Speech to Speech Translation using Encoder Decoder Architecture

Rohan Nakave[1], Zarana Prajapati[2], Pranav Phulware[3], Prof. Akshay Loke[4]

[1,2,3]Students of Department of Information Technology, Vidyalankar Institute Of Technology, Mumbai, Maharashtra
[4]Prof. of Department of Information Technology, Vidyalankar Institute Of Technology, Mumbai, Maharashtra

---***---

Abstract - *In this paper, we address the task of Audio language translation. We present a Two-way translation method for translating spoken sentences from English language into spoken sentences in another language and vice versa. In this device it consists of mainly three modules speech recognition in English language, text to text translation from English to other language and other language speech generation. It first recognize the speech (in English) using speech recognition and displays the text on the screen and then translates the text from English (text) into the desired language(text) and displays it on the screen, after that it converts into desired language (speech) which can heard at the other end of the device.*

Key Words: Speech Recognition, Speech Generation, Audio language translation, Two-way translation method, text to text translation.

1.INTRODUCTION

Nowadays with people's emerging interest in travelling around the world and meeting and interacting with different types of people with different culture and tradition, most of the people face the difficulty in communicating and speaking with other people, who does not know their languages. To avoid these difficulties the developed system records the interaction and translate it into the desired languages.

Speech-to-speech translation technology represents a technology which automatically translates one language to another language in order to enable communication between two parties with different native languages [1]. Speech to speech translation systems are developed over the past several decades with the goal of helping folks that speak different languages to talk with each other. These systems have usually been broken into three separate components: automatic speech recognition to transcribe the source speech as text, MT to translate the transcribed text into the target language, and text-to-speech synthesis (TTS) to urge speech within the target language from the translated text. Additionally, the technology helps to understand and recognize the native language and the user interface-related technology integrated with the UI also play a crucial role during this speech-to-speech translation system from known language to the target language.

1.1. LITERARTURE SURVEY

In [1] they had proposed a device that automatically recognize the speech in the form of English language and translate into Tamil language. In the system and technique, they record the speech in English language and translate into Tamil language using manual transaction, so that difficulties can be avoided during communication.

In [2] they had proposed a method for translating spoken sentences from one language into spoken sentences in another language using Spectrogram pairs. This model is trained completely from scratch using spectrogram pairs, so that it translates unseen sentences. A pyramidal-bidirectional recurrent network is combined with a convolutional network to output sentence-level spectrograms in the target language.

In [3], the proposed system is designed in such a way that it helps the patients speaking English language to communicate and describe their symptoms to Korean doctors or nurses. It is a one-way translation system. Speech recognition, English-Korean translation, and Korean speech generation are the three main parts of the system.

In [4], they have presented a deep attention model (Deep ATT) for NMT systems. For high level translation tasks, Deep ATT allows low-level attention information to decide on what should be passed or suppressed from the encoder layer. The model is designed to perform translation tasks on both NIST Chinese-English, WMT14 English-German and English-French. The model is easy to implement and flexible to train and shows effectiveness in improving both translation and alignment quality.

2. PROPOSED SYSTEM

A. System Overview:

In this project, we have created this application which can be used as an alternative for the traditional translating method i.e. a human translator which is expensive in a day to day life. Here we take one language as an audio input from Speech Recognition Device. Then input audio will get converted to text format. This input text is then passed to our neural network model.

Once the entire input text is translated from the source language to the target language (the language in which the sentence needs to be translated), the translated sentence will the shown in a text format. The next and the final step is to convert the sentence from the text format into the audio format. The converted audio format in the target language is presented as output.

Block diagram

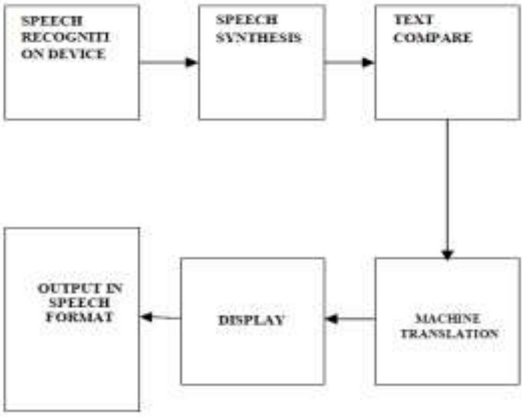

Fig. 2.1 Block diagram

B. Proposed Model:

The overall structure of our Neural Network model consists of an Encoder and Decoder both of these are Recurrent Neural Networks and they are connected to Thought Vector as shown in fig. 2.2. The Encoder network here outputs a Thought Vector which is an array of float point numbers roughly between -1 and 1 which summarizes the contents or the meaning or the intentions of the input text. We then use this Thought Vector as the initial state of the recurrent units in the decoder part of the network and then we start by inputting a marker which have taken "ssss because it does exist in vocabulary for the dataset and given this initial state which is the thought vector summarizing the input text and the start marker we want the decoder to output the word 'once' as shown in fig. 2.2 and in the next time stamp we then include the same initial state, the thought vector but we have we both the start marker and the word 'once', then we want the network to produce the word 'upon' and we input the start marker 'once upon' which output the word 'a' .We input all the remaining words in a similar way and in the end we input the start marker and the whole input text and get the end marker "eeee" as the output. End marker also does not exist in vocabulary for the dataset. This is the basic overview of our Neural Network model. Now we will study on how each layer works in the encoder and decoder of the Neural network.

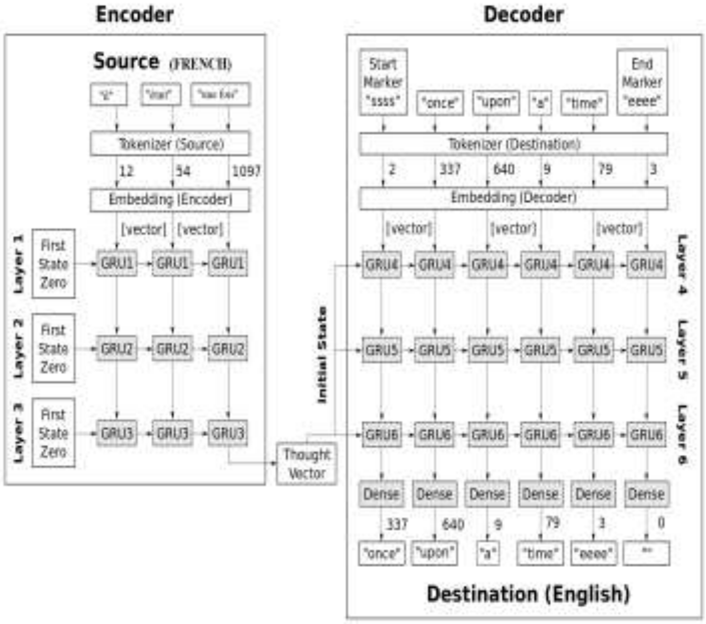

Fig. 2.2 Encoder Decoder architecture

The Recurrent units cannot work directly on text data so there is a two-step process to convert it to numbers for the Neural Network to work on. The first step is to use a Tokenizer. A Tokenizer basically turns words into integers, we need a Tokenizer for source language and we also need a Tokenizer for the destination language because they have different vocabularies. So, if we take the words in the input text, it gets converted into their integer token using Tokenization as shown in the figure. The tokens are enumerated according to their frequency in the dataset that we are using. But the Neural Network can still not work on integers, so we have to convert each of these integers into a vector a float point numbers roughly between -1 and1 so that the Neural Network can work on it.

The Embedding layer is also different for the encoder and the decoder because of the work on different languages and it learns semantic similarities between layers. The Embedding layer will learn that certain words have possibility to occur in same situations so they probably have the same semantic meaning. Another thing to note is that the Tokenizer is applied outside the Neural Network because there is no need to do this every-time we run data through the Neural Network, so we just process the data once. So the Neural Network actually consists of the Embedding layer and then 3 layers of gated Recurrent units and we use these instead of LSTM because we want to set the initial state for the Recurrent unit in the decoder and the LSTM has to internal states and this force to take out the last internal state for the last Recurrent unit here and use that as the thought vector and the initial state at the decoder. When we use the gated recurrent units it's a lot easier we can either use the internal state or the output in either case we get a vector out which has size of the internal state of the gated recurrent unit and because it has only on internal unit we need only one vector to initialize it. Now what we get out of the final layer of Recurrent units is a sequence which only outputs one vector and not a sequence of vectors because we only need one thought vector to summarize the contents of the input text. But now we want to generate a sequence of words ultimately so if the internal state of the gated recurrent units have 256 elements and we have a vocabulary and destination language of 10,000 words then we somehow need to convert this vector of 256 elements to a number between 1 and 10,000. One way of doing that is by using encoded arrays. So, we want to output a vector of 10,000 elements and we take the index of the highest element and this is the integer that we want to output. The dense layers of Recurrent unit maps from a vector that is 256 elements long to a vector that is 10,000 elements long so that we can take the maximum and can take the integer out and then we use the Tokenizer again to convert that to the original word.

once the entire input text is translated from the source language to the target language (the language in which the sentence needs to be translated), the translated sentence will the shown in a text format. The next and the final step is to convert the sentence from the text format into the audio format. The converted audio format in the target language is presented as output.

C. Flowchart:

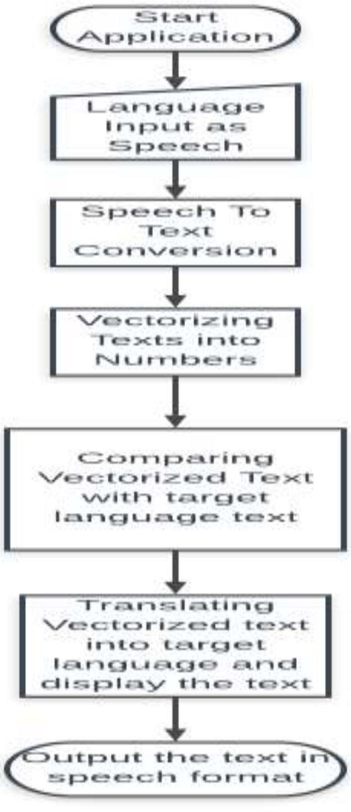
Start Application
Language Input as Speech
Speech To Text Conversion
Vectorizing Texts into Numbers
Comparing Vectorized Text with target language text
Translating Vectorized text into target language and display the text
Output the text in speech format

Fig. 2.3 Workflow of the project

3. CONCLUSIONS AND FUTURE WORK

This paper is showing that neural networks can be very powerful in translating text from one language to another. Using a simplified model, a set of words can be recognized easily and can be translated to other language. Thus, the proposed model helps to avoid the language barrier between people and helps them to communicate and interact with each other easily.

This model could be integrated with a function of translating the text written hoardings of shop from other languages in English. This can be done by first clicking the picture of hoarding, then extracting the text on the image and then translating it into English language. We can add more languages for translation. This would mainly help the solo travelers while travelling.

REFERENCES

[1] J.Poornakala , A.Maheshwari, “Automatic Speech-Speech Translation System from English to Tamil Language”, International Journal of Innovative Research in Science, Engineering and Technology, Vol. 5, Issue 3, March 2016.

[2] Michelle Guo, Albert Haque, “End-to-End Spoken Language Translation”.

[3] Sangmi Shin, Eric T. Matson, “Speech-to-Speech Translation Humanoid Robot in Doctor’s Office”.

[4] Biao Zhang, Deyi Xiong and Jinsong Su, “Neural Machine Translation with Deep Attention”, Journal of latex class files, Vol. 14, NO. 8, August 2015.

[5] Wouter Gevaert, Georgi Tsenov, Valeri Mladenov, Senior member IEEE, “Neural Machine Translation with Deep Attention”, Journal of automatic control, University of Belgrade, Vol. 20:1-7, 2010

[6] Shi Fan, Yili Yu, “Evaluating GRU and LSTM with Regularization on Translating Different Language Pairs”.