Android Speech-to-speech Translation System For Sinhala

Layansan R., Aravinth S. Sarmilan S, Banujan C, and G. Fernando

Abstract— Collaborator (Interactive Translation Application) speech translation system is intended to allow unsophisticated users to communicate in between Sinhala and Tamil crossing language barrier, despite the error prone nature of current speech and translation technologies. System build on Google voice furthermore also support Sinhala and Tamil languages that are not supported by the default Google implementation barring it is restricted to common yet critical domain, such as traveling and shopping. In this document we will briefly present the Automatic Speech Recognition (ASR) using PocketSphinx, Machine Translation (MT) architecture, and Speech Synthesizer or Text to Speech Engine (TTS) describing how it is used in Collaborator. This architecture could be easy extendable and adaptable for other languages.

Index Terms— Speech recognition, Machine Translation, PocketSphinx, Speech to Text, Speech Translation, Tamil, Sinhala

1 INTRODUCTION

Language barriers are a common challenge that people face every day. In order to use technology to overcome from this issue many European and Asian countries have already taken steps to develop speech translation systems for their languages. In this paper we discuss how we managed to build a speech translator for Sinhala and Tamil languages that were not supported by any speech translators so far. This model could be adopted to build speech translator that support any new language.

Various approaches to speech translation have been proposed and investigated by [1], [2], [3] and [4]. Presently finite-state technology allows us to build speech-to-speech translation (ST) systems by coupling automatic speech recognition (ASR), machine translation (MT) and Speech Synthesizer or Text to Speech Engine (TTS). Then again there are no application or libraries available which supports Tamil and Sinhala ASR or TTS. However Google Translate API provide MT for these languages with reasonable error rate [5].

Although the breath and width of Google's language technology support is remarkable, there are many reasons not to rely only them. Such as accuracy and control over privacy.

This project uses PocketSphinx, the android version of CMUSphnix which is an open source speech recognition toolkit that capable of maximizing the accuracy of recognizing the speech [6] while Google voice search API is used for speech recognition of its supported languages. Language models & dictionaries can be changed to maximize the accuracy of the recognizer. This research mainly focuses on spoken Sinhala and spoken Tamil. For this purpose, vocabularies and grammars in both languages specific to the development domain is considered.

The paper is organized as follows: In section II, technologies and already existing systems and literature that is available with similar studies are briefly discussed, in section 3, the details regarding the use of components and how each of them can be build is discussed. In the following sections the findings are described, and test results are shown with conclusion.

2 BACKGROUND STUDY

2.1 Speech Recognition

In the early 1920s The first machine to recognize speech to any significant degree commercially named, Radio Rex was manufactured in 1920[6], The earliest attempt to devise systems for automatic speech recognition by machine were made in 1950s.During 1950s [7], most of the speech recognition systems investigated spectral resonances during the vowel region of each utterance which were extracted from output signals of analogue filter bank and logic circuits.

In 1960s was the pioneering research of Reddy in the field of continuous speech recognition by dynamic tracking of phonemes [8].Reddy's research recognition research program at Carnegie Mellon University remains a world leader in continuous speech recognition systems.

One of the demonstration of speech understanding was achieved by CMU in 1963 CMU's Harphy system [9] was shown to be able to recognize speech using a vocabulary of 1011 words with reasonable accuracy. In the 1970s speech recognition research achieved a number of significant milestones. By the mid 1970's, the basic ideas of applying fundamental pattern recognition system to voice and speech recognition were proposed by Itakura, Rabiner and Levinson and others [10] based on LPC methods. Another system was introduced in the late 1980's was the idea of Artificial Natural Networks.

Speech research in the 1980s was characterized by shift in technology from template based approaches to statistical modeling methods especially the hidden Markov model approach [11, 12].In mid 1980s that the techniques became widely applied in virtually in every speech recognition research in the world. Today most speech recognition systems are based on the statistical frame work developed in the 1980s and 1990s significant additional improvements have been made in their results.

In 2008, Google has launched voice search (Application that lets user search the web using a spoken query) is support 39 languages as 2014 December [13].

Depends on Google Voice API there are several attempts made by researchers to speech recognition process but Google Voice API does not support Tamil, Sinhala .In 2000, the Sphinx group at Carnegie Mellon committed to open source several speech recognizer components, including Sphinx 2 and later Sphinx 3, Sphinx 4, PocketSphinx for mobile devices .PocketSphinx as an open source library enables developers to add new language [9]. However, it requires an acoustic model and a language model. Using
PocketSphinx technology in 2014 P.Vijai Bhaskar and Dr. S.Rama Mohana Rao came up with Telugue Speech recognition system[14] and there are few attempts were made in translating Tamil to English through voice [15].

2.2 Translator

The mechanization of translation has been one of humanity's oldest dreams. In the twentieth century it has become a reality, in the form of computer programs capable of translating a wide variety of texts from one natural language into another. The first requirement was to demonstrate the technical feasibility of Machine Translation (MT). Accordingly, at Georgetown University Leon Dostert collaborated with IBM on a project which resulted in the first public demonstration of a MT system in January 1954[16].

In 1976 the Commission of the European Communities decided to install an English-French system called Systran, which had previously been developed by Peter Toma for Russian-English translation for the US Air Force, and had been in operation since 1970.In 1980s the transfer-based design has been joined by new approaches to the Interlingua idea. The research on knowledge based systems, remarkably at Carnegie Mellon University, Pittsburgh, which are founded on developments of natural language understanding systems within the Artificial Intelligence (AI) community.

Google launched the Google Translate API for software developers had been deprecated and would cease functioning on December 1, 2011.

2.3 Text-to-Speech

In 18th century the first synthetic speech was developed. The first audiable speech machine that generating speech was built with wood and leather [19]. It was demonstrated by Wolfgang von Kempelen and had great importance in the early studies of Phonetics.
In the 20th century, the first known electric speech synthesis was Voder and its creator Homer Dudley. During the 1950s Gunnar Fant was responsible for the development of the first Swedish speech synthesis OVE (Orator Verbis Electris). During that time Walter Lawrences Parametric Artificial Talker (PAT) that could compete with OVE in speech quality. OVE and PAT were text-to-speech systems using Format synthesis [20].
During the last decades greatest improvements in natural speech. Diphone synthesis was used for Read Speaker and the voices are sampled from real recorded speech and split into phonemes and Concatenation synthesis was used in small unit of human speech. They have a synthetic sound [21]. Diphone voices for some smaller languages are still in use and they are widely used to speech-enable handheld computers and mobile phones due to their limited resource consumption [22].

2.4 Research Gap

For the past 60 years voice and Speech recognition research and translation to other languages has been characterized by the steady accumulation of small increment improvement. Therefor the project is to use technology to solve communication conflicts within our country by providing Tamil-English-Sinhala translator using speak reorganization for android mobile operating system which were not supported by the default Google implementation and to provide speaker-independent continues speech recognition system.

- Converting user voice input to text using CMU Sphinx speech recognition library.
- Translating the input text to another language
- Reading text aloud naturally.

There are some reasons which make this project to be unique and to justify why we should carry on this project.

- This is the first Tamil to Sinhala translator related tool can be used while shopping.
- Reduce Language barrier among people.
- Easy to use

3 METHODOLOGY

Following section explains the Methodology. Figure 1 illustrate the system architecture diagram.

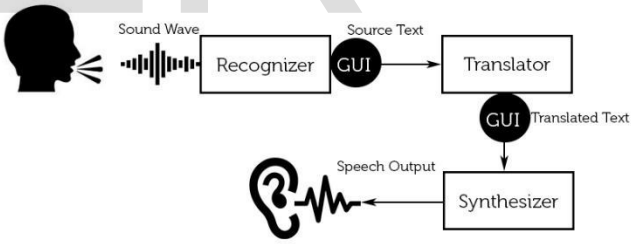

Figure 1. Architecture diagram

Based upon a Hidden Markov Models (HMM) speech translation system, in this study, we try to typically integrate the following three software technologies: automatic speech recognition (ASR), machine translation (MT) and speech synthesis (TTS). The mobile application will initially convert user voice into text. Translator built on top of Google translate API and modifications are performed in order to increase the accuracy of the results. Synthesizer vocalize translated text with authentic and original voices that express meaning and intent. Up to 2 users will able to converse on a single device by face-to-face or through text for the hearing impaired.

3.1 Speech Recognition

As there are no speech recognition libraries or API exists

that support Sinhala or Tamil so we had to build one. But building a continuous speech recognizer for language like Sinhala and Tamil is a challenging task due to the unique inherent features of these language like lack of enunciated stops, pronounced consonants, short and long vowels and many occurrences of allophones. Stress and accent vary in spoken Sinhala and Tamil language from region to region.

For this purpose, PocketSphinx, A version of Sphinx that can be used in embedded systems is under active development and incorporates features such as fixed-point arithmetic and efficient algorithms for GMM computation is used.

To describe more complex language statistical language model could be used as they contain probabilities of the words and word combinations but as we have restricted the development [18], We Initially gathered list of words and phrases of Sinhala and Tamil in selected situations by crawling websites and human observation. With that, Pronunciation dictionary from describe content for recognition is generated automatically with CMU LOGIOS Lexicon Tool. And then multiple Grammar file with JSGF format with .gram extension is originated for each development domain. It should be noted Grammar do not have probabilities for word sequences and describe very simple types of languages.

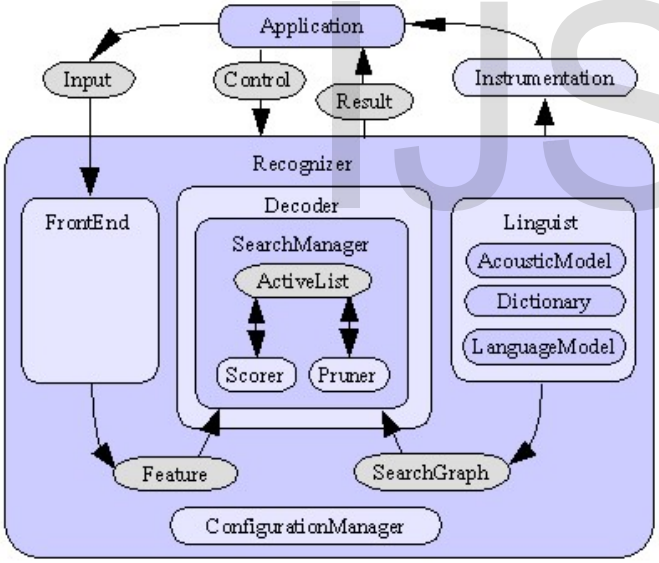

Figure 2. Architecture diagram of CMUSphinx

3.2 Transliteration

One of the most frequent problems translators must deal with is transliteration. Sinhala, Tamil & English languages have different alphabets, Transliteration is the rendering of a text from one script to another. For instance, a Sinhala transliteration of the phrase "āyaubaeāvan" is "ආයුබෝවන්" usually translated as 'Hello'. For that characters are one-toone mapped in the source language into the target script according to the user's preference

A SQLite database that contains source and target text block and characters is embedded into the application.

Table 1 and Table 2 to shows partial Romanization table of Sinhala and Tamil languages.

TABLE I
ROMANIZATION TABLE OF SINHALA

Vowels and Diphthongs			
අ	a	ඒ	ē
ආ	ā	ඔ	o
ඇ	ă	ඕ	ō
ඈ	â	ඍ	
ඉ	i	ඎ	
ඊ	ī	ඏ	
උ	u	ඐ	
ඌ	ū	ඓ	ai
එ	e	ඖ	au
Gutturals		Palatals	
ක	ka	ච	ca
ඛ	kha	ඡ	cha
ග	ga	ජ	ja
ඝ	gha	ඣ	jha
ඞ	ṅa	ඤ	ña
Labials		Semivowels	
ප	pa	ය	ya
ඵ	pha	ර	ra
	ba	ල	la
භ	bha	ළ	ḷa
ම	ma	ව	va
Cerebrals		Dentals	
ට	ṭa	ට	ṭa
ඨ	ṭha	ඨ	ṭha
ඩ	ḍa	ඩ	ḍa
ඪ	ḍha	ඪ	ḍha
ණ	ṇa	ණ	ṇa

TABLE I
ROMANIZATION TABLE OF SINHALA

Vowels and Diphthongs			
அ	a	எ	e
ஆ	ā	ஏ	ē
இ	i	ஐ	ai
ஈ	ī	ஒ	o
உ	u	ஓ	ō
ஊ	ū	ஔ	au
Consonants			
ஃ	ka	ம	ma
க	ka	ய	ya
ங	ṅa	ர	ra
ச	ca	ல	la
ஞ	ña	வ	va
ட	ṭa	ழ	la
ண	ṇa	ள	ḷa
த	ta	ற	ra
ந	na	ன	na
ப	pa		
Sanskrit Sounds			
ஜ	ja	ஸ	sa
ஶ	śa	ஹ	ha
ஷ	ṣa		

3.3 Transliteration

MT use Google translate API as it is the best available translator that supports 90 languages as of now. (Google Trans-late API is a paid service) Figure 3 shows the architecture used with Translate API.

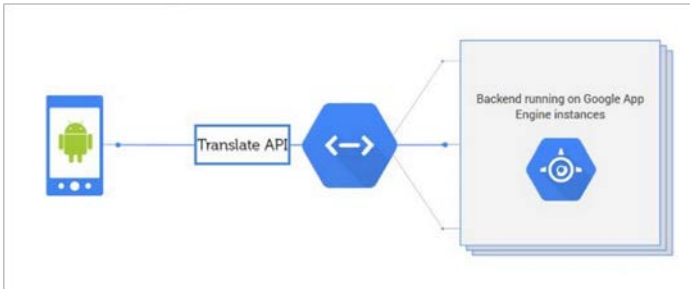

Figure 3. Google translation process

Stress and accents are ignored in formal reading form of Tamil and Sinhala language, it is necessary to modify the result so that it can be used in the manner people speak. But as the translation backend is running on Google application engine instance, developers cannot override any functions. Therefor Rule-Based Machine Translation (RBMT) is built to provide realistic output in addition, it could work off-line with limited words and phrases.

Initially the translation engine takes in the input text from the translitor that must be a sentence. Translation process is further divided into two segments. At the first step source language sentence pattern is analysed partitioned into sub-sets. Afterward they are mapped in to target language. This process not going in to depth considering all the grammar rules. The output is displayed as Sinhala text and delivered as speech using Sinhala TTS.

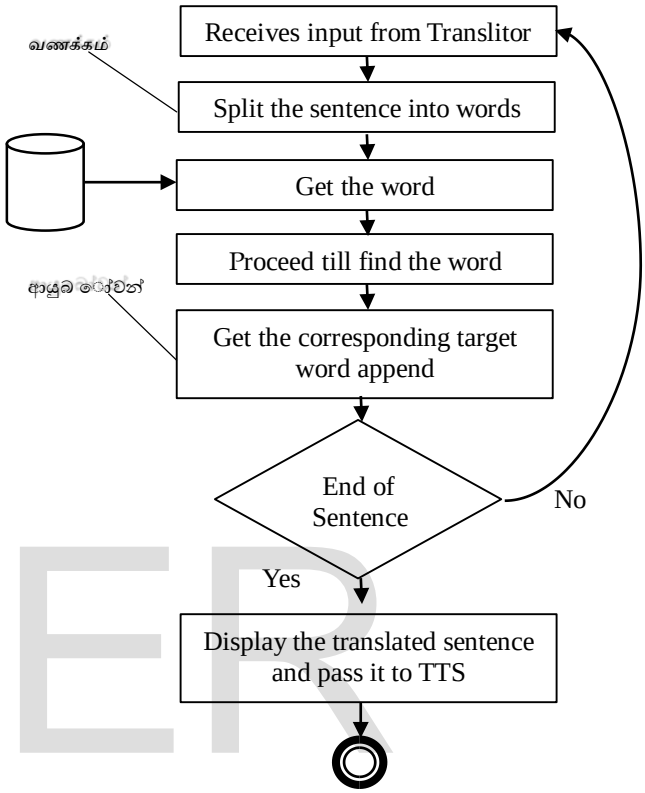

Figure 4. Diagram of Translation

3.4 Text to Speech Engine

English TTS

Application initially let the user to choose preferred TTS which is already installed in their device and that support corresponding language. Such as Google TTS engine, Samsung TTS or etc. If there is no TTS installed on the device or the current TTS doesn't support the language that application trying to read out, it will prompt the user to install Google TTS as it is recommended for android.

Sinhala and Tamil TTS

Significant issue of Google TTS engine is so far it does not support Tamil or Sinhala locale therefor development has to be carried out using other locales which results unrealistic output to the listener. As a minimal solution outcome of the language translator divided into sets and corresponding encoded pre-recorded audio files are streamed from the servers to the device.

4 RESULTS AND DISCUSSIONS

We implemented an open and extendable ASR architecture that enables to develop speech-based applications for mobile platforms. This challenging development could be done by integrating ASR, MT and TTS adopting them with new technologies.

A speech recognizer will never obtain the quality of a human speech. However, a recognition test for 100 sample sentences was conducted using our system by 5 speakers (3 male and 2 female speakers) in a noiseless environment, and an acceptable recognition result of 70 percent was achieved .

The framework makes it relatively easy to develop mobile speech input applications for other languages that are, for example, not yet supported natively by the OS vendor. The support for grammar based decoding allows to create limited vocabulary applications even for languages that don't have enough training data for creating robust LVCSR models.

5 REFERENCES

[1] "Finite-State Speech-to-Speech Translation", Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, pp. 111–114, Munich, Germany, 1997.

[2] H. Ney, "Speech Translation: Coupling of Recognition and Translation", Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, pp. 1149–1152, Phoenix, AZ, 1999.

[3] S. Bangalore and G. Riccardi, "Finite-State Models for Lexical Reordering in Spoken Language Translation", Proc. Int. Conf. on Spoken Language Processing, vol. 4, pp. 422–425, Beijing, China, 2000.

[4] S. Saleem, S.-C. Jou, S. Vogel, and T. Schultz, "Using Word Lattice Information for a Tighter Coupling in Speech Translation Systems", Proc. Int. Conf. on Spoken Language Processing, pp. 41–44, Jeju Island, Korea, 2004.

[5] U. REST, 'Using REST', Google Developers, 2015. [Online]. Available: https://cloud.google.com/translate/v2/using_rest. [Accessed: 29- Jun- 2015].

[6] Cmusphinx.sourceforge.net, 'About CMUSphinx [CMUSphinx Wiki]', 2015. [Online]. Available: http://cmusphinx.sourceforge.net/wiki/about. [Accessed: 29- Jun- 2015].

[7] Stefan Windmann and Reinhold Haeb-Umbach,Approaches to Iterative Speech Feature Enhancement and Recognition , IEEE Transactions On Audio,Speech, And Language Processing, Vol. 17, No. 5, July 2009.

[8] Sadaoki Furui, 50 years of Progress in speech and Speaker Recognition Research , ECTI Transactions on Computer and Information Technology,Vol.1. No.2 November 2005.

[9] D.R.Reddy, An Approach to Computer Speech Recognition by Direct Analysis of the Speech Wave, Tech.Report No.C549, Computer Science Dept., Stanford Univ., September 1966.

[10] B.Lowrre, The HARPY speech understanding system, Trends in Speech Recognition, W.Lea,Ed., SpeechScience Pub., pp.576-586,1990.

[11] F.Itakura, Minimum Prediction Residula Applied to Speech Recognition, IEEE Trans.Acoustics, Speech,Signal Proc., ASSP-23(1):6772,February 1975. J.Ferguson, Ed., Hidden Markov Models for Speech, IDA,Princeton, NJ,1980.

[12] L.R.Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proc.IEEE, 77(2):257-286, February 1989.

[13] "Google Voice Search "Google ResearchBlog. 2014.

[14] D.H.Daines, M.Kumar, A.Chan,M.Ravishankar,and I. Rudnicky," POCKETSPHINX: A FREE, REAL-TIME CONTINUOUS SPEECH – Mellon University,Language Technologies Institute 5000 Forbes Avenue, Pittsburgh, PA, USA 15213,2006

[15] V. S. Dharun, M. Karnan," Voice and Speech Recognition for Tamil Words and Numerals", Vol.2, Issue.5, Sep-Oct. 2012.

[16] K.M. Ganesh,S. Subramanian," Tamil IT! : Interactive Speech Translation in Tamil", 2002.

[17] J. Hutchins, "The first public demonstration of machine translation" the Georgetown-IBM system, 7th January 1954.

[18] Cmusphinx.sourceforge.net, 'Building Language Model [CMUSphinx Wiki]', 2015 [Online]. Available: http://cmusphinx.sourceforge.net/wiki/tutoriallm. [Accessed: 30- Jun- 2015].

[19] Abadjieva E., Murray I., Arnott J Applying Analysis of Human Emotion Speech to Enhance Synthetic Speech. Proceedings of Eurospeech 93 (2): 909-912, 1993.

[20] Karjalainen M. (1978). An Approach to Hierarchical Information Process with an Application to Speech Synthesis by Rule. Doctorial Thesis. Tampere University of Technology

[21] Liu, Liqing, and Tetsuya Shimamura. 'Pitch-Synchronous Linear Prediction Analysis of High-Pitched Speech Using Weighted Short-Time Energy Function'. Journal of Signal Processing 19.2 (2015)

[22] Kumara, KH, NGJ Dias, and H Sirisena. 'Automatic Segmentation Of Given Set Of Sinhala Text Into Syllables For Speech Synthesis'. Journal of Science of the University of Kelaniya Sri Lanka 3.0 (2011): n. pag. Web.