Neural Network Based Speech Synthesizer: A Preliminary Report

James A. Villarreal
Artificial Intelligence Section/FM7
Lyndon B. Johnson Space Center (JSC)
NASA
Houston, Texas

Gary McIntire
Advanced Systems Engineering Dept.
Ford Aerospace
Houston, Texas

Abstract

The growth and development of our goals in space utilization will reach out to utilize every possible technology. Artificial Neural Systems (ANS) is a newly emerging technology which has already indicated a potential solution to many space engineering problems. A particularly interesting feature of ANS's is their ability to construct vital generalizations or inferences from sample data without the need for conventional programming. In order to evaluate ANS's, the Artificial Intelligence Section is conducting several initial projects implementing ANS's, developing a dedicated ANS workstation, and developing applications to assist with the immigration of this technology.

This paper will describe the neural net based speech synthesis project. The novelty is that the reproduced speech was extracted from actual voice recordings. In essence, the neural network (NN) learns the timing, pitch fluctuations, connectivity between individual sounds, and speaking habits unique to that person. The parallel distributed processing network used for this project is the generalized backward propagation network which has been modified to also learn sequences of actions or states given a particular plan.

Introduction

An inherent nature to human behavior is its dependency on serial order. Our learning methods, speech, and chain of thoughts all follow a succession of events. The parallel distributed processing network being used by this project is a generalized backward

propagation network with the usual input, hidden, and output layers but with an added layer which learns a sequence of actions which are produced in a learned order given a particular plan. In describing the NN based speech synthesizer, the technique used to formulate the various characteristics in a persons speech is first discussed. An introductory explanation of the generalized backward propagation network along with its modification to learn sequences will also be provided. This will be followed with a description of the various components of the NN and their relation to the speech parameters.

Linear Predictive Coding

Humans produce speech by sending pulses of air (the vocal chords) through a resonating cavity (the vocal tract) which is constantly changing shape (velum, tongue, lips, and nasal cavity) to produce a wide range of sounds. The human vocal tract can then be modeled as a time-varying linear filter. The filter can be excited by a series of pulses to represent the pitch or the voiced segments of speech and white noise to represent the unvoiced segments of speech.

Linear prediction coding (LPC) is a technique which efficiently represents the speech signals in terms of a small number of slowly varying parameters[1]. Linear predictive analysis converts the combined spectral contributions of the glottal flow within a pitch period, the vocal tract, and the radiation at the lips into a single recursive (all-pole) time-varying filter. The transfer function in the complex Z domain of the LPC filter can be written as:

$$H(z) = \frac{GAIN}{1 - \sum_{k=1}^{p} a_k z^{-k}}$$

Thus the linear filter is completely specified by a scale factor GAIN and p coefficients $a_1, a_2, \ldots, a_p$. The linear filter has p poles which are determined on factors such as the length of the vocal tract, the coupling of the nasal cavities, the place of the excitation, and the nature of the glottal flow function. Nineteen poles and a 16 KHz sampling rate were selected to compute the LPC coefficients for this

project. Speech recordings were taken with the MIT developed software package: Speech Phonetic Research Environment (SPIRE) [2].

Ordinarily, LPC is used to compress speech on bandwidth limited communications channels. LPC analysis produces a set of all pole filter coefficients with an added "residual". The resultant residual is the "true" excitation source. To compress speech, this residual is usually discarded and replaced with an approximation of the gain, and an indicator of whether the segment is voiced or unvoiced. For voiced segments of speech the pitch rate is also expected. The approximations for the voiced/unvoiced decisions and pitch rate explain why the reconstructed signal sounds mechanized. At this point, it is important to understand that if the filter coefficients produced by LPC analysis are re-excited by the "true residual", then the waveform can be reconstructed without any degradation.

Parallel Distributed Processing

The parallel distributed processing network used in this project is a form of the "generalized delta-rule" known as "back propagation of error"[3,4]. In summary, the NN consists of a network of unidirectional processing units connected by weights. The state of each processing unit is determined according to the activation function:

$$x_j = \emptyset \left(\sum_{i=1}^{n} w_{ji} x_i + ¥_j \right)$$

where x_i is the activation function of the ith unit, w_{ji} is the weight from the ith unit to the jth unit, $¥_j$ is a bias associated with the jth unit, and n is the number of units in the network. A processing node whose output is fedback onto itself is termed *recurrent*.. This recurrence is the technique which provides a NN the capability to learn a sequence of events given a particular plan[5]. A NN with an input, hidden, state, and output layer are depicted in figure 1.

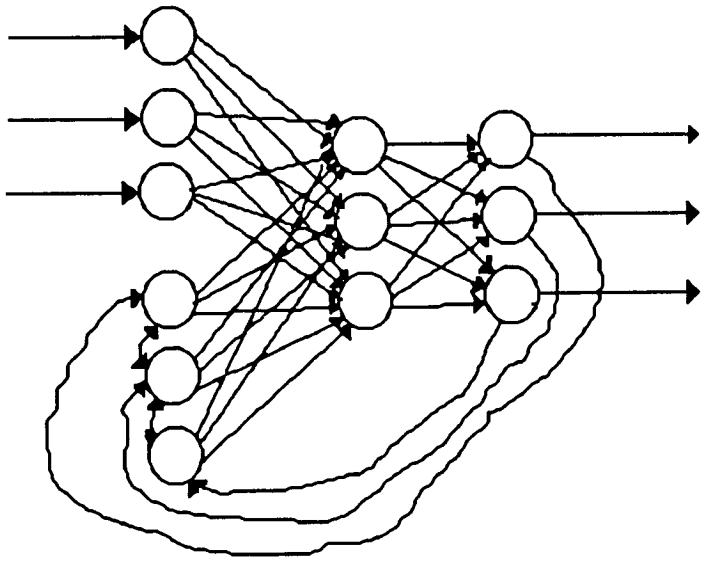

Figure1. Network showing input, hidden, output, and state layers.

Speech Synthesis Using a NN

The network produced here is similar to that produced by Sejnowski[6]. Unlike NETtalk which generates the phonetic translation for an English word, this synthesizer will produce the appropriate sequence of LPC coefficients and the spectrum of the residual for an English input.

To generate the data for the net, phrases which are rich in the English speech sounds were selected. For instance, for a phrase rich in the sounds [i] as in *see*, the phrase "Eva likes green peas" was selected. Other examples include the phrases "Ethel held that you lose your friends if you permit them to be in debt." for the vowel sound [e] as in *help* and "Alice planted three rows of asters." for the vowel sound [æ] as in *bat*. The attempt here is to generate a database with many examples of the various phonetic sounds in context with other sounds to capture the varying filter characteristics as phonemes are merged. After recording a phrase, the laborious task of segmenting and labeling the time based waveform with the appropriate letter follows. Having completed the segmentation and labeling of a phrase, the LPC analysis and spectrum for the entire phrase is computed. Several more phrases and 500 of the most commonly used words in the English language are still require[7, 8].

The NN is presently configured such that the input layers consist of 5 slots for the alphabetic characters and 2 special characters ("." and "?"). Training takes place by shifting a character

into the input layer at each cycle. The output layers consist of the 19 LPC coefficients and the spectrum of the true residual at a particular instant in time. Because the network is also learning sequences, each letter is represented by a series dependent set of 19 LPC coefficients and 256 spectral lines between 0 and 8kHz to represent the residual of the waveform. An added output layer is trained as a decreasing counter of the number of LPC sets for each letter. This output layer is monitored for a zero count to shift in a new letter.

Conclusion

The synthesizer is still under development, however, preliminary experiments have provided positive indications that the procedure will work. The experiments were conducted on the Symbolics computer. Due to the large size of this NN and the large data base, significant computer time is required. Currently under development by the AIS is a transputer based NN workstation. This workstation is being developed to satisfy the processing speed requirements inherent to most NN problems[9].

Acronyms

ANS	artificial neural systems
kHz	kilo Hertz (cycles per second)
LPC	linear predictive coding
NN	neural network

References

1. Rabiner, L. R., and R. W. Schafer. "Digital Processing of Speech Signals." New Jersey: Prentice Hall, 1978, pp. 396-453.

2. Kassel, R. H. "A User's Guide to SPIRE." Speech Communications Group Research Laboratory of Electronics, Massachusetts Institute of Technology, 1986.

3. Rumelhart, D. E., Hinton, G. E., and Williams, R. J. "Learning Internal Representations by Error Propagation." (Tech. Rep. 8506). La Jolla: University of California, San Diego, Institute for Cognitive Science, 1985.

4. McClelland, J. L. and Rumelhart, D. E. "Distributed Memory and the Representation of General and Specific Information", Journal of Experimental Psychology: General, Vol. 114, No. 2, pp. 159-188, 1985.

5. Jordan, M. I. "Serial Order: A Parallel Distributed Processing Approach." (ICS Report 8604), La Jolla: University of California, San Diego, Institute for Cognitive Science, 1986.

6. Sejnowski, T. J. and Rosenberg C. R. "NETtalk: A Parallel Network that Learns to Read Aloud." Johns Hopkins University, 1986.

7. Eisenson, J. "The Improvement of Voice and Diction." New York: The Macmillan Company, 1958.

8. Thorndike, E. L. and Lorge, I. "The Teachers Word Book of 30,000 Words." New York: Teachers College Press, 1972.

9. Mcintire, G., Villarreal, J., Baffes, P., and Rua, M. "Design of a Neural Network Simulator on a Transputer Array." Houston: Presented at Space Operations Automation and Robotics Workshop, 1987.