The normal value of fault tolerance was set to 10^{-22}. It was impossible for the algorithm to reduce the error to the value of 10^{-23} for the second

sub network. The algorithm was trapped on a local minimal. In the next figure the training error is presented and the attempt of the back propagation algorithm to reduce the Error less than the value of 10^{-23}. The attempt of the back propagation algorithm to train the sub networks is shown in the figure 3.3.3 and 3.3.4. The algorithm is trapped and only the first 200 iterations of the training process are shown.

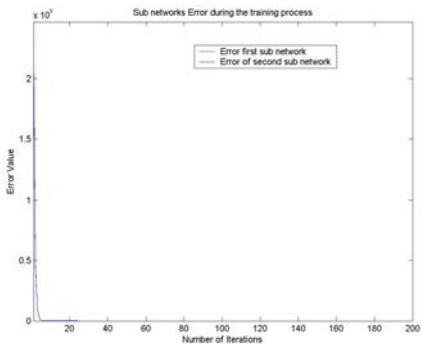

Figure 3.3.3. *Training Error the first 200 iterations of the trapped back propagation algorithm*

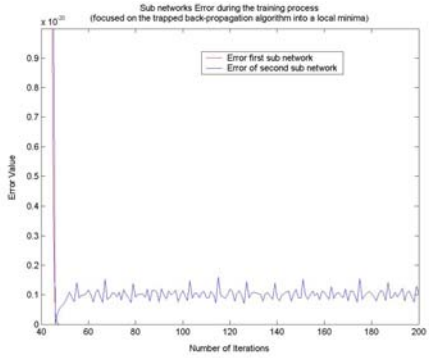

Figure 3.3.4 *The trapped point of the back propagation algorithm*

With more details the trapped point of the back propagation algorithm is focused on the next figure 3.3.4.

On the above figure a focused has been done on the trapped area of the back-propagation algorithm. Only the iterations from 40 to 200 are shown. The first sub network is trained after 46 iterations. On the other hand the second sub network can't be trained.

3.4 Hypothesis of unexpected network behaviour

Albeit the expectations for a successful research, the results reclined the initial aims and objectives. The network was unable to learn the patterns of the phonemes. One of the main reasons is focused on the problems of phonemes duration calculation and how they are located on the spectrogram space.

The phonemes duration was calculated by approaching the values. Another mathematical way had to be used in order to calculate the exactly duration of them. The calculations were completed by a way that the sum of the individual phonemes always to be the same with the word's duration. It is known that there is an overlap between phonemes on real speech samples. The overlap is different for each combination of phonemes. Some phonemes are effected more and some others less. It is depending on the sequence of them. If the word has many phonemes the duration of the overlap is a countable percentage of the whole word's duration. As a result the

same phoneme might have different duration in different words. Moreover the stress of the phonemes may vary the overlap duration. In the figure 3.4.1 the overlap of the simple word 'the' is presented. It seems that affects more the first phoneme. The uniform of the data in the overlap area looks similar to the second phoneme.

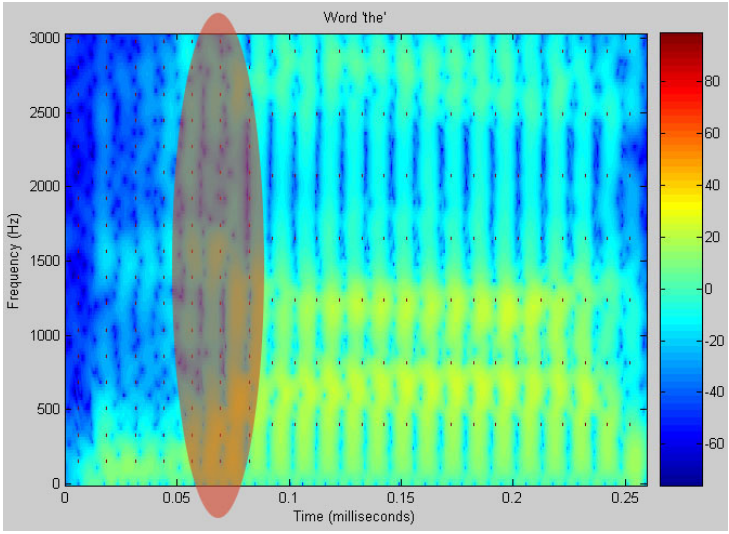

Figure 3.4.1. *Presenting the overlap between phonemes. The word 'the' is presenting*

The overlap area has been marked by a transparent ellipse. The uniform of the data of the spectrogram looks like the uniform of the second phoneme rather than the first ones. These data are about 40% of the first phoneme duration. These overlaps can cause a wrong matching phoneme and output vectors distributions.

Except the possible failure of the target vectors calculation and the analysis of the sound samples, another problem had to be faced. The training process of the network was another important reason for this behaviour of the system. All the input and output were rescaled within the range of zero and one. For small number on this range the square sum or different is decreased rapidly. The dimensionality of the output forced the error calculation function to set the difference of the error two times in square. That means that the different is actually decreased by sixteen times. In addition the training error couldn't reach values lower than 10^{-22}. Considering the decrease of the error by setting it twice to the power of two, it seems that the network hadn't trained enough. All the output data are very small numbers (after normalization process) and the error tolerance has to be much smaller. Unfortunately, as it has mentioned the back propagation algorithm was trapped on a local minima and it was impossible to train more the network. As the segmentation of the real speech samples, for the definition of phonemes wasn't successful, it might be possible

to find the same input vectors with different output vectors in some other words. That will cause confusion and it will reduce the possibilities of training process.

Moreover another important reason for the unexpected behaviour of the network is the input data. A text to an acoustic English dictionary was used. It is known that the number and the duration of phonemes are depending on the speaker. The dictionary that was used it has 39 different phonemes and 3 different stresses for some phonemes. That representation might not match with the sound samples that were created by the Microsoft's speech synthesis engine. Likewise it has proved that a word can be represented by many ways with phonemes without changing the pronunciation of it. Many different paths of phonemes sequence can lead to the same result. On this research only one path was followed and it was with a dictionary of 39 phonemes.

4. Conclusions

The whole analysis and research results weren't the expected ones. Considering the hypothesis of failure and the sensitivity of the sound data, it can be summarised that the whole task is harder that it was looking the beginning. The sound data are very sensitive. There is no any standard process for the speech analysis and there many different paths that can be followed. In case the input data of phonemes the acoustic representation and the recorded samples have to match exactly.

The whole analysis that was introduced on this research seems theoretical to be functional. There were many problems and the sample data wasn't the right ones. On the other hand, the problems that caused these results are now known and they can be faced.

One approach of facing the major problem of calculating the right duration of phonemes and matching the input vector with the right output data is to perform some steps of speech recognition process before the analysis that was introduced. By the speech recognition the different phonemes will be found. The analysis of sound can lead to the exactly calculation of the phonemes duration the matching of output and input data. In most of the cases a neural network can be trained and perform these tasks. These steps are accomplished by the speech recognition process in order to recognise phonemes, letters, syllables and finally the whole word [Yu H.]. Forwarding this output information to this methodology will be able to support or reject this approach.

In conclusion, we can say that during the whole research a new approach was implemented. The new approach was simulated and accomplished but the results couldn't be generalised for words that wasn't in the training set of data. The generalization error was too big comparing with the training one. The input vectors couldn't match the output patterns. There many problems with the initial data and the

calculation of them. A revised version of this approach has to be reconsidered with a better set of data.

5. References

Lemmetty S. (1999) “Review of Speech Synthesis Technology”. Master’s Thesis, Department of Electrical and Communications Engineering at University of Helsinki
<http://www.acoustics.hut.fi/~slemmett/dippa/thesis.pdf>

Pearson P. “Sound Sampling”.
http://www.hitl.washington.edu/scivw/EVE/I.B.3.a.SoundSampling.html

The American Radio Relay League (1991) “The ARRL Handbook for Radio Amateurs”. The League

Vlahabas I., Kefalas P., Bassiliades N., Refanidis I., Kokkoras F. and Sakellariou H. (2002) “Artificial Intelligence”. Gartaganis Publications.

Yiannakoudakis E. and Hutton P. (1987) “Speech Synthesis and Recognition systems”. Ellis Horwood Limited

Yu H. and Oh H. “A neural network using acoustic sub-word units for continuous speech recognition”. Department of computer science, KAIST (Korea Andvanced Institute of Science and Technology)
<http:// www.asel.udel.edu/icslp/cdrom/vol1/405/a405.pdf>

Zurada J. (1992) “Introduction to artificial neural systems”. West Publishing Company