2019 IEEE International Symposium on Technology in Society (ISTAS) Proceedings
Miriam Cunningham and Paul Cunningham (Eds)
ISBN: 978-1-7281-5480-0

Machine Translation of English Videos to Indian Regional Languages using Open Innovation

Srikar Kashyap Pulipaka[1], Chaitanya Krishna Kasaraneni[2], Venkata Naga Sandeep Vemulapalli[3], Surya Sai Mourya Kosaraju[4]
[1]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India, Email: srikar.kashyap@gmail.com
[2]Department of Computer Engineering, San Jose State University, San Jose, CA, United States, Email: chaitanyakrishna755@gmail.com
[3]Synopsys Inc, Hyderabad, India, Email: vemulapalli1996@gmail.com
[4]Department of Computer Science, University of Birmingham Dubai, Email: suryamkosaraju@gmail.com

***Abstract* - In spite of many languages being spoken in India, it is difficult for the people to understand foreign languages like English, Spanish, Italian, etc. The recognition and synthesis of speech are prominent emerging technologies in natural language processing and communication domains. This paper aims to leverage the open source applications of these technologies, machine translation, text-to-speech system (TTS), and speech-to-text system (STT) to convert available online resources to Indian languages. This application takes an English language video as an input and separates the audio from video. It then divides the audio file into several smaller chunks based on the timestamps. These audio chunks are then individually converted into text using IBM Watson's speech-to-text (STT) module. The obtained text chunks are then concatenated and passed to Google's machine translate API for conversion to the requested Indian language. After this translation, a TTS system is required to convert the text into the desired audio output. Not many open source TTS systems are available for Indian regional languages. One such available application is the flite engine (a lighter version of Festival engine developed by Prof. Alan Black at Carnegie Mellon University (CMU)). This flite engine is used as TTS for generating audio from translated text. The accuracy of the application developed can be as high as 91 percent for a single video and averages about 79 percent. This accuracy is verified by comparing naturality of the audio with the general spoken language. This application is beneficial to visually impaired people as well as individuals who are not capable of reading text to acquire knowledge in their native language. In future, this application aims to achieve ubiquitous communication enabling people of different regions to communicate with each other breaking the language barriers.**

Keywords - machine translation, text-to-speech, speech-to-text, concatenation, open innovation.

I. Introduction

India is a multilingual country with over 122 major languages and 1599 other languages. Of all these languages spoken over different parts of India, only twenty-two were referred to as scheduled languages and given recognition. However, most informational resources available over the Internet are in one global language – English, which cannot be understood by many people. Translating these online resources manually into regional languages is time-consuming and tedious. Machine translation [1], a sub-division of Artificial Intelligence (AI), offers a solution to this problem. Machine translation helps to translate these online resources to the regional language with minimal human involvement. The main principle of machine translation is to translate text from one human-understandable language to another human-understandable language [1]. Machine translation (MT) is based on Natural Language Processing (NLP), a prominent research area. NLP, also called Computational Linguistics, takes a natural language text/speech as input, synthesizes this input and explores meaningful and useful insights from the input. Translation is encouraged and widely accepted by the Indian government since this helps people to expand their knowledge. This machine translation can be used to translate the contents of many domains, such as sports, history, business, health, literature, news and social media. In this paper, open innovation [2] is used for translation i.e., using already existing software applications at each phase to make the translation process much easier.

Text output from videos are a good way to translate video content between languages through automated closed captions. To reach a wider audience, including visually impaired individuals and individuals incapable of reading text, the text can be converted into audio output. This can be done using a text-to-speech software.

The task of a text-to-speech (TTS) [1] software is to convert any given input text into system-generated speech, which can be applied in many ways, such as reading the news

from webpages, SMSs, emails, etc. The advantages of TTS systems include i) reducing strain on eyes, and ii) aiding the visually impaired people iii) helping the illiterate people gain knowledge. These TTS systems are stratified into three categories: "articulatory synthesis; formant synthesis; concatenative synthesis" [3]. Each of the above categories has its own advantages as well as limitations.

In this paper, English videos are machine translated into Indian regional languages using the pipeline of pre-existing software applications such as ffmpeg library of UNIX operating system, speech-to-text module of IBM Watson supercomputer, Google translate API, and flite engine developed at Carnegie Mellon University. This project was initially developed by a team of four undergraduate students as a solution to one of the challenges in the Communications vertical of "Open Innovation hackathon for Smart Cities" by Berkeley-Haas [20], University of California - Berkeley in collaboration with Andhra Pradesh State government organized at Koneru Lakshmaiah Education Foundation, Guntur, India in July 2017. This proposed solution won the first position in the Communications vertical of the competition.

Section 2 discusses the previously developed related works. Section 3 discusses the proposed architecture while section 4 contains result analysis. Section 5 concludes the paper providing recommendations for the future developments considering some inputs taken from villagers when this application was shown to them.

II. Related work

"Machine translation can be done in three ways: Direct machine translation, Rule based machine translation and Corpus or Empirical based machine translation" [4]. The first effort of translating English to Indian regional languages was made in 1995 when a group of Professors, RMK Sinha et al., at IIT Kanpur, India developed a software application named AnglaBharti [5]. This application was developed using transfer based and interlingua method. Pseudo target language is generated by the system for the English text passed through the application, and this pseudo language is converted into Indian regional language with the help of text generation module. The post-editing purpose requires human involvement to check and correct grammatical errors if any. However, to reduce human involvement in post-editing procedure of the output that is generated by this application, IIT Kanpur and IIIT Hyderabad, India have developed another application called "Anusaaraka" in the same year. This application "works on language knowledge, word knowledge and statistical knowledge." [6].

A statistical framework for machine translating English documents to Malayalam was proposed by Mary Priya et al. This framework needs a bilingual dictionary and Malayalam corpus [7]. This framework uses "suffix separation and stop word elimination" [7] for reducing the complexity of training dataset. However, this framework consumes a huge amount of memory in order to store Malayalam corpus. To overcome this space complexity, Remya Rajan et al. proposed a "rule-based machine translation technique" for translation of English to Malayalam [21]. To obtain a target tree, transfer-link rules were developed. In the similar way, to assign morphological features, morphological rules were derived from the given input. "Roman to Unicode file, Unicode to Roman file, word dictionary file, morph dictionary file, and transfer links" are the major factors involved in this way of rule-based machine translation [21].

While the above machine translation system involves Roman and Unicode files, a morphological analyzer for the Malayalam language was proposed by Rinju O.R. et al. [10]. This analyzer takes the input, processes the input, and retrieves grammatical information like morphemes. It also provides information about the speech like tense of the verb, gender of the speaker, numbers, and case information of the noun. For analysis accentuation, a list is derived from the information retrieved. It can be inferred from the results that a rule-based approach is better than the probabilistic method.

Focusing on translation in sports domain, mainly cricket, and using the transfer-based approach, Aasha V C and Amal Ganesh developed a translation method from the English language to the Malayalam language [11]. To improve the quality of translation, 44 rules are developed. "Each word in the given English input sentence is tokenized and a parse tree is formed using them. Using the breadth first search (BFS) order, rules are applied on the parse tree to obtain the target language. The lengthier sentences sometimes generate incorrect outputs, and this is can be handled by dividing the longer sentences into shorter ones" [11]. There are several similar machine translation proposals and applications for the Malayalam language.

Apart from the machine translation applications developed for Malayalam, there are systems for each Indian language. In 2014, Keerthi Lingam et al. proposed "Rule-based machine translation from English to Telugu with emphasis on prepositions." [18]. This method mainly focuses on handling prepositions during system translation of documents in English language to Telugu language. This is because there are multiple meanings for the same preposition in Telugu. Factors like "time, gender, context, and many other features play an important role in translation" [18]. As the title suggests, this is

developed on a rule-based approach. For this reason, a finite set of production rules, and English to Telugu dictionary based on these rules are developed. To identify text in diagrams and flowcharts, "deformatting and reformatting" [18] are used.

In 1799, a Danish scientist Christian Kratzensteim was the first to work on text-to-speech systems. His research marked the beginning of TTS evolution. Kratzensteim built a "model of human vocal tract", which produces vowels. Thereafter, with a lot of research, many improvements were made. In 1981, P. Rubin and Thomas Baer initiated an articulatory synthesizer for perpetual research [12]. "In this approach, human vocal tracts are labelled with six articulators like tongue tip, tongue body, velum, lips, jaw and hyoid bone position which will outline the vocal tract." [13]. From these parameters, "area and width functions" [13] are calculated. After calculation of these functions, "speech is produced by digital filter representation of the given vocal tract transfer functions [13]."

In 2001, Alan Black et al. developed a small fast run-time synthesis machine named "Flite" (pronounced as "F"-lite standing for Festival Lite), which is designed as an alternative for Festival platform. This is a TTS synthesizer application for many Indian languages [14]. As the name suggests, it is much faster and smaller in size compared to the Festival system [14]. This Festival speech system also was originally developed by Alan Black and some of his colleagues in Center for Speech Technology Research (CSTR) at the University of Edinburgh. These combinedly are released as a festvox project. Later, a Silicon Valley entrepreneur Suresh Bazaj in collaboration with Prof. Alan Black empowered a non-profit initiative, Hear2read, which aids the visually impaired (VI) people to gain knowledge by converting the visual text into audio and reads the website contents. This initiative utilizes the flite TTS software [14]. According to Suresh, "Only 10% of blind children in India get any type of formal education. The rest are either are too poor or do not have access to institutions that can teach VI children" [17].

Of the many TTS systems developed for Indian languages, some of the prominent ones include Sruthi, a TTS system developed at IIT Kharagpur for Hindi and Bengali languages. Vani is another TTS software application developed at IIT Bombay. Swaram is another TTS system that accepts Malayalam text and produces speech. A "multilingual newspaper reading system" is developed to read the contents of newspapers and this system supports Hindi, Malayalam, Tamil, and English [19]. In these systems, there is the flexibility for the user to select the required language, favorite topics or domains in the newspapers. The main challenge in developing this system was maintenance of a single database to handle memory and develop translation rules for each language.

In the recent years, TTS systems for optical character recognition (OCR) documents have been developed, which can be used to read OCR documents of many languages and generate speech from the text in OCR documents. The text is digitalized using OCR and then transformed to a grayscale image. These images are divided into segments to obtain relevant features [15]. A TTS system was developed for the Tamil language in 2014 that consists of "text analysis, syllabication, Letter-to-sound rules and concatenation" [16]. Phonemes are mapped to words and a database of these phonemes is developed. Then, to locate the best possible order of speech, a joining cost is computed. Akshara-to-font mapping, pronunciation of aksharas (letters) and normalizing text are the major issues that arise during machine translation to Indian regional languages. The "term frequency - inverse document frequency (TF-IDF)" is applied to identify different types of fonts. Similarly, in 2016, Ancy Anto and K.K. Nisha, students of Computer Science department at Rajiv Gandhi Institute of Technology developed a "Text to Speech Synthesis System for English to Malayalam translation." [13]

Open Innovation [2]:

Open innovation [2] is "a paradigm that assumes that firms can and should use external ideas as well as internal ideas, and internal and external paths to market, as the firms look to advance their technology" [2]. This term was first coined and promoted by Henry Chesbrough, a professor in the Haas School of Business at University of California, Berkeley [2].

There are certain advantages with open innovation. It reduces cost of research, improves productivity, and enhances viral marketing, digital transformation, and scope for new business models. Open innovation also has certain disadvantages like losing confidentiality of information and losing competitive advantage for revealing intellectual property [2]. Open innovation is classified into certain models depending upon the circumstances of its usage. Examples are the Government driven model, product platforming model, etc." [2]. Open innovation is an ecosystem consisting of three parts, which describe "open innovation, innovation systems and business ecosystems" [2].

Open source and open innovation are not similar and have conflicts in terms of patent rights. "This conflict is particularly apparent when considering technologies that may save lives, or other open-source-appropriate technologies that may assist in poverty reduction or sustainable development" [2]. These two terms are not mutually exclusive. Some companies and open source initiatives merge into these two concepts. Example is IBM where the company presents its Eclipse platform as an open innovation case when companies are inside open

innovation network. This project uses both open innovation technologies and open source software applications.

III. PROPOSED ALGORITHM

There are several machine translation (MT) and text-to-speech (TTS) applications developed for Indian regional languages. However, they were developed for individual languages and this proposed system is a collective software application where the user can choose the desired language among Bengali, Gujarati, Hindi, Kannada, Marathi, Punjabi, Tamil and Telugu. As for now, the proposed system only supports these languages.

Of the many text-to-speech (TTS) systems developed, the flite engine is only used in this project as it is faster in terms of execution time, supports parallel processing and is also lighter in terms of memory usage. The remaining machine translation applications consume more memory and are also slower compared to this flite engine developed by Prof. Alan Black and his colleagues.

As mentioned earlier, this is a basic version developed and tested on the Linux and MacOS operating systems. Figure 1 shows the proposed system architecture. This system is developed completely using open innovation i.e. by utilizing the open source tools and some open libraries of IBM Watson and Google Translate.

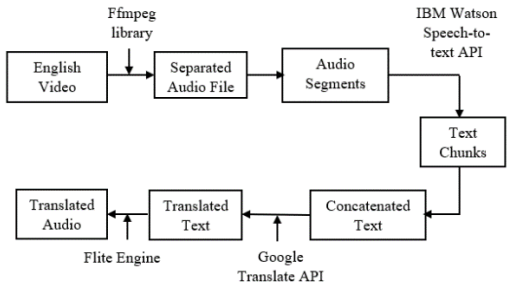

Fig. 1 Architecture of Machine Translation of English Videos to Indian Regional Languages using Open Innovation

A. Requirements

To test this project, there are certain requirements:

- Load the system/laptop with UNIX/LINUX or MacOS operating system with python 3 packages pre-loaded.
- Install the ffmpeg library of UNIX system.
- Get the API keys for IBM Watson speech-to-text module and Google translate.
- Install the flite engine. This can be done by cloning the GitHub repository and following the given steps in the main page.

B. Phases of Machine Translation

As seen in the architectural diagram, this project has four phases:

- Audio separation
- Speech-to-text (STT) conversion
- Translation to desired language
- Text-to-Speech (TTS) conversion

i. Audio Separation

In this phase of translation, the input is English video. From this video, **the audio file is separated using the ffmpeg UNIX library**. This ffmpeg library is a command line multimedia framework that allows the user to convert multimedia files between formats i.e. a ".mp4" file can be converted into a ".mp3" file. It is one of the leading frameworks that has the capacity to "decode, encode, transcode, mux, demux, stream, filter and play" any media file that is created by humans or machines. This ffmpeg library also supports many legacy formats and is highly portable.

The ffmpeg library is open sourced, and anybody can contribute and use it. It is even compatible with many external libraries like AMD Advanced media framework (AMF), AviSynth, codec2, etc. enabling this library to generate audio fingerprints, decoding, encoding, etc.

In summary, the input for the first phase in translation is an English Video and the output is a separated audio file.

ii. Speech-to-text (STT) conversion:

In this phase, the audio file from the previous step is divided into chunks of fifteen seconds for parallel and faster processing. The number of chunks obtained depends on the length of the video given as input. These chunks are then parallelly passed to IBM Watson supercomputer to obtain text using the API. For this, an IBM Bluemix account is required. Create IBM Bluemix account and obtain the API key and URL by following the instructions given there. IBM Watson is a supercomputer, a kind of question-answering system that has capacity that for natural language processing. It was developed as a part of IBM's DeepQA project.

This IBM Watson system runs on Linux enterprise edition version 11 operating system and provides distributed computing using the Apache's Hadoop Framework. After the application passes the chunks of audio to speech-to-text module, these audio chunks are converted into different text segments. As per order of the audio files, these text segments are concatenated and stored as a concatenated text file. In this paper, IBM Watson is used as its response time for speech-to-

text conversion is faster when compared to other speech-to-text engines.

In this phase, parallel processing is used while passing the audio chunks to the speech-to-text module as it saves a lot of time. If the audio chunks are passed sequentially, this process consumes about 70 percent more time than parallel processing. Figure 2 is shows comparison between processing times of parallel and sequential audio processing.

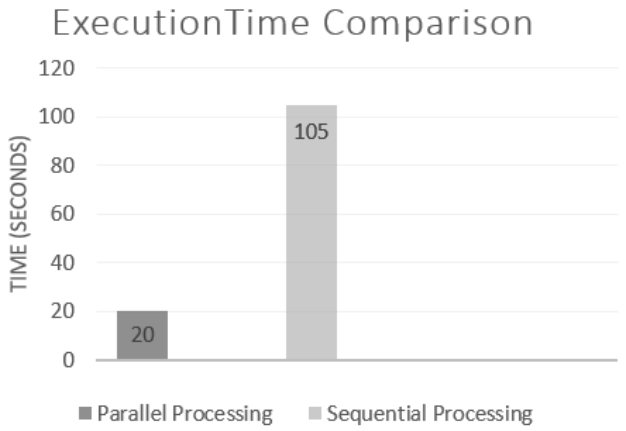

Fig. 2 Execution Time Comparison for Parallel Processing and Sequential Processing of the Audio Chunks

In summary of this phase, the input is the audio file obtained from previous phase and output is a concatenated text file, which contains the contents of audio input in text format. Until this phase, the outputs are in English only.

iii. Text translation

In this phase, the translation of language takes place. i.e. English is translated into the required regional languages such as Bengali, Gujarati, Hindi, Kannada, Marathi, Punjabi, Tamil or Telugu selected by the user at the time of giving input, using the Google translate API. For this step, the API key of Google translate is required. This can be obtained from the cloud transactions API section in the new project created from the user's Google account developers console. This will provide the user with a key for machine translation.

The English text output file in the previous phase is passed to this machine translation API of Google. Google translate software takes care of the machine translation from English to the required language. Here Google translate is used, as the translation accuracy of this software is higher compared to the other machine translation applications available.

In the third phase, the English text file is passed through machine translation API of Google and the translation output is stored as another text file. This file consists of text in required language which may be Bengali, Gujarati, Hindi, Kannada, Marathi, Punjabi, Tamil or Telugu.

iv. Text-to-speech conversion

This is the last phase in translation. For this phase, the open source software named Flite engine developed by Alan Black et al., Professors at Carnegie Mellon University (CMU) is used to convert the text file to audio file.

The flite engine consists of fundamental objects that reflect the Edinburgh speech tools which is also known as C Speech Tools (CST). This flite engine requires three major parts to build the complete synthesizer. They are:

- **language model** [14]**: "**this part provides the basic rules for tokenization, text analysis, phoneset, prosodic structures, etc." [14]
- **lexicon** [14]**:** a pronunciation model for letter to sound rules which depends on the domain and inventory of the language. [14]
- **voice:** this depends on the primitives supplied by the language model. This includes the prosody models and voice itself. [14]

These three sections are compiled as different libraries and the first two can be shared among the different voices of same language.

This flite engine was developed to overcome the time and space complexities of the Festival speech synthesis system which was developed using C++ language. Flite is developed from festival system and festvox project.

In summary, the regional language text output of the previous phase is passed as input to the festival lite (flite) engine to get a system generated audio output of the given text.

IV. Result Analysis

Here, machine translating English videos to Indian Regional languages is implemented using open innovation. Many open source libraries and APIs are used in this process. For evaluating the machine translation system, various videos containing narrations from various domains are given as input and tested. The accuracy of the system is tested using the naturalness of the sentence i.e. how close is a sentence in the video to the natural spoken language.

Table 1 shows the evaluation of proposed system for machine translating English videos to Telugu only. For example, consider an English video that describes the biography of Adolf Hitler, a German politician and leader of the Nazi Party. This video is a narrative video that gives a brief

idea about Hitler, his rise and fall, his policy making skills, and how he dealt with people.

TABLE I: EVALUATION OF THE PROPOSED SYSTEM

Count of videos taken to translate	15
Total number of minutes (combining all videos)	253
Total output minutes (combining all videos)	364
Average naturalness of sentences generated.	79%

Sample English Text from the video:

Entered Clean Text

most on Paris once again Napoleon was narrowly defeated at the battle of Waterloo and he was imprisoned in exile to Helena in the Atlantic and he died there in eighteen twenty one.

Entered Clean Text

at the time British naval power prevented Napoleon from invading England so he planned to invade Egypt and unite with his allies to defeat the British and takeover British trade routes with India in seventeen ninety eight Napoleon's armies defeated the Egyptians but later that year the British captured and destroyed.

Fig. 3 Sample English Text obtained from the video using Speech-to-text module of IBM Watson Supercomputer

Sample Machine Translated Telugu Text from the video:

కానీ చరిత్రలో గొప్ప సైనిక నాయకులలో ఒకడు మరియు ముందు చక్రవర్తిగా ఉన్నారు అతను తన సైన్యాలను యూరప్ అంతటా విజయవంతం చేసేందుకు నెపోలియన్ కోర్సికాలో మిలటరీ అకాడమీలో చదువుకున్నాడు మరియు తరువాత ఇటలీలో ఫ్రెంచ్ సైన్యాన్ని కైవసం చేసుకున్నాడు, ఇటలీలో ఆస్ట్రియన్ల విజయానికి ఫ్రాన్స్ సైన్యాలకు నాయకత్వం వహించి ఇటలీలో శాంతి చర్చలు పోరాడుతున్న వర్గాల మధ్య బ్రిటీష్ నావికా దళం నెపోలియన్ను ఇంగ్లాండ్ను ఆక్రమించకుండా అడ్డుకుంది. తద్వారా అతను ఈజిప్టును ముట్టడించేందుకు మరియు తన ముస్లిం మిత్ర పక్షాలతో బ్రిటీష్వారిని ఓడించటానికి మరియు పదిహేను ఎనిమిది ఎనిమిది నెపోలియన్ సైన్యాల్లో భారతదేశానికి బ్రిటీష్ ట్రేడ్మార్గాలను స్వాధీనం చేసుకుని ఐగుప్తీయులను ఓడించాడు. బ్రిటిష్ వారు స్వాధీనం చేసుకున్నారు నెపోలియన్ యుద్ధ నౌకల యుద్ధంలో నౌకల యుద్ధంలో ఫ్రెంచ్ నౌకలో రెండింటిలోనూ నౌకాదళం యొక్క ఆధిక్యతలను ఎదుర్కోవడంతో అతను తన దళాలలోని పదమూడు వేలమంది డమాస్కస్కు చేరాడు, ఇక్కడ అతను అనేక తీరప్రాంత పట్టణాలను నెపోలియన్ యొక్క మధ్యప్రాచ్యం యొక్క ఆక్రమణను అనారోగ్యంతో మరియు అనారోగ్యంతో అందువలన అతను ఒక కు వెనక్కి పద్దెనిమిది తొమ్మిది తొమ్మిది మందిలో ఫ్రాన్సుకు తిరిగి రావడంతో అతను తన సొంత ఎన్నికను పద్దెనిమిది ఓహ్లో మొదటిసారి కాన్సుల్గా నియమించాడు, అతడు బానిసత్వం మరియు ఫ్రెంచ్ వలసరాజ్యాలను పునఃస్థాపించాడు, హైతీలో బానిస తిరుగుబాటుకు దారితీసింది మరియు తరువాతి వివాదం నెపోలియన్ దళాలు ఓడించబడ్డాయి, ఉత్తర అమెరికా లూసియానాలో సంయుక్త రాష్ట్రాలకు భూభాగాలు పద్దెనిమిది ఓహ్ నాలుగు నెపోలియన్లో చక్రవర్తి ఫ్రాన్స్ను నియమించాయి మరియు మరోసారి బ్రిటన్కు దళాలు చేరేందుకు రష్యా మరియు ఆస్ట్రియాకు చెందిన నెపోలియన్ నౌకాదళం ట్రఫాల్గర్ యుద్ధంలో ఓడిపోయింది కానీ ఆరు వారాల తర్వాత అతని సైన్యం ప్రముఖంగా ఆస్ట్రో రష్యన్ ఆస్టెర్లిట్జ్ యుద్ధంలో ఉన్న దళాలు అతను ప్రస్తుతం పశ్చిమ ఐరోపాలో చాలా వరకు నియంత్రించబడుతున్న విక్టోరియన్లో కొత్త భూభాగాన్ని సంపాదించాడు, అతను విజయం సాధించిన జ్ఞాపకార్థం ఆర్క్ ట్రైమ్ఫ్ను నియమించాడు, నెపోలియన్లో పవిత్ర రోమన్ సామ్రాజ్యం కరిగినది తన బంధువులు మరియు విశ్వాసపాత్రులను ఐరోపా అంతటా నెపోలియన్ దళాల పద్నాలుగు ది అప్పుడు నకిలీలు, తరువాత నలభై వేల వేల మంది సైనికులను ఫ్రెంచ్ సైన్యం నుండి పద్దెనిమిదవ పన్నెండు లో రష్యా దండయాత్ర చేసాడు, కేవలం నలభైవేల మంది రష్యా నుండి వెనక్కి తీసుకున్న ఒక అభిప్రాయాన్ని తట్టుకుని, మిత్రరాజ్యాలు ఒక చిన్న ప్రవాస నెపోలియన్ తప్పించుకుని, పారిస్లో చాలామంది నెపోలియన్ వాటర్లూ యుద్ధంలో నెమ్మదిగా ఓడిపోయాడు మరియు అట్లాంటిక్లో హెలెనాకు చెరసాలలో బంధించబడ్డాడు మరియు అతను 18 వ శతాబ్దంలో

Fig. 4 Sample Machine Translated Telugu Text generated using Google Translate API

In figure 3, a sample English text obtained using IBM Watson API is shown. If observed, the image contains a phrase "Entered Clean Text", which means that the part of text is from the audio chunk in that time interval order. This text is concatenated and passed to the next phase. Figure 4 contains the sample text of Indian regional language Telugu which is obtained by applying machine translation using Google translate API. This Telugu text is the input for Text-to-speech system, here the flite engine, which generates the audio output of the regional language.

When this application was shown and tested with a group of villagers from Revendrapadu, a village near Koneru Lakshmmaiah University (KLEF), they appreciated the efforts taken to develop this application. In addition, they requested to develop an extension for this application that would aid them in communicating with people who speak other languages, and this would be helpful for them during travel to other places.

There were some issues while building the pipeline of above-mentioned stages. One was selecting an accent for the target language. This was addressed by taking suggestions from the villagers. Their suggestion was to select the common accent that is used in movies as it will be understood by many people.

V. Recommendations for Future

Limitations and Future scope of the proposed system:

There is a vast scope for improvement and research in this kind of applications. In the proposed system, even though the average accuracy is 79 percent, it is only for the single speaker/voice input videos (monologues or narrations). If a video with multiple speakers i.e. a video having lengthy conversations is given as input, the output audio returned consists of only a single voice.

In the future, the system can be improved to overcome this limitation by mapping the timestamps with the number of different voices and using different voices for text in that time interval. It can be further extended by adding moods to the voices so that the original emotion in the source video is not taken away. This system can also be used to improve ubiquitous communication, one of the major aspects in Artificial General Intelligence (AGI). As the current proposed system is limited in translation of videos from English to languages like Bengali, Gujarati, Hindi, Kannada, Marathi, Punjabi, Tamil or Telugu, it can be extended to other Indian languages.

VI. Conclusion

Natural Language Processing (NLP) is currently one of the most prominent technologies that every technology enthusiast is interested to work on. The role of an NLP developer is to explore the depths and develop interesting applications. In this work, a machine translation application is developed for translating English videos into Indian regional languages utilizing the available open source tools. This system enables visually impaired individuals and individuals who lack a full education to gain knowledge by listening to the text content from web sources like Wikipedia, Quora, etc. or video content available in sources like YouTube, Massive Open Online Courses (MOOCs) (coursera, edX, etc.). It also helps people living in rural areas to understand content in their native languages. At present, the ubiquitous communication extension for this application is being developed. This extension aims at

conversion of voice input of one language to another instantly allowing travelers to communicate with local people. In future, the application will be developed so that it has the capability to translate even conversational videos into regional languages.

ACKNOWLEDGMENT

Thanks to Prof. Henry for promoting open innovation, which allows the companies and other organizations to share their domain knowledge allowing them to collaborate and solve the problems like poverty, illiteracy, etc. in the world. The author would like to thank Professor Alan Black for developing the festival lite (flite) text-to-speech engine as this system reduces the time taken for producing the audio output from regional language text. Also, thanks to Mr. Suresh Bazaj, a silicon-valley based entrepreneur, for responding immediately and providing references for the flite engine. Thanks to Mr. Shyam Krishna for his guidance in installation and usage of the flite engine.

REFERENCES

[1] R.M.K. Sinha, K. Sivaraman, Aditi Agrawal, Renu Jain, Rakesh Srivastava, Ajai Jain, "ANGLABHARTI: A Multilingual Machine Aided Translation Project on Translation from English to Indian Languages", *Proc. IEEE*, 1995. Available: IEEE https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=538002&tag=1

[2] Henry William Chesbrough, "Open Innovation – The New Imperative for Creating and Profiting from Technology", 2nd Edition, March, 2003. [Online]. Available: Open Library https://openlibrary.org/books/OL8754224M/Open_Innovation

[3] D.Sasirekha, E.Chandra, "Text To Speech: A Simple Tutorial," International Journal of Soft Computing and Engineering, ISSN: 2231-2307, vol. 2, March 2012.

[4] Sugata Sanyal, Rajdeep Borgohain, "Machine Translation System in India," ICSCN 2015.

[5] R.M.K. Sinha, K. Sivaraman, Aditi Agrawal, Renu Jain, Rakesh Srivastava, Ajai Jain, "ANGLABHARTI: A Multilingual Machine Aided Translation Project on Translation from English to Indian Languages", *Proc. IEEE*, 1995. Available: IEEE https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=538002&tag=1.

[6] Akshar Bharati, Vineet Chaitanya, Amba P. Kulkarni and Rajeev Sangal, " ANUSAARAKA: Machine Translation in Stages", A Quaterly in Artificial Intelligence, Vol. 10, No. 3, July 1997.

[7] Mary Priya Sebastian, Sheena Kurian K and G. Santhosh Kumar, "A Framework of Statistical Machine Translator from English to Malayalam," Proc. 1st Amrita ACMW Celebration on Women in Computing, September, 2010.

[8] Sanjay Kumar Dwivedi and Pramod Premdas Sukhdev, "Machine Translation in Indian Perspective", Journal of Computer Science, june 2010.

[9] B. Hettige, A. S. Karunananda, "Existing Systems and Approaches for Machine Translation: A Review, " Srilanka Association for Artificial Intelligence, Eight Annual Session 2011

[10] Rinju O.R., Rajeev R. R., Reghu Raj P.C., Elizabeth Sherly, " Morphological Analyzer for Malayalam: Probabilistic Method Vs Rule Based Method", International Journal of Computational Linguistics and Natural Language Procsseing, ISSN: 2279- 0756, vol. 2, Issue 10, October 2013.

[11] Aasha V C, Amal Ganesh, "Machine Translation from English to Malayalam Using Transfer Approach," IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2015.

[12] Philip Rubin, Thomas Baer, "An Articulatory Synthesizer for Perceptual Research", Bell-Northern Research and INRS Telecommunications. Available: [Online] https://pdfs.semanticscholar.org/2b5c/3ed2d95abf3f11df47385e76032a32e130a7.pdf

[13] Ancy Anto, K. K. Nisha, "Text to speech synthesis system for English to Malayalam translation", *International Conference on Emerging Technological Trends (ICETT)*, October 2016, Available: IEEE https://ieeexplore.ieee.org/document/7873642?cv=1

[14] Alan W Black and Kevin A. Lenzo, Home Page- Dr. Alan W Black, 2001. [Online]. Available: http://www.cs.cmu.edu/~awb/papers/ISCA01/flite.pdf

[15] Priyanka Jose, Govindaru V, "Malayalam Text-to-Speech System", International Journal of Engineering and Technical Research (IJETR), ISSN: 2321-0869, vol-1, issue-3, May 2013.

[16] M.Karthikadevi, Dr.K.G.Srinivasagan, "The Development of Syllable Based Text to Speech System for Tamil language", IEEE International Conference on Recent Trends in Information Technology, 2014.

[17] https://hear2read.org/about.php

[18] Keerthi Lingam, E. Rama Lakshmi and L. Ravi Teja, "Rule-based Machine Translation from English to Telugu with Emphasis on Prepositions", International Conference on Networks & Soft Computing, 2014. Available [Online] IEEE: https://ieeexplore-ieee-org.libaccess.sjlibrary.org/stamp/stamp.jsp?tp=&arnumber=6906669

[19] Jose Stephen, Anjali.M, Bhadran V.K, "Voic e Enabled Multilingual Newspaper Reading System", in Proc. IEEE, 2013.

[20] https://corporateinnovation.berkeley.edu/2017/09/open-innovation-hackathon-smart-villages/

[21] Remya Rajan, Remya Sivan, Remya Ravindran, K.P Soman, "Rule Based Machine Translation from English to Malayalam", International Conference on Advances in Computing, Control and Telecommunication Technologies, 2009.