International Journal of Electronics Communication and Computer Engineering
Volume 4, Issue 2, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

Survey on Real Time Voice Translation System

Prof. D. J. Pereira
Govt. College of Engineering & Research
Govt. Awasari(kd), Pune
dannyjpereira@gmail.com

Vishal Jadhav
Govt. College of Engineering & Research
Govt. Awasari(kd), Pune
jadhav.vishal92@gmail.com

Jaydeep Joshi
Govt. College of Engineering & Research
Govt. Awasari(kd), Pune
jjaydeep.joshi@gmail.com

Prashant Bhosale
Govt. College of Engineering & Research
Govt. Awasari(kd), Pune
bhosaleprashant777@gmail.com

Swapnil Dhalpe
Govt. College of Engineering & Research
Govt. Awasari(kd), Pune
swadhalpe@gmail.com

***Abstract* — In today's world language translation is very important, because if two person talking with each other but talking language is different, in this situation, problem is occurred, at that time voice translation is very useful. Voice translator is mediator between two languages. In this paper, we have review various issues related to voice translation as well as various difficulties in voice translation. The purpose of this paper is to develop a Voice translator that was able to do a real time translation from a spoken sentence in one language to a spoken sentence in other language.**

***Keywords* - Speech Recognition (ASR), SPHINX, Machine Translation (MT), Text-to-Speech Synthesis (SS).**

I. INTRODUCTION

Basically any form of written or speech communication is through various Languages. The communication among human computer interaction is called human computer interface. Nowadays, the presence of multiple languages has been a hindrance to effective communication. In India the language and dialect changes with region, the requirement of a middle translation layer that can eliminate the linguistic barriers becomes essential. [1]
There are two ways of Translation:
1: Text-based translation and
2: Voice-based translation.

In this paper we are discussing about voice based translation .The objective of a speech to speech translation (SST) system is to convert speech from one language to another. A typical SST system consists of three components:
1. Speech recognition system (ASR) which converts speech in source language to its corresponding text
2. Machine translation system (MT) which translates text in source language to text in target language and
3. Speech synthesis system (SS) which converts text in target language to its spoken form.

The conventional architecture of such SST system is shown in Figure 1. It is a cascade architecture, where ASR, MT and SS systems are loosely coupled to form a SST system. The output obtained from an ASR system is given as input to an MT system, and the output obtained from the MT system is given an input to a SS system. Given a perfect ASR and MT systems, this architecture may be sufficient to achieve the goal of SST system. However, due to the limitations of current state-of-art technology in ASR and MT areas there are errors or ambiguities in the output of these components which are propagated to its successive components. Near about all translation methods make use of dictionaries to find the translated words. However, searching for required words and synonyms from such large dictionaries is very slow and very time-consuming. It also depends on the content of the sentence being translated. That means if you have larger sentence then it takes much time to convert into target language [4].

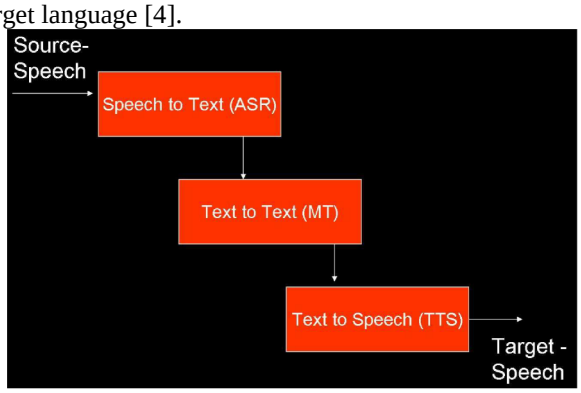

Fig.1. Cascade architecture of a typical speech to speech translation system [4].

II. RELATED WORKS

According to understanding the need of speech translation in now day's. we are going to developing speech translation system. Figure 2 give the details about system architecture of our system. The input speech is recognized using an automatic speech recognizer and then parsed by a statistical natural language understanding (NLU) module. An information extraction numbers and other attributes detected by our semantic model. The resulting representations are sent to an natural language generation (NLG) engine to render in the target language. The two types of information are translated using distinct models, with the specific attributes of items, such as times and dates, using conventional techniques familiar to the machine translation community. The Interlingua translation, however, uses statistical techniques and can perform considerable surface changes when required for the target language. Finally, when a textual representation of the utterance in the target language is complete, a text-to-speech synthesizer is used to produce spoken output [3].

In this paper highlights the following area:
1. Speech to text Translation (ASR)
2. Text to text translation
3. Text to speech translation

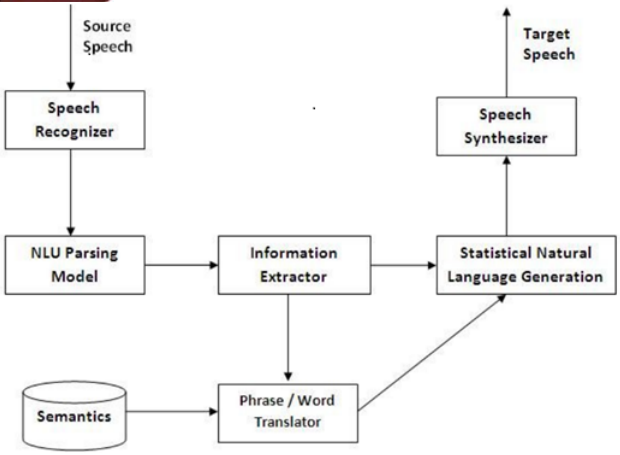

Fig.2. System Architecture for a Speech-to-Speech Translation System

III. SPEECH TO TEXT TRANSLATION (ASR)

Speech recognition is an area of speech processing that enables humans to communicate with computer through speaking [5]. Recognition is 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 [2]. The goal of speech-to-text transfer is type into text of spoken Words. However, this process will be carried out at the backside. If children are not sufficiently exposed to spoken language, their oral language system may develop slowly and less effectively compared with their peers.

Over the years, there are four basic approaches to attain ASR goals [6]:

Template-based approach, where incoming speech is compared with stored units in an effort to find the best match.

Knowledge based approach that emulate human expert ability to recognize speech.

Stochastic or statistical-based approach, which exploit the inherent statistical properties of the occurrence and co-occurrence of individual speech sounds.

Connectionist approach that use networks of interconnected nodes, which are trained to recognize speech.

Structure of Speech Recognisation

Speech recognition process or framework in general view is shown below in Figure 3 [7].

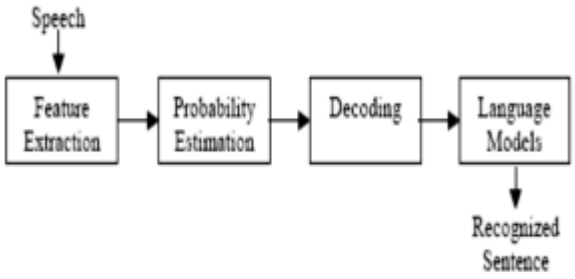

Fig.3. Speech recognisation process

The first stage is record the speaker voice which is consists of acoustic environment and transduction equipment (microphone).

The first block, feature extraction is intended to derive acoustic representations that are both good at separating classes of speech sounds and effective at suppressing irrelevant sources of variation.

Then the next two blocks are the core acoustic pattern matching operations in of speech recognition. Probability estimation is the local match, where comparisons are made between speech frames and spectra frames that used for training. As for decoding component, it can be viewed as a global match. The global match is a search for the best sequence of words and is determined by integrating many local matches.

Finally last block consists of language model, which determines the hypotheses that are considered in the global search. This block can further process the global decoder output. If the decoding block generates more than one most likely sentence, language model could re-score the sentence according to grammar or semantics.

SPHINX

To implement speech recognisation module, we use SPHINX 4.SPHINX provides more flexible framework for speech recognisation. This is written totally in java programming languages. when user speak into microphone then it need to convert into text, at that time sphinx modules on the processing side accept the word into audio form and convert it into text form. Figure 4: shows the basic flow diagram of Sphinx 4 implementation [2].

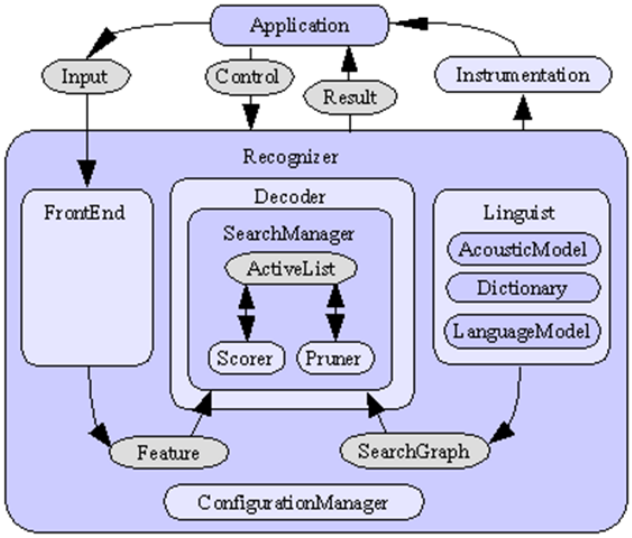

Fig.4. Basic Flow Diagram

The basic components of the Sphinx 4 model are as follows:

1. Input: The process starts with the voice input of the user, from the microphone of the mobile.
2. Configuration Manager: The configuration file is used to set all variables. These options are loaded by the configuration manager as the first step in any program.
3. Front End and feature: The front end is constructed, generating feature vectors from the input using the same process used during training.

International Journal of Electronics Communication and Computer Engineering
Volume 4, Issue 2, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

4. Decoder: The decoder constructs the search manager which in turn initializes the scorer, pruner and active list.
5. Result: In the final step, the result is passed back to the application as a series of recognized words. Once the initial configuration is complete, the recognition process can repeat without re-initializing everything [2].

Sphinx -4 Performance

Table 1: shows the performance details about SPHINX -4

Test	WER		RT		
	Sphinx -3.3	Sphinx -4	Sphinx -3.3	Sphinx -4 (1 CPU)	Sphinx -4 (2 CPU)
TI46 (11 words)	1.217	0.168	0.14	0.03	0.02
TIDIGITS (11 words)	0.661	0.549	0.16	0.07	0.05
AN4 (79 words)	1.300	1.192	0.38	0.25	0.20
RM1 (1000 words)	2.746	2.739	0.50	0.50	0.40
WSJ5K (5000 words)	7.323	7.174	1.36	1.22	0.96
HUB-4 (64000 words)	18.845	18.878	3.06	4.40	3.80

Table 1. SPHINX-4 Performance.

We were able to improve the runtime speed for the RM1 regression test by almost 2 orders of magnitude merely by plugging in a new Linguist and leaving the rest of the system the same. Word Error Rate (WER) is given in percentage. Real Time (RT) speed is the ratio of utterance duration to the time to decode the utterance. For both, a lower value indicates better performance. Furthermore, the modularity of Sphinx-4 also allows it to support a wide variety of tasks. For example, the various Search Manager implementations allow Sphinx-4 to efficiently support tasks that range from small vocabulary tasks such as TI461 and TIDIGITS2 to large vocabulary tasks such as HUB-4[12].

An interesting result of this comparison helps to demonstrate the strength of the pluggable and modular design of Sphinx-4. Sphinx-3.3 has been designed for more complex N-Gram language model tasks with larger vocabularies. As a result, Sphinx-3.3 does not perform well for “easier” tasks such as TI46 and TIDIGITS. Because Sphinx-4 is a pluggable and modular framework, we were able to plug in different implementations of the Linguist and Search Manager that were optimized for the particular tasks, allowing Sphinx-4 to perform much better. For example, note the dramatic difference in WER and RT performance numbers for the TI46 task [12].

IV. TEXT TO TEXT TRANSLATION

The output obtained from the speech recognizer is given to this module. In this translation recognized text is converted into destination language text. To translate the Source language text into destination language at that time internet connection is necessary. Text to text translation done through Google API. The output of this module is given to next text to speech synthesis module. Due to use of Google API there is no need of creating the database manually. Time required for searching equivalent destination language text is reduced.

V. TEXT TO SPEECH TRANSLATION

Speech synthesis is the artificial production of human speech. A computer system used for this purpose is called a speech synthesizer, and can be implemented in software or hardware products. A text-to-speech (TTS) system converts normal language text into speech. One of the key components of Text to Speech Synthesizer is prosody generator. There are basically two types of Text to Speech Synthesizer,

(i) single tone synthesizer
(ii) Multi tone synthesizer.

The basic difference between two approaches is the prosody feature. If the output of the synthesizer is required in normal form just like human conversation, then it should be added with prosody feature. The prosody feature allows the synthesizer to vary the pitch of the voice so as to generate the output in the same form as if it is actually spoken or generated by people in conversation [8]. In this module text translated into human speech form in the target language. Figure 5 shows the typical overview of Text To Speech system.

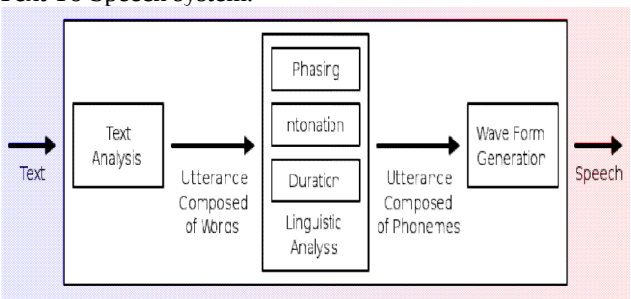

Fig.5. TTS system

Synthesized speech can be created by concatenating pieces of recorded speech that are stored in a database. Systems differ in the size of the stored speech units; a system that stores phones or diaphones provides the largest output range, but may lack clarity. For specific usage domains, the storage of entire words or sentences allows for high-quality output. Alternatively, a synthesizer can incorporate a model of the vocal tract and other human voice characteristics to create a completely "synthetic" voice Output [9]. A text-to-speech system (or "engine") is composed of two parts:[10] a front-end and a back-end. The front-end has two major tasks. First, it converts raw text containing symbols like numbers and abbreviations into the equivalent of written-out words. This process is often called text normalization, pre-processing, or tokenization. The front-end then assigns phonetic transcriptions to each word, and divides and marks the text into prosodic units, like phrases, clauses, and sentences.

The process of assigning phonetic transcriptions to words is called text-to-phoneme or grapheme-to-phoneme conversion. Phonetic transcriptions and prosody information together make up the symbolic linguistic representation that is output by the front-end. The back-end often referred to as the synthesizer then converts the symbolic linguistic representation into sound. In certain systems, this part includes the computation of the target prosody (pitch contour, phoneme durations), which is then imposed on the output speech [11].

VI. Conclusion

Human can interact with each other through natural language. If both people understood his languages then interaction between these two people done successfully. In this voice translation system there is no need of creation of database manually for matching/converting source text to destination text, due to this translation time will be reduced. SPHINX method is more suitable for speech recognisation. From the comparison between techniques in speech recognition, Sphinx model is identified as one of the popular connectionist techniques and suitable to use in speech recognition. We plan to develop two way translations like Hindi to English and English to Hindi. Finally, the Real time voice translation system is done in this way.

References

[1] Aakash Nayak, Santosh Khule, Anand More, Avinash Yalgonde, Dr. Rajesh S. Prasad, "Study of various issues in voice translation" International Journal of Advanced Research in Computer Engineering & Technology, Volume 2, Issue 1, January 2013,ISSN: 2278 – 1323

[2] Yen Chun Lin, " An optimized approach to voice translation on mobile phones", IEEE Transaction on Voice Recognition, ISSN: 1002 – 1989, Volume 2, Issue 5- 2011

[3] Bowen Zhou, Yuqing Gao, Jeffrey Sorensen, Daniel D´echelotte and Michael Picheny, "A Hand-Held Speech-to-Speech Translation System", The DARPA BABYLON project, 0-7803-7980-2/03/$17.00 © 2003 IEEE, ASRU 2003

[4] Kishore Prahallad, "A Direct Approach for Speech to Speech Translation 11-731 Project Report" Language Technologies Institute, Carnegie Mellon University

[5] B. H. Juang and L. R. Rabiner, "Automatic Speech Recognition--A Brief History of the Technology", Elsevier Encyclopedia of Language and Linguistics, Second Edition, 2005

[6] Hesham Tolba & Douglas O'Shaughnessy, "Speech Recognition by Intelligent Machines", IEEE Canadian Review – Summer, 2001

[7] Morched Derbali, Mu'Tasem Jarrah, Mohd Taib Wahid, "A Review of Speech Recognition With SPHINX Engine In Language Detection", Journal of Theoretical and Applied Information Technology, June 2012. Volume. 40 Issue.2, ISSN: 1992-8645

[8] M.B. Chandak, Dr.R.V. Dharaskar & Dr.V.M.Thakre "Text to Speech Synthesis with Prosody feature: Implementation" International Journal of Computer Science and Security, Volume 4 Issue (3) of Emotion in Speech Output using Forward Parsing

[9] Rubin, P.; Baer, T.; Mermelstein, P. (1981). "An articulatory synthesizer for perceptual research". Journal of the Acoustical Society of America 70 (2): 321–328. doi:10.1121/1.386780

[10] van Santen, Jan P. H.; Sproat, Richard W.; Olive, Joseph P.; Hirschberg, Julia (1997). Progress in Speech Synthesis. Springer. ISBN 0-387-94701-9.

[11] Van Santen, J. (April 1994). "Assignment of segmental duration in text-to-speech synthesis". Computer Speech & Language 8(2): 95–128. doi:10.1006/csla.1994.1005.

[12] Willie Walker, Paul Lamere, Philip Kwok, Bhiksha Raj, Rita Singh, Evandro Gouvea, Peter Wolf, Joe Woelfel " Sphinx-4: A Flexible Open Source Framework for Speech Recognition" SMLI TR2004-0811 C2004 SUN MICROSYSTEMS INC.

Author's Profile

Mr. Danny J. Pereira
Assistant Professor, Department of Computer Engineering, Government College of Engineering & Research, Pune University, Awasari (kd) Dist. Pune, Maharashtra, India

Mr. Vishal L. Jadhav
Bachelor of Computer Engineering,
Government College of Engineering & Research, Pune University, Awasari (kd) Dist. Pune, Maharashtra, India

Mr. Jaydeep C. Joshi
Bachelor of Computer Engineering,
Government College of Engineering & Research, Pune University, Awasari (kd) Dist. Pune, Maharashtra, India

Mr. Prashant S. Bhosale
Bachelor of Computer Engineering,
Government College of Engineering & Research, Pune University, Awasari (kd) Dist. Pune, Maharashtra, India

Mr. Swapnil G. Dhalpe
Bachelor of Computer Engineering,
Government College of Engineering & Research, Pune University, Awasari (kd) Dist. Pune, Maharashtra, India