

TEXT-TO-SPEECH CONVERSION WITH STAGED NEURAL NETWORKS

FABIO ARCINIEGAS **MARK J. EMBRECHTS**
Department of Decision Sciences and Engineering Systems.
Rensselaer Polytechnic Institute, Troy, NY 12180, USA

ABSTRACT

This paper presents a series of staged artificial neural networks (ANNs) for phoneme recognition for text-to-speech applications. Applying ANNs for phoneme mapping for text-to-speech conversion creates a fast distributed recognition engine. This engine not only supports the mapping of missing words in the database, but it can also reduce contradictions related to different pronunciations for the same word. The staged ANN presented in this work was trained based on the 2000 most common words in American English, resulting in a 97% accuracy. Performance metrics for the 5000, 7000 and 10000 most common words in American English were also estimated to test the robustness of the staged neural networks.

INTRODUCTION

This paper describes a staged artificial neural network (ANN) approach for text-to-speech conversion that can be applied to large texts and builds on the pioneering work of Sejnowski and Rosenberg (NetTalk Sejnowski (1987)). NetTalk uses a single NN to deal with all phoneme cases. The work presented in this paper is different in the sense that multiple stages of NNs were applied and that a different alignment structure for the mapping between letters-to-phonemes was used. The first NN stage distinguishes between single and dual phoneme cases (i.e., one letter is mapped to two phonemes). Then, in the second stage two different neural networks are used in parallel to deal with one and two-phoneme cases separately.

NetTalk introduced a slicing window where the letter to be mapped to phonemes is placed in the middle of the window (i.e., central window positioning, Arciniegas (2000)). In the alignment approach presented in this paper a new window positioning structure is introduced and labeled the Second Position Asymmetric Windowing (SPAW), where the number of spaces before and after the object letter (i.e., the letter which will be mapped to a phoneme) are not equal. It is called "Second Position" because the object letter is located in the second space from the middle of a central window (Figure 1). Second Position Asymmetric Windowing will be labeled "N-M SPAW" for notation purposes, where N and M refer to the number of spaces before and after the central window position. This concept will be explained in more detail later in this paper.

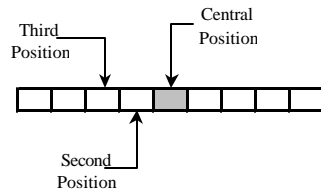


Figure 1. Windowing Positions

This paper is organized as follows: section I introduces the text-to-phoneme conversion approach, section II describes the staged NN model and discusses the NN performance, and in section III Self-Organizing Maps (SOMs) are applied for text-to-phoneme conversion and suggests additional information that could be presented to the ANN.

TEXT-TO-SPEECH CONVERSION

Three fundamental issues need to be addressed for text-to-phoneme mapping with staged neural networks: training datasets, window alignment, and context.

The 2000 Most Common Words in American English (MCWE) were selected for developing text-to-phoneme staged NNs and performance metrics were estimated for the 5000, 7000 and 10000 MCWE (See References, CMPD web page). The 1000 MCWE were used in order to compare our results with NetTalk (Sejnowski (1987)). We used the 1000 MCWE from the CUE practice set (See References), which is different from the 1000 most commonly words used by NetTalk, which were based on The Webster's Pocket Dictionary.

The Carnegie Mellon Pronouncing Dictionary (CMPD) was utilized for generating the windowed training dataset. CMPD is a publicly available web-based machine-readable pronunciation dictionary for North-American English that contains over 100,000 words and their phonetic transcriptions. The implemented CMPD phoneme set contained 39 phonemes, for which the vowels may carry lexical stress (0 for no stress, 1 for primary stress, and 2 for secondary stress). As of yet, voice stress-related features were not implemented. A 40th phoneme was added to represent the blank or punctuation marks. The staged networks were trained based on the 2000 MCWE.

Window alignment is important because a unique map from text to phonemes is needed to generate the training/test sets for the NNs. Words are sliced starting with the first letter in the window and shifted to the left until the whole word has been passed by. This implies that the number of patterns per word will be equal to number of letters in the word. Bullinaria (1997) considered issues related to the choice for the appropriate window size (Bullinaria considered only central windowing). A first issue relates to the proper choice for the window size in order to accommodate any long-range dependencies. Also, a large window sizes implies that many units and

connections would be vastly underutilized because of the prevalence of empty window spaces.

Different arrangements for central windowing and Second Position Asymmetric Windows (SPAW) alignments were investigated. Sometimes two different phonemes can be mapped from the same information window: this is considered an inconsistency. Looking at Figure 2a, it can be seen that different alignments can lead to inconsistencies between {"CAME", "CAMERA", "BECAME"}, and {"THOUGH", "THOUGHT"}. Only a 2-6 SPAW avoids any inconsistency. The main idea was to explore how much information (spaces) were needed in order to minimize the number of inconsistencies per arrangement. Inconsistencies were counted for different window representations of the objective letter for both central and SPAW alignments (Figure 2b). It was found that Second Position Asymmetric Windowing alignments lead to fewer inconsistencies. A SPAW representation requires also less information (i.e., shorter window size).

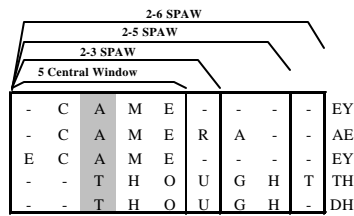


Figure 2a. Some Examples of Inconsistencies.

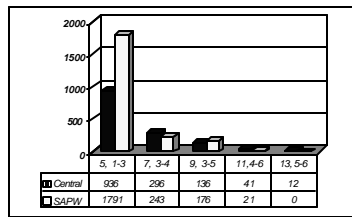


Figure 2b. Inconsistencies per Windowing Class.

Pronunciation of a particular letter or string of letters depends on the context. Therefore, context information should be used to generate pronunciation rules with a realistic generalization of unknown words and non-words (Bullinaria (1994)). Context related issues were not addressed in this study.

THE STAGED NEURAL NETWORK MODEL

Two different letter representations for the inputs to the ANNs were analyzed in Arciniegas (2000): a continuous and a categorical letter representation. The categorical representation resulted far better than the continuous input so that it was used in all NNs. The categorical representation assigns to each window space a set of 26 binary values, one for each letter of the English alphabet (i.e., a window with 7 spaces results in 182 inputs for the NN). For the phoneme output representation 40 categorical values were used (one for each phoneme and the 40th neuron represents the blank space). For the Backpropagation ANN, output values were encoded as 0.1 or 0.9 to avoid saturation problems. For the phoneme output representation 40 categorical

values were used (one for each phoneme and the 40th neuron represents the blank space). Output values were encoded as 0.1 or 0.9 to avoid saturation problems.

Initially single feedforward NNs (with one or two hidden layers) were trained on 80 percent of the 2000 words considered and tested on the remaining 20 percent of the patterns. This results in a training set of 10,251 patterns when all the possible window positions are considered. For the set of phonemes used in this study, 0.83% of the samples were cases in which the match between letters and phonemes was not one-to-one (single phoneme case) but one-to-two (dual phoneme case).

Different NN configurations (i.e., windowing, number of neurons in hidden layer, and number of hidden layers) were tried to obtain the best model for the text-to-phoneme mapping for both stages. Different combinations of inputs, alignment structures, size, and outputs configurations were analyzed for each NN. For the first stage, all window sizes were tested. However, for the ANN for the single phoneme case in the second stage, only large windows were used (4-6 and 5-6 SPAW and 11 and 13 central windowing). For all NNs training was halted using an early stopping criterion.

The first stage ANN initially utilized two categorical output neurons with target values [0 1] or [1 0] depending whether the outcome was a single or dual phoneme. The best NN employed a 1-4 SPAW window and one hidden layer with 11 neurons. It was able to recognize 100% and 67% of single and dual cases, respectively. A second approach used 40 neurons in the output layer. For this approach a 1-5 SPAW and a NN with two hidden layers (43 and 67 neurons, respectively) was able to reach 100% recognition for both single and dual phoneme cases.

For the second stage different ANNs with different window sizes were considered as well. Based on the results obtained for the first stage, only one output layer with 40 neurons and categorical inputs were used for the second stage. For the single phoneme case ANN, a 4-7 SPAW and two hidden layers was large enough to accommodate all long-range dependencies. Different numbers of neurons were tried for both layers. A weak optimum was found with 67 and 91 neurons for the first and second hidden layer, respectively. This ANN achieved 97% accuracy for a test set using 20 percent of the 2000 MCWE. The best net for the dual phoneme case used a 3-5 SPAW and two hidden layers (with 43 and 67 neurons, respectively) and achieved 100% accuracy.

BENCHMARK METRICS

Sejnowski, and Rosenberg trained NetTalk using the 1000 most commonly occurring words from the Webster's Pocket Dictionary based on frequency counts in the Brown corpus. The best performance achieved was 98% on the 1000 word corpus, and 80% and 91% without and with additional training on the 20,012 words on the Webster's corpus, respectively. The word corpus used in this paper is based on the CMPD dictionary from Carnegie

Mellon University and other word frequency counts referred to in the Carnegie Mellon's web site, which are different from the ones used by NetTalk. For the 1000 MCWE, approximately 200 words are different and recognition of 99% was reached.

In order to test the robustness of the staged NN model, performance metrics are compared for the 2000, 5000, 7000 and 10000 MCWE obtained for the LOB and ACL_DCI corpus (See references) (Table 1). For the first staged NN (i.e., simple or dual phoneme distinction), the recognition level was 100% for the 2000 and 5000 MCWE. However, for the 7000 and 10000 MCWE the first stage performance was 99% due to the presence of a new phoneme case. Once that case was introduced into the training process, the recognition level was 100% again. Also, many of the new words do not contain dual phoneme cases, which contributes to the robustness of the first stage of the model.

Table 1. Performance Results.

<i>Training Set</i>	<i>Performance over the Most Common Words in American English</i>			
	2000	5000	7000	10000
2000	97%	94%	91%	85%
5000	(98%)	95%	92%	84%
7000	(94%)	(91%)	89%	82%

Note: Corpus and word frequency counts were obtained from the CMPD, and the LOB and ACL_DCI corpus.

For the second stage single phoneme case NN there is a decrease in the level of recognition from the ANN trained on the 2000 MCWE, resulting in 94%, 91% and 85% recognition for the 5000, 7000 and 10000 MCWE, respectively. A decrease in the recognition levels was to be expected at a certain point due to the appearance of new exotic words with complex structures. This decrease became significantly when the training set exceeds the 7000 MCWE. One attempt to increase the recognition level was to train the NNs with the 5000 MCWE and measure the recognition rate on the 7000 and 10000 MCWE. It was found that using the 5000 MCWE does not improve the recognition level significantly (95, 92% and 86% for the 5000, 7000 and 10000 MCWE). Moreover, when using the 7000 MCWE for the training set the recognition level were 89% and 82% for the 7000 and 10000 MCWE, respectively.

New words appearing in American English when the frequency increases beyond the 7000 MCWE have either Spanish, Chinese, French and other language roots or complex structures. Because of differences from traditional English grammar and syntax, and the fact that people try to pronounce these new words as they sound in the native language, these words present a challenge to any text-to-speech model. It also is worth noting that the word corpus we used contained more slang, which should be taken into account when comparing with NetTalk. However, for more robust commercial

applications, the use of a word corpus containing slang is actually more appropriate.

CONCLUSIONS

This paper described the successful implementation for text-to-speech conversion with a two-staged neural network. It is novel in the sense that staged NNs were used and a better window alignment structure (SPAW) was applied. This staged approach first recognizes whether the letter shown has to be mapped into a single (one-to-one) or a dual (one-to-two) phoneme case and does the actual phoneme matching in the second stage. For the 2000 MCWE, 100% recognition in the first stage was obtained. In the second stage, 97% and 100% recognition was reached for the single and dual phoneme cases. Good recognition levels (91%) can be achieved for up to 7000 of the MCWE. For more than 7000 words it was found that many of the new words either do not belong to the American English (i.e., are exotic) or show complicated structures and are therefore hard to learn by the ANN.

REFERENCES

- Arciniegas, F., and Embrechts, M., 2000, "Artificial NNs (ANNs) for Phoneme Recognition for Text-to-Speech Applications," *Proceedings of the 2000 IEEE-INNS-ENNS International Joint Conference on NNs*, Como, Italy.
- Bullinaria, J. A., 1994, "Internal Representations of a Connectionist Model of Reading Aloud," *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*, pp. 84-89.
- Bullinaria, J. A., 1995, "NN Learning from Ambiguous Training Data," *Connection Science*, Vol. 7, pp. 99-122.
- Bullinaria, J. A., 1997, "Modeling Reading, Spelling, and Past Tense Learning with Artificial NNs," *Brain and Language*, Vol. 59, pp. 236-266.
- Colheart, M., Curtis, B., and Atkins, P., 1933, "Models of Reading Aloud: Dual-Route and Parallel-Distributed-Processing Approaches," *Psychological Review*, Vol. 100, pp. 589-608.
- Patel, M., 1996, "Using Neural Nets to Investigate Lexical Analysis," *Proceedings of PRICAI'96: Topics in Artificial Intelligence*, pp. 241-252.
- Seidenberg M., and McClelland, J., 1989, "A Distributed, Developmental Model of Word Recognition and Naming," *Psychological Review*, Vol. 96, No. 4, pp. 523 - 568.
- Sejnowski, T. J., and Rosenberg, C. R., 1987, "Parallel Network that Learn to Pronounce English Text," *Complex Systems*, Vol. 1, No. 1, pp. 145-168.
- www.speech.cs.cmu.edu/cgi-bin/ The CMPD Pronouncing Dictionary. Carnegie Mellon University.
- http://gopher://gopher.sil.org/11/gopher_root/linguistics/info/. LOB and ACL_DCI corpus frequency list.
- http://www.uri.edu/comm_service/cued_speech/1000most.html. CUE practice with the 1000 most common words.