Advances in Computer Science and Information Technology (ACSIT)
Print ISSN: 2393-9907; Online ISSN: 2393-9915; Volume 2, Number 10; April-June, 2015 pp. 51-54

http://www.krishisanskriti.org/acsit.html

Detection of Word Boundary in Speaker Independent Assamese Speech

Sudarshana Sarma[1] and Uzzal Sharma[2]

[1,2]Dept. of Computer Science & Engineering and IT Assam Don Bosco University Guwahati, India
E-mail: [1]sudarshanasarma8@gmail.com, [2]uzzal.sharma@dbuniversity.ac.in

Abstract—*In the presented work we are dealing with modelling a speaker independent word boundary detection system. Many of the methods are applied in different Indian languages like Hindi, Telugu, Marathi, Bengali. Many methods can also be found in German language to detect word boundary. But there are very few such methods or work done for word boundary detection in Assamese language. This is the main reason for which we want to develop a technique, by which we can detect word boundary in Assamese language. This paper presents, how to develop speaker independent word boundary detection system using Hidden Markov Model. To model HMM, we use HTK..*

1. INTRODUCTION

1.1 What is Speech

Speech is the verbal means of communicating. Speech is the expression of or the ability to express thoughts and feelings by articulate sounds. It is the vocalized form of human communication. Speech is the syntactic combination of lexicals and names that are drawn from very large vocabularies. Each spoken word consist of the phonetic combination of a limited set of vowel and consonant speech sound units.

Speech consists of the following:

Articulation: How speech sounds are made.

Voice: Use of the vocal folds and breathing to produce sound

Fluency: The rhythm of speech.

In speech processing, speech sounds are divided into two broad classes voiced speech and unvoiced speech, which depend on the role of the vocal chords on the speech production mechanism. Voiced speech are produced, when the vocal chords vibrate in the production of a sound. Vocal chords are vibrate at a particular frequency, which is called the fundamental frequency of the sound.

50: 200 Hz for male speaker

150:300 Hz for female speaker

200:400 Hz for child speaker.

Unvoiced speech is produced when vocal chords are not vibrate.

1.2 What is Word Boundary

Word boundaries are represented by pause between words. Word boundary detection (WBD) is very common and important issue in the field of speech synthesis and recognition. The recognition of continuous speech presents the listeners (human or machine) with a problem which does not arise in the recognition of isolated words.

Difficulties [1] are discussed below:

Language Independence: Several language features can be exploited for word boundary detection. But, clearly these are language dependent, and cannot be applied commonly for all languages. Different languages have different features. Mainly there are two categories of languages: lexical accent and tone language. A lexical accent type of language is having an autosegmental feature that does not provide any clues about its phonetic manifestation. A lexical accent[8] has two types, it can be ‘strong’ or ‘weak’. A strong accent corresponds to a head and is phonetically realized as stress in languages with dynamic stress or high pitch in pitch- accent languages. A weak accent, on the other hand, is an accent that lacks prominence. Example of lexical accent are Greek language etc.. Tonal language is a language in which variations in pitch distinguish different words. Chinese languages are tonal language. This means that in case of lexical accent language, we need to calculate the stress to find the behaviour of sentence and expressive style. Similarly in case of tone language, we need to calculate the pitch to find the behaviour of sentence and expressive style. The method developed must use some universal characteristics of the languages rather than features that are language specific.

Length of word string: Continuous speech can either be some collected words, a phrase or a sentence. There is no constant length of the speech input. So a method used should be effective over all continuous speech input.

No explicit clues: Continuous speech offers no explicit clues for placing the word boundaries. The speaker does not pause

consciously between words to signify the word boundaries while speaking or reading continuously. So speech signal related clues must be used for marking the word boundaries.

1.3 Speaker Dependent and Speaker Independent System

In a speaker dependent system is intended for use by a single speaker, but a speaker independent system is intended for use by any speaker. In speaker dependent, system depends on the particular speaker. But in case of speaker independent, system does not depends on the particular speaker. In speaker dependent systems, we have to "train" the system for each user individual speech pattern, dialect or language. But In Speaker-independent solutions try to match the user's voice to generic voice patterns.

1.4 Summary of the Result and Technique Used

In this paper we have used the technique hidden markov model to develop a speaker independent word boundary detection system in Assamese language. For building Hidden Markov Models (HMMs), we have used HTK. HTK is primarily designed for building HMM-based speech processing tools, in particular recognisers. After the experiment, we found that, more than 80% word boundary correctly detected. It will give better performance than the other methods, which are used to detect word boundary in Assamese language.

2. HMM AND HTK

2.1 Hidden Markov Model

A hidden Markov model (HMM) is a statistical Markov model in which the system being model is assumed to be a Markov process with unobserved (hidden) states. In a hidden Markov model, the state is not directly visible, but output is visible. Output is dependent on the state. Also each state is dependent on the previous state, x(t) is the state at time t and y(t) is the output. Here x(t) is hidden state and y(t) is observed state.

2.2 Relationship of HMM to Speech

Speech signal are some message encoded as a sequence of one or more symbols. To effect the reverse operation of recognising the underlying symbol sequence given a spoken utterance, the continuous speech waveform is first converted to a sequence of equally spaced discrete parameter vectors. These parameter vector are observed state and speech waveform are hidden state.

The role of the recognizer[3] is to effect a mapping between sequences of speech vectors and the wanted underlying symbol sequences. Two problems make this very difficult. Firstly, the mapping from symbols to speech is not one-to-one since different underlying symbols can give rise to similar speech sounds. Furthermore, there are large variations in the realised speech waveform due to speaker variability, mood, environment, etc. Secondly, the boundaries between symbols cannot be identified explicitly from the speech waveform. Hence, it is not possible to treat the speech waveform as a sequence of concatenated static patterns.

Let each spoken word be represented by a sequence of speech vectors or observations O, defined a

$$O= O1,O2,O3…..,OT$$

where OT is the speech vector observed at time t. Only the observation sequence O is known and the underlying state sequence X is hidden.This is why it is called a Hidden Markov Model.

2.2 HTK

HTK is a toolkit for building Hidden Markov Models (HMMs). HMMs can be used to model any time series and the core of HTK is similarly general-purpose. However, HTK [2] is a tool, which is primarily designed for building HMM-based speech processing tools, in particular recognisers. As shown in the picture alongside, there are two major processing stages involved. Firstly, the HTK training tools are used to estimate the parameters of a set of HMMs using training utterances and their associated transcriptions. Secondly, unknown utterances are transcribed using the HTK recognition tools.

Available HTK[3] tools :

- Data preparation tools: Convert speech waveforms into parametric format (e.g. MFCC)
- Convert the associated transcriptions into appropriate format (e.g., phone or word labels)
- Training: Define the topology of the HMMs (i.e., prototypes), initialize models (e.g., bootstrap, flat start), train models (e.g., parameter tying, Baum-Welch, adaptation)
- Testing: Viterbi based recognizer (HVite) can also be used for forced alignment. Decoder for large vocabulary speech recognition (HDecode)
- Analysis: Evaluate model performance (e.g., WER, ROC, …)

3. EXPERIMENTATION

Steps to Detect Word Boundary Using HMM:

Step 1: Record data from different informants.

Step 2: Create grammar

The HTK recogniser requires a word network to listed each word-to-word transition. For this purpose, we need to create grammar to build word network using the HParse tool.

$WORD = aath | ase | bhal | dui | ek | khat| khunyo | lora | nou | pas | porhi | ram | sari | soy | tai | tini ;

(<$WORD>)

Step 3: Create Dictionary

The first step in building a dictionary is to create a sorted list of the required words.

aath aath

ase ase

bhal bhal

dui dui

ek ek

khat khat

khunyo khunyo

lora lora

nou nou

pas pas

porhi porhi

ram ram

sari sari

soy soy

tai tai

tini tini

Step 4: Creating the Transcription Files

To train a set of HMMs, we have done phone level transcription for every file of training. The first line of the file identifies the file as a Master Label File (MLF). MLF is a single file containing a complete set of transcriptions. HTK allows each individual transcription to be stored in its own file but it is more efficient to use an MLF.

When HTK processes speech files, it expects to find a transcription (or label file) with the same name but a different extension. Thus, if the file /root/data/1_1.wav is being processed, HTK would look for a label file called /root/data/1_1.lab. When MLF files are used, HTK scans the file for a pattern which matches the required label file name.

#!MLF!#

"*/1_1.lab"

khunyo

ek

dui

tini

sari

pas

soy

khat

aath

nou

Step 5: Calculate MFCC

In this step, Mel Frequency Cepstral Coefficients (MFCCs) is used, which are derived from FFT-based log spectra. Coding can be performed using the tool HCopy. HCopy automatically convert its input into MFCC vectors. To do this, a configuration file (config) is needed which specifies all of the conversion parameters.

STEP 6: Creating Monophone HMMs

The starting point will be a set of identical monophone HMMs in which every mean and variance is identical. The tool HVite can be used to perform a forced alignment of the training data.

Step 7: Creating Flat Start Monophones

The first step in HMM training is to define a prototype model. The parameters of this model are not important, its purpose is to define the model topology.

Step 8: Recogniser Evaluation

The recogniser is now complete and its performance can be evaluated. The recognition network and dictionary have already been constructed, and test data has been recorded. Thus, all that is necessary is to run the recogniser and then evaluate the results using the HTK analysis tool HResults.

Step 9: Recognising the Test Data

Here each test file will be recognised and produce output in recout.mlf file.

4. RESULT &DISCUSSION

By the above experiment, we get different result for each speaker. Because each people uttered differently. But it gives more accuracy than the other method like autocorrelation method ,cepstrum method. Also our system is speaker independent system. It doesnot depend on particular speaker.

Speaker	Hit Rate(%)	FalseAlarm Rate(%)
1	78	23
2	100	30
3	100	28
4	80	35
5	100	21

5. CONCLUSION

From the above observation it can be conclude that word boundary detection using HMM, gives more accurate and efficient speaker independent word boundary detection system.

REFERENCES

[1] Srichand “Word Boundary Detection in Indian Languages and Application to Keyword Spotting”, Department of Computer Science and Engineering Indian Institute of Technology, Madras- 600 03G, India, August 1996.

[2] HTK Tutorial,Giampiero Salvi,KTH (Royal Institute of Technology),Dep. of Speech, Music and Hearing,Drottning Kristinas v. 31,SE-100 44, Stockholm, Sweden giampi@speech.kth.se. .

[3] The HTK Book,Steve Young,Gunnar Evermann,Mark Gales,Thomas Hain,Dan Kershaw,Gareth Moore,Julian Odell,Dave Ollason,Dan Povey,Valtcho Valtchev,Phil Woodland,Cambridge University Engineering Department,First published December 1995,Revised for HTK Version 3.3 April 2005.

[4] Jan Bartoˇsek and V´aclav Hanˇzl ,“Foot Detection in Czech Using Pitch Information and HMM,”,Department of Circuit Theory, FEE CTU in Prague, Technick´a 2, 166 27 Praha 6-Dejvice, Czech Republic.

[5] Manash Pratim Sarma and Kandarpa Kumar Sarma, “Speech Recognition of Assamese Numerals Using Combinations of LPC-Features and Heterogenous ANNs”, Department of Electronics And Communication Technology,Gauhati University, Assam, India, Springer-Verlag Berlin Heidelberg 2010, pp. 8–12..

[6] J. Srichand “Word Boundary Detection in Indian Languages and Application to Keyword Spotting,” Department of Computer Science and Engineering Indian Institute of Technology, Madras- 600 03G, India, August 1996.

[7] Fry, D.B.: Experiments in the perception of stress. Language and Speech 1, 126–152 (1958).

[8] http://linguistics.berkeley.edu/bls/past_meetings/bls34/abstracts34/Sugiyama.pdf.

[9] http://roa.rutgers.edu/files/388-0400/388-0400 REVITHIADOU-31.PDF