A Methodology for Adaptive Speech Recognition Systems and a Development Environment

Akbar Ghobakhlou and Nikola Kasabov

Knowledge Engineering and Discovery Institute
Auckland University of Technology, PO Box 92006,
Auckland, New Zealand
E-mail: Akbar,nkasabov@aut.ac.nz
Web: www.aut.ac.nz/

Abstract

A methodology and environment for building adaptive speech recognition systems is presented. The development environment is designed for isolated word recognition systems. A small speech recognition system is developed for a home automation system. It is demonstrated that one of the available connectionist models within this environment called ECoS, can accommodate new speakers and expand its output dimensions to recognise new words while maintaining its previously learned knowledge.

1 Introduction

Artificial neural networks (ANN) have been intensively applied to speech recognition task throughout the past decades [5]. Here they are applied to adaptive speech recognition systems which are systems that can adapt to new pronunciation and recognise new words during their operation [7].

The main principle behind the notion of adaptive systems is their ability to modify themselves to account for new data. Several paradigms have been suggested that are useful for this task, including Resource Allocation Networks (RAN) [6], Nearest-Neighbour MLPs (NN-MLP) [8], Cascade-Correlation Learning Architecture [1] and Evolving Connectionist Systems (ECoS) [4, 2, 3].

This paper describes the development of an adaptive speech recognition methodology and environment for creating isolated word recognition systems. This model can be expanded to recognise new words and adapt to new speakers.

2 Adaptive Speech Recognition Methodology and a Development Environment

This project provides an environment for developing adaptive speech recognition systems. It consists of several modules, including, signal processing module, word recognition design module , validation of the module and adaptation of the module.

The signal processing module includes most common techniques used in speech processing (eg. FFT, MSCC, DCT, PCA).

The word recognition design module includes evolving connectionist systems such as ECoS, Zero Instruction Set Computing(ZISC) [9], statistical methods Hidden Markov(HMM) Models and Support Vector Machine(SVM) to construct an architecture for an adaptive speech recognition system. The recognition engine is adaptable to new speakers and can expand its output domain (ie. vocabulary expansion).

At present, an expert will determine the best features and recognition model to create a word recognition system. In the future development, the optimal features and recognition model should be suggested by this environment.

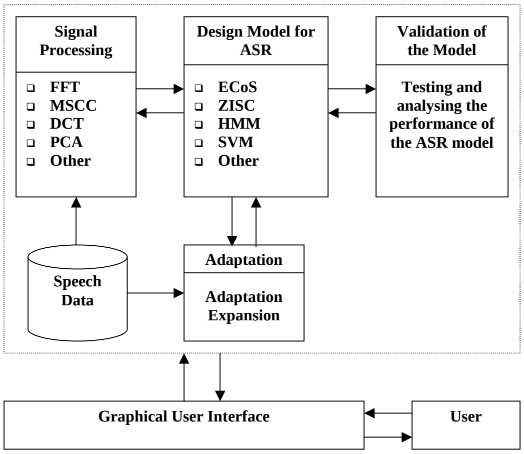
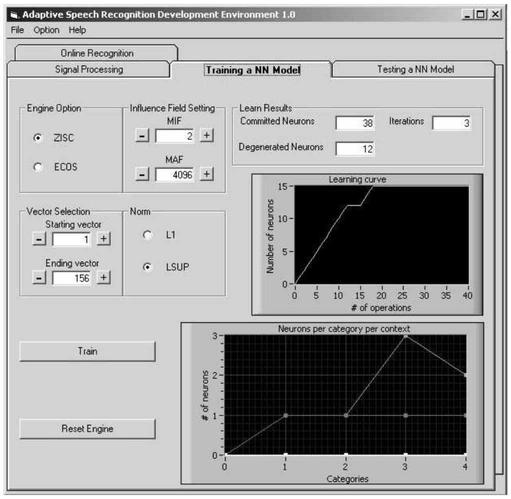


Figure 1: A schematic view of the adaptive speech recognition development environment

A graphical user interface is designed to allow interaction between the various modules of the system. It facilitates visualisation of modules within this project. The environment enables users to build their own customised speech recognition systems, see Figure 2. SECoS is one of the models facilitated by this environment and described in the following section.



Engine Option: ● ZISC ○ ECOS

Influence Field Setting: MIF 2; MAF 4096

Learn Results: Committed Neurons 38; Iterations 3; Degenerated Neurons 12

Vector Selection: Starting vector 1; Ending vector 156

Norm: ○ L1 ● LSUP

Figure 2: Adaptive speech recognition development environment's graphical user interface

3 The Simple Evolving Connectionist Structure

The **S**imple **E**volving **Co**nnectionist **S**ystem (SECoS) is a minimalist implementation of the ECoS paradigm. It was created as a simpler version of EFuNN. There are several advantages to using SECoS over EFuNN. Firstly, their much simpler architecture means they are easier to understand and analyse. Secondly, their unfuzzified input space is of a lower dimensionality than a corresponding EFuNN (which always have at least two condition nodes attached to each input node, thereby doubling the dimensionality of the input space), which allows the SECoS to model the training data with fewer nodes in the evolving layer than an equivalent EFuNN.

3.1 The SECoS Architecture

Figure 3 is a simplified graphical representation of the SECoS architecture. A SECoS consists of only three layers of neurons, the input layer, with linear transfer functions, an evolving layer based upon the rule layer of the Evolving Fuzzy Neural Network (EFuNN) [4] model, and an output layer with a simple saturated linear activation function . The evolving layer activation is calculated as with the EFuNN, with the exception of the distance measure D_n being calculated as the normalised Hamming distance, as shown in Equation 1:

$$D_n = \frac{\sum_{i}^{I} \mid E_i - W_i \mid}{\sum_{i}^{I} \mid E_i + W_i \mid} \tag{1}$$

where:
I is the number of input nodes in the SECoS,
E is the input vector,
W is the input to evolving layer weight matrix.

The SECoS architecture is similar to the Zero Instruction Set Computer (ZISC) architecture [9]. However, ZISC is based on RBF ANN and requires several training iterations over input data.

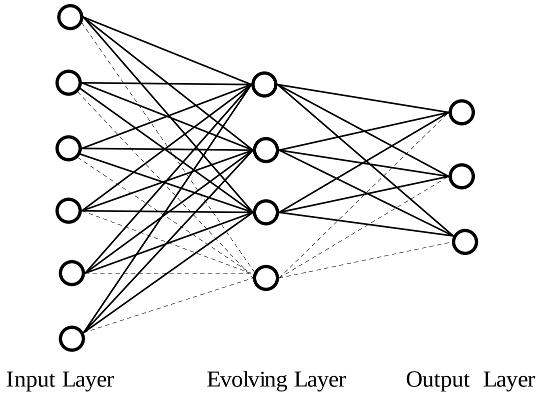


Figure 3: A simplified and exemplified diagram of a SECoS network. The dotted lines represent the creation of a new node in the evolving layer

4 Case Study: Voice Activated Home Automation System

Home automation systems refer to systems which permit remote control of electronic devices in the immediate surroundings. A person can turn lights, fan, and television on and off. Essentially any aspect of the environment can be controlled depending upon the system's complexity. The aim of this case study is to develop a small speech recognition system to control certain devices at various locations of a private home. Figure 4 illustrates the layout of the commands to operate certain home appliance. For instance, the command sequence "kitchen, fan, on", turns on the fan in the kitchen.

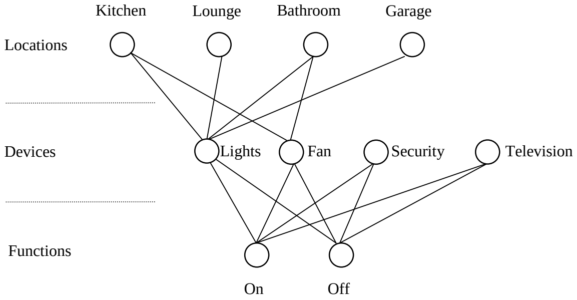


Figure 4: An overview of voice activated home automation command structure

A SECoS is employed to create an isolated word recognition system. The following section describes the experiments performed to evaluate the performance of the SECoS on recognition, adaptation and vocabulary expansion.

5 Experimental Procedure and Results

The experiments were carried out in two distinct phases. In the first phase, a SECoS is trained to recognise 10 voice commands. In the second phase, the output space of the SECoS was expanded to recognise 5 additional words as alternative voice commands.

The speech data was recorded in a quiet room environment to obtain clean speech signals. For the sake of this case study, speech data collected from 6 native New Zealand English speakers. The speech was sampled at 22.05 kHz and quantised to a 16 bit signed number. Each word was uttered 6 times with distinct pauses in between. Each of these words was then carefully manually segmented and labelled.

In the first phase, speech data was prepared for the 10 basic voice commands. The speech data used in these experiments obtained from two groups of speakers; group A (2 males and 2 females) and group B (1 male and 1 female). The data from each groups were divided into two sets; training set A, testing set A, training set B and testing set B.

Training set A with 160 examples was obtained from 4 utterances of each word in group A. The remaining 2 utterances were used in the testing set A for a total of 80 examples. Training set B with 40 examples was obtained from 2 utterances of each word in group B. Testing set B with 80 examples was obtained from the remaining 4 utterances of each word in group B.

Spectral analysis of the speech signal was performed over 20 msec with Hamming window and 50% overlap, in order to extract Mel Scaled Cepstrum Coefficients (MSCC) as acoustic features. Discrete Cosine Transformation (DCT) was applied on the MSCC of the whole word in the following manner.

For an m frame segment, DCT transformation will result in a set of m DCT coefficients. This sequence is truncated to achieve a fixed-size input vector consisting of $20 \times d$, where d is the dimensionality of the feature space.

5.1 Training SECoS

A SECoS was initialised with 100 nodes in the input layer and 10 nodes in the output layer (one for each word). In phase one of the experiments the SECoS was trained on the training set A. There were 33 nodes created during the training period. The trained SECoS was then tested on training set A and both testing sets A and B. The SECoS performance on training set A was 100%. Table 1 shows the performance of the SECoS on testing sets A and B.

Table 1: Performance of SECoS on both Testing Set A and Testing Set B of the voice commands

	Testing Set A		Testing Set B	
Words	Positive Accuracy	Negative Accuracy	Positive Accuracy	Negative Accuracy
On	100.00	100.00	100.00	91.67
Off	100.00	100.00	75.00	95.83
Lights	100.00	100.00	75.00	100.00
Fan	100.00	100.00	100.00	98.61
Security	100.00	100.00	100.00	100.00
Television	100.00	100.00	100.00	100.00
Kitchen	100.00	100.00	100.00	100.00
Lounge	100.00	100.00	50.00	93.06
Bathroom	100.00	100.00	75.00	100.00
Garage	100.00	100.00	25.00	98.61
Total	100.00	100.00	80.00	97.78

The second phase of the experiment was designed to expand the current SECoS to recognise 5 new commands. The same procedure was applied to prepare a training and testing sets. For the 5 additional words, there were a total of 80 examples in training set A, 40 examples in testing set A, 20 examples in training set B and 40 examples in testing set B. The structure of the existing SECoS was expanded to accommodate 5 additional outputs. The expanded SECoS was further trained on the training set A of the additional words. 38 new nodes were added to the evolving layer of the expanded SECoS during the training process. Table 2 illustrates the performance of the expanded SECoS on the original and additional testing set A and testing set B.

To test the adaptation vs forgetting capabilities of SECoS, the testing set B (ie: new speakers), the trained SECoS was additionally trained on training set B. 52 additional nodes were added to the new SECoS to accommodate the variations in the new speakers. The performance of the adapted SECoS on its training set B was 100%. The adapted SECoS was then tested on the original training (training set A) and testing sets A and B. Performance of the adapted SECoS on training set A was remained unchanged. Table 3 illustrates the performance of the SECoS on testing sets A and B after adaptation on new speakers. SECoS retained its recognition ability on old data while achieving

Table 2: Performance of the expanded SECoS on both initial and additional commands

	Testing Set A		Testing Set B	
Words	Positive Accuracy	Negative Accuracy	Positive Accuracy	Negative Accuracy
On	87.50	100.00	37.50	97.32
Off	100.00	100.00	87.50	96.43
Lights	100.00	100.00	62.50	100.00
Fan	100.00	100.00	50.00	98.21
Security	100.00	100.00	87.50	100.00
Television	100.00	100.00	100.00	100.00
Kitchen	100.00	100.00	100.00	99.11
Lounge	100.00	100.00	62.50	94.64
Bathroom	100.00	100.00	87.50	100.00
Garage	100.00	100.00	25.00	100.00
Stop	100.00	100.00	50.00	93.75
Start	100.00	100.00	37.50	99.11
Disarm	100.00	100.00	87.50	100.00
Arm	100.00	99.11	100.00	89.29
TV	100.00	100.00	75.00	100.00
Total	99.17	99.94	70.00	97.86

excellent adaptation on new data.

Table 3: Performance of the expanded SECoS on both initial and additional commands after adaptation on new speakers

	Testing Set A		Testing Set B	
Words	Positive Accuracy	Negative Accuracy	Positive Accuracy	Negative Accuracy
On	87.50	100.00	87.50	99.11
Off	100.00	100.00	100.00	100.00
Lights	100.00	100.00	100.00	100.00
Fan	100.00	98.21	62.50	99.11
Security	100.00	100.00	100.00	100.00
Television	100.00	100.00	100.00	100.00
Kitchen	100.00	100.00	100.00	100.00
Lounge	100.00	100.00	100.00	98.21
Bathroom	100.00	100.00	100.00	100.00
Garage	100.00	100.00	75.00	100.00
Stop	100.00	100.00	87.50	98.21
Start	100.00	100.00	75.00	99.11
Disarm	75.00	100.00	100.00	100.00
Arm	100.00	99.11	75.00	96.43
TV	100.00	100.00	100.00	100.00
Total	97.50	99.82	90.83	99.35

6 Conclusions and Future Research

This paper describes a methodology and a development environment for analysing and building adaptive speech recognition systems. It can be applied in various applications such as, voice activated wheel chairs, robotic control, etc.

Experiments carried out in the case study explored the performance of the SECoS locally learning algorithm. Although a small set was used to train the SECoS, it was demonstrated that SECoS is capable of expanding its vocabulary and accommodating new speakers. It was also shown that SECoS maintained its previously learned knowledge after vocabulary expansion and speaker adaptation.

In a future development, feature and model selection along with parameter selection of the model(e.g. SECoS) will be automated.

7 Acknowledgement

This research is funded by the NZ Foundation for Research, Science and Technology under the grant AUT02001.

References

[1] S. E. Fahlman and C. Lebiere, "The Cascade-Correlation Learning Architecture," in D. S. Touretzky (ed.), **Advances in Neural Information Processing Systems**, Denver 1989: Morgan Kaufmann, San Mateo, 1990, vol. 2, pp. 524–532.

[2] N. Kasabov, "Evolving Fuzzy Neural Networks - Algorithms, Applications and Biological Motivation," in T. Yamakawa and G. Matsumoto (eds.), **Methodologies for the Conception, Design and Application of Soft Computing**, World Scientific Publishing Co, 1998, pp. 271–274.

[3] N. Kasabov, "Evolving connectionist systems:Methods and applications in bioinformatics, brain study and intelligent machines," **Springer, London**, 2002.

[4] N. Kasabov, "Evolving Fuzzy Neural Networks for Supervised/Unsupervised Online Knowledge-Based Learning," **IEEE Transactions On Systems, Man and Cybernetics, Part B: Cybernetics**, vol. 31, no. 6, pp. 195–202, December 2001.

[5] R. P. Lippmann, "Review of Neural Networks for Speech Recognition," **Neural Computation**, vol. 1, no. 1, pp. 1–38, 1989.

[6] J. Platt, "A Resource-Allocating Network for Function Interpolation," **Neural Computation**, vol. 3, no. 2, pp. 213–225, 1991.

[7] D. Saad, "On-line learning in neural networks," **London, U.K.: Cambridge University Press**, 1998.

[8] Q. Zhao and H. T., "Evolutionary Learning of Nearest-Neighbour MLP," **IEEE Transactions on Neural Networks**, vol. 7, no. 3, pp. 762–767, 1996.

[9] ZISC-Manual, "Zero Instruction Set Computing (ZISC), 2000, http://www.silirec.com," .