Connected Speech Recognition with an Isolated Word Recognizer

Perera K.A.D., Ranathunga R.A.D.S., Welivitigoda I.P, and Withanawasam R.M.
Department of Computer Science and Engineering,
University of Moratuwa, Sri Lanka
Email: superdilan_returns@yahoo.com
No. 64, Woodward Rd, Galle, Sri Lanka
Email: lady_radsu@yahoo.com
‘Jayanthi’, Uduwa South, Horana, Sri Lanka
Email: indunil12@yahoo.com
‘Sampath’, Poorwarama Rd, Bataduwa, Galle, Sri Lanka
Email: rasikamw@yahoo.com
“Withana”,Dorape,Angulugaha,Galle, Sri Lanka

Abstract—This paper is based on an on-going research work and it explains how a speaker independent isolated word recognizer could be incorporated in a connected word recognition system. The recognizer used here is an Artificial Neural Network (ANN), which is presented as an alternative to the traditional statistical approaches to speech recognition. Here, more weight is given to describe word boundary detection and splitting the input speech signal into separate words. Another area that is being explained here is the concept of language modeling. These two are the areas which make the connected word recognition distinct from isolated word recognition. The main contribution of this paper is the attempt to extend a simple isolated word recognizer into a connected speech recognizer, by introducing sophisticated signal pre-processing and fine-tuning the recognizer output using a language model. Here, the language model was implemented in a very unique way, extending the concept of Hidden Markov Model (HMM), which is a completely new idea introduced.

I. INTRODUCTION

CONNECTED-SPEECH recognition is a technique that is sometimes used in continuous-speech applications. In this technique, the sentence is decoded by patching together models built from discrete words and matching the complete utterance to these concatenated models. The system usually does not attempt to model word boundary allophonic effects, nor sloppy intra or inter-word articulation [1 pg610]. There is an implicit assumption that, while distinct boundaries cannot be located among words, the words are reasonably well articulated. The accuracy of the system could be increased when probabilistic relationships among words (syntax) are known.

II. BACKGROUND

A. Word Boundary Detection

When trying to incorporate an isolated word recognizer to recognize connected speech, the most crucial part is the separation of the incoming signal into the constituent words. This is extremely difficult in noisy environments. It is particularly troublesome with words that begin or end in low-energy phonemes or with words which have a silence before release [1 pg 611]. Some speakers also habitually allow their words to trail off in energy. Others tend to produce bursts of breath noise at the end of the words [3].

In the conventional endpoint detection algorithms, the short-time energy or spectral energy is usually used as the primary feature parameters with the augmentation of zero-crossing rate, pitch and duration information. But these features become less reliable in the presence of nonstationary noise and various types of sound artifacts. Some other algorithms utilizing the noise adaptive thresholds have been also proposed [3-5].

B. Feature Extraction

A speech waveform contains a lot of information, making it difficult to use the raw signal directly for recognition. So, feature extraction methods are used to convert the speech waveform to some type of parametric representation which consists of a lower information rate. One of the most common and successful methods is the Linear Prediction Coding (LPC). LPC provides a good model of the speech signal. This is especially true for the quasi steady state voiced regions of speech in which the all-pole model of LPC provides a good approximation to the vocal tract envelope [6]. The way in which LPC is applied to the
analysis of speech signals leads to a reasonable source-vocal tract representation.

C. Artificial Neural Networks in Speech Recognition

The concept of Artificial Neural Network (ANN) is gaining a lot of attention in the world of computing. An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information [7]. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems.

A lot of research has been carried out to combine ANN computing with conventional algorithms used in speech recognition. Some of them are Dynamic Time Warping, Hidden Markov Models and Viterbi Search. The ANN contribution to these techniques is principally to serve as an alternative computing structure for carrying out the necessary mathematical operations [1 pg841].

D. Language Modeling

Presently most of the high-accuracy speech recognition systems incorporate a language model in its operations. This is used to implement a model of correspondence between acoustic signals and words. It is concerned on the sequence of the words that are possible in the natural language. Typically, the language models used in speech recognition systems pay little attention to the linguistic structure of utterances. Perhaps the most common language models used for speech recognition incorporate only “n-gram” statistics, looking at a sliding two-word (bigram) or three words (trigram) window to judge the likelihood of a recognition hypothesis [8]. Also there can be static or dynamic language models.

III. PROPOSED SYSTEM

An abstract view of the connected speech recognizer is shown in Fig. 1. As it depicts, the isolated word recognizer could act as a separate module. The connected speech recognition is obtained solely by the introduction of the word boundary detection and the language model. In an abstract sense, the word boundary detector will split the incoming speech signal into its constituent words. It then buffers these words and feeds them into the isolated word recognizer one by one. So the recognizer part is completely unaware of whether we have recorded connected speech or not. The output of the recognizer is also buffered, and then with the help of the language model, the recognized word is rectified by considering the past words.

A. Word Boundary Detection

One problem we faced when testing our system in different environments was that the inputs to the system were giving different levels of signals. The reason for this was the distortion level in different environments and different machine interfaces were in different levels. So we needed our system to adapt to the operational environment considering the input signal level. To cope with this problem we developed this dynamic threshold adaptation module. The task of this module was to get the initial inputs to the system and analyze them in terms of the signal strength when the initial input is given, and noise level when no input is available. The noise is obtained when there is no speech signal initially. Then the system will adapt to a general threshold level which enables the application to filter the noise by setting up an appropriate threshold level. So the system will automatically adapt to the environment where it’s being used.

B. Feature Extraction

The feature extractor used in our system is shown in Fig.2.

Linear Predictive Coding (LPC) Model has been used here. The calculated LPC parameters are converted to more robust cepstral coefficients.
C. Constant Trajectory Mapping
The output of the feature extractor is a set of 12 arrays. These correspond to the 12 cepstral coefficients calculated in the feature extraction phase and are meant to be fed into the ANN as input. Although these arrays are of the same size, this size varies from speech file to speech file. Since these varying length arrays can’t be fed into the ANN (since it requires a constant number of inputs), A self Organizing Map (SOM) is introduced to map those arrays to a constant trajectory. The SOM will reduce the dimensionality of the input vectors. It will organize each input array into 6 clusters. So, the output of the SOM is 72 values. These values are then fed into the ANN recognizer.

D. ANN Recognizer
The recognizer is implemented using the concept of Artificial Neural Networks. The ANN is implemented as an isolated word recognizer. Feature vectors corresponding to a word are fed to the ANN along with the desired output. Using these, the ANN will learn by adjusting weights. Then, once a test word is given, the ANN is supposed to recognize the word.

E. Language Model
The language model is used to rectify the probable grammatical mistakes that could occur during the isolated recognition process. This also may help to improve speech recognition in circumstances as the unconstrained speaking style, frequent grammatical errors, hesitations, start-overs, etc.

The Upasirasi language model mainly concentrates on two major aspects of continuous speech recognition,
- **Rectifying the grammatical errors** - At any instance the isolated recognizer may produce errors in outputs. The direct output from the recognizer may be unconstrained when it compiles a sentence. So we have developed a language model with specifically defined grammar rules. To reduce the dimensionality of parameters, we have chosen a moderately low vocabulary.
- **Improve recognising unconstrained speaking styles** - There can be two different sentences that sound alike in pronunciation. The recognizer must be capable of identifying the proper sentence in confronting such situations.

In surveying the existing language modeling techniques, Artificial Neural Network(ANN) based and probability based language modeling were the prominently used methods. But we came up with a model that uses a combination of both these approaches. We used a grammatical structure that is used in ANN based method. As we were testing on a fairly small vocabulary, we integrated the probabilistic model with the above structure. The probabilistic model was based on Markov chains. The words in a sentence were considered as a Markov chain. The occurrence probability of each word depends on the previous word in the sentence.

Developing and normalizing the Markov probability chains was a tiresome task. The probability chain values needed to be tested and updated to a global optimum. To make this improvement, we used evolutionary algorithms. Markov chain vectors were used in an evolutionary process to obtain its optimum. This process resulted in considerably well normalised vectors, for the Markov model.

Implementing this design, we managed to improve the recognition process very well. With the introduction of the language model, we managed to achieve a proper structuring and minimized errors in continuous recognition. But, in order to keep the design manageable, the model was designed only for three word sentences.

A fundamental problem that makes language modeling and other learning problems difficult is the “curse of dimensionality”. As the number of parameters grows, the complexity of the model increases. In our language model, as the vocabulary increases the complexity increases exponentially. For 12 words that are used, there can be maximum $3^{12}$ of three letter sentences.

IV. SYSTEM TESTING AND RESULTS
The system has two phases. During the training process, the speech utterances were recorded. These wave files were split into separate words and their features were stored in text files. Using these text files, the ANN is trained. During the testing phase, the user is asked to say a three word sentence (a limitation introduced by the language model). The system will retrieve the individual words and these words are then set through the ANN for recognition. (All the recordings were done using 16 KHz sampling rate and one sample was encoded using 16 bits)

![Fig. 3. The sentence uttered in connected speech style](image)
The above diagrams show how a speech signal is broken into its constituent words in a stepwise manner. As shown in them, the word boundary detection was successful. The ANN was tested in the following ways:

- By changing the number of layers
- By changing the number of neurons per layer
- By changing the parameters such as learning rate, back propagation learning coefficient, number of training iterations.

The SOM was also tested by changing the initial learning rate, initial neighborhood distance and their decaying rates.

Despite of these rigorous testing, the ANN was unable to recognize when a sentence was said during the testing phase. But we managed to get the error rate decaying in an acceptable manner.

V. FUTURE DIRECTIVE

This system could be easily extended to recognize continuous speech. If a more robust algorithm could be incorporated to split the speech signal into the constituent words, these words could be fed into the recognizer, without it worrying whether the speech signal has been continuous or not. To cater for a large vocabulary, the recognizer can also be modified. Right now, one output node of the ANN corresponds to one word. If this approach is taken, a problem arises when the number of words increases because the number of output nodes also increases proportionally. To address this problem, the neural network output can be considered as a digital sequence number. Then, 12 output nodes can be used to recognize $2^{12}$ output words.

A better approach is to modify the recognizer to recognize the basic sounds of English, called phonemes. This way, despite of the size of the vocabulary, the output size of the ANN will remain the same. To achieve this, the word boundary detection has to be modified in order to split the speech signal into separate phonemes. At the language model end, sophisticated probability estimation models could be implemented to construct words from the recognized phonemes and sentences from those words. The recognition of phonemes is a very difficult task, especially to break the speech signal into phonemes and to construct such a complex language model. But if this could be achieved, this system could show a new direction in the speech recognition process.

VI. CONCLUSION

In this paper, we tried to introduce a simple system which could be used to recognize connected speech. The basic
process has been implemented completely, though there were some problems when working with the SOM and the ANN.

Though this is still an on-going research-work, this system could be very easily fine-tuned into a commercial product, especially for a speech recognition system where there’s a cooperative user such as a sales clerk entering the digits of a credit card. Since the system is speaker independent, it can easily be adopted to use in similar situations where the size of the vocabulary is small.

The importance of this system is that it has been built on an isolated speech recognized system, making it very easy to enhance and expand the modules separate from each other. This makes it easy to reuse the recognizer part in isolated word recognition, connected speech recognition and continuous speech recognition by modifying the signal pre-processing front end and the language modeling back end. As mentioned under future directives, there’s a possibility of enhancing this system into a more sophisticated system, which would open yet another direction for the process of speech recognition.

ACKNOWLEDGMENT

The authors would sincerely like to thank Dr. Thrishantha Nanayakkara, Senior Lecturer, Department of Mechanical Engineering, University of Moratuwa, Dr. Chatura de Silva, Senior Lecturer, Department of Computer Science and Engineering, University of Moratuwa, for their supervision on this work, and Mr. Shantha Fernando, Senior Lecturer, Department of Computer Science and Engineering, University of Moratuwa, for his excellent coordination of the project.

REFERENCES