Project Title: Sinhala speech recognition system

Introduction

Understanding human speech is an idea which was once being an appeal of application users and developers. With the advancement of technology today many research projects have progressed with developing applications that use speech technology to boost application user’s experience.

This area has become a feature which is under many researches. As pioneers such as Microsoft and Sun Microsystems have released specific speech APIs in order to help application developers, today many researches are going on to develop applications which support locale specific languages.

There are two main speech technology concepts as speech recognition and speech synthesis. This project is based on speech recognition where the idea is to develop a system which is capable of converting human speech to text/command. Rest of this proposal describes the objective, requirement and selected methodology of the intended project.

Objective

Objective of this project is to develop an application which is embedded with the capability of understanding human speech done in Sinhala language. End product may be capable of tracking the human speech done in Sinhala language and print it in a text field while leaving a reliable DLL for Sinhala speech recognition that can be used for other applications extensively.

Problem / Requirement

Speech recognition is the process of converting an acoustic signal, captured by a microphone or a telephone, to a set of words. The recognized words can be the final results, as for applications such as commands & control, data entry, and document preparation. They can also serve as the input to further linguistic processing in order to achieve speech understanding, a subject covered in section.

Speech recognition systems can be characterized by many parameters, such as speaking model, speaking style, vocabulary. An isolated-word speech recognition system requires that the speaker pause briefly between words, whereas a continuous speech recognition system does not. Spontaneous, or extemporaneously generated, speech contains disfluencies, and is much more difficult to recognize than speech read from script. Some systems require speaker enrollment, a user must provide samples of his or her speech before using them, whereas other systems are said to be speaker-independent, in that no enrollment is necessary. A speaker independent system is developed to operate for any speaker of a particular type (e.g., American English), whereas a speaker adaptive system is developed to adapt its operation to the characteristics of new speakers. Some of the other parameters depend on the specific task. Recognition is generally more difficult when vocabularies are large or have many similar-sounding words. When speech is produced in a sequence of words, language models or artificial grammars are used to restrict the combination of words.
The simplest language model can be specified as a finite-state network, where the permissible words following each word are given explicitly. More general language models approximating natural language are specified in terms of a context-sensitive grammar.

One of the difficulties in speech recognition is that although different recordings of the same words may include more or less the same sounds in the same order, the precise timing - the durations of each sub-word within the word - will not match. As a result, efforts to recognize words by matching them to templates will give inaccurate results if there is no temporal alignment.

One popular measure of the difficulty of the task, combining the vocabulary size and the language model, is perplexity, loosely defined as the geometric mean of the number of words that can follow a word after the language model has been applied (see section for a discussion of language modeling in general and perplexity in particular). Finally, there are some external parameters that can affect speech recognition system performance, including the characteristics of the environmental noise and the type and the placement of the microphone.

And also there is an immense tendency of developing as well as a massive demand from the clients for applications which are capable of working with local languages. Even in Sri Lanka significant numbers of people get the use of various kinds of information systems and computer applications in their day-to-day life. Also there is a great demand of them for applications which can be work in Sinhala rather than foreign languages. By embedding this requirement with the speech recognition capability, it is possible to develop much useful programs ranging over different fields such as banking, mobile, finance etc.

But still there is no such reliable methodology to recognize Sinhala speech has been developed. This problem is the origin for this project.

**Methodologies**

There are two main methodologies which can be used in speech recognition.

- Hidden Markov model (HMM)-based speech recognition
- Dynamic time warping (DTW)-based speech recognition

- **Hidden Markov model (HMM)-based speech recognition**

  Modern general-purpose speech recognition systems are generally based on Hidden Markov Models. These are statistical models which output a sequence of symbols or quantities. One possible reason why HMMs are used in speech recognition is that a speech signal could be viewed as a piecewise stationary signal or a short-time stationary signal. That is, one could assume in a short-time in the range of 10 milliseconds, speech could be approximated as a stationary process. Speech could thus be thought of as a Markov model for many stochastic processes.
Another reason why HMMs are popular is because they can be trained automatically and are simple and computationally feasible to use. In speech recognition, the hidden Markov model would output a sequence of \( n \)-dimensional real-valued vectors (with \( n \) being a small integer, such as 10), outputting one of these every 10 milliseconds. The vectors would consist of cepstral coefficients, which are obtained by taking a Fourier transform of a short time window of speech and de correlating the spectrum using a cosine transform, then taking the first (most significant) coefficients. The hidden Markov model will tend to have in each state a statistical distribution that is a mixture of diagonal covariance Gaussians which will give likelihood for each observed vector. Each word, or (for more general speech recognition systems), each phoneme (is basically the smallest unit of phonetic speech that distinguishes one word from another. Every word can be broken down into units of individual sounds that make up that word. Each of these units is a phoneme), will have a different output distribution; a hidden Markov model for a sequence of words or phonemes is made by concatenating the individual trained hidden Markov models for the separate words and phonemes.

Decoding of the speech would probably use the Viterbi algorithm to find the best path, and here there is a choice between dynamically creating a combination hidden Markov model which includes both the acoustic and language model information, or combining it statically beforehand.

- **Dynamic time warping (DTW)-based speech recognition**

Dynamic time warping is an approach that was historically used for speech recognition but has now largely been displaced by the more successful HMM-based approach. Dynamic time warping is an algorithm for measuring similarity between two sequences which may vary in time or speed. For instance, similarities in walking patterns would be detected, even if in one video the person was walking slowly and if in another they were walking more quickly, or even if there were accelerations and decelerations during the course of one observation. The basic principle is to allow a range of 'steps' in the space of (time frames in sample, time frames in template) and to find the path through that space that maximizes the local match between the aligned time frames, subject to the constraints implicit in the allowable steps. The total `similarity cost` found by this algorithm is a good indication of how well the sample and template match, which can be used to choose the best-matching template. DTW has been applied to video, audio, and graphics – indeed, any data which can be turned into a linear representation can be analyzed with DTW.

A speech-enabled application does not directly interact with the audio hardware of the machine on which it runs. Instead, there is a common application, termed the Speech Engine, which provides speech capability and mediates between the audio hardware and the speech-enabled application.

**Conceive Attempts**

Inside the subject, Sinhala speech recognition as it name implies it has two main sub problems. Recognition of the speech and take it into a strings is first part of it and the other is working with a Sinhala language. The first sub problem has attempted by many developers for English
language in to significant accuracy level. But the next sub problem has attempted only by very few people and also success level is very poor.

For the solution there are few promising approaches within existing technologies. Windows operating system’s latest release which well known as Windows Vista has touched speech recognition for English to some extent. In this attempt speech engine which they get used has great support while being in middle layer. But solution is only for subjected language and also for Windows based systems. Plugging Sinhala in to it is yet to find out.

Apart to windows systems Sun Microsystems has try out about speech recognition through the JVM for Java language. The Java Speech API brings to the table all of the platform- and vendor-independent features commonly associated with any Java API. The Java Speech API enables speech applications to interact with speech engines in a common, standardized, and implementation-independent manner. Speech engines from different vendors can be accessed using the Java Speech API, as long as they are JSAPI-compliant.

However several other speech engines, both commercial and open source are exists. Among open source engines, the Festival speech synthesis system is one of the popular speech synthesis engines. Also there are attempts by bridging Windows and Java technologies. By literature surveying most positive approach can be found for the solution.