International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 03 | Mar 2020 www.irjet.net p-ISSN: 2395-0072

Storytelling App for Children with Hearing Impairment Using Natural Language Processing (NLP)

Palavi Bhole[1], Samiksha Kale[2], Ajaykumar Gujja[3], Prof. Deepti Vijay Chandran [4]

[1,2,3]Student, Smt. Indira Gandhi College of Engineering, Navi Mumbai, Maharashtra, India
[4]Professor, Smt. Indira Gandhi College of Engineering, Navi Mumbai, Maharashtra, India

---

Abstract - *During childhood, children's teachers, parents or grandparents read them a lot of fantastic stories. They did that for the time we couldn't read. Unfortunately, not everyone is blessed with an ability to hear. The children with hearing impairments might not have had a chance to know such stories at least in their childhood. This project is based on an app which narrates children's stories to hearing impaired children by taking in stories in form of text as input and giving images of sign language gestures and speech as output. In this paper, a platform Kahani which translates written English into Indian Sign Language is presented.*

Key Words: **Hearing impairment, Natural language processing, Application, Storytelling, Tokenization, HashMap.**

1. INTRODUCTION

Multiple computational works dealing with the translation of sign languages from and into their spoken languages have been developed in the last years. Some of the current research focuses on sign language recognition, some in translating text or speech into a sign language. Some works aim at recognizing words, others only letters. The Microsoft Asia group system and the Virtual Sign Translator perform the two-sided translation. Huawei StorySign app is a story telling app for hearing impaired children but it does not provide English to Indian Sign Language (ISL) translation.

Unfortunately, sign languages are not same universally or they are a mere mimic of its country's spoken counterpart. For instance, British Sign Language is not related with the Indian one. Therefore, none or little resources can be re-used when one moves from one (sign) language to another. However, to the best of our knowledge, none of these works explored how current Natural Language Processing (NLP) tasks can be applied to help the translation process of written English into ISL, which is one of the focuses of this paper [1].

2. Problem statement

There are millions of children who are hearing impaired or require sign language for communication. These children have to go through a lot of trouble during education. Stories are really fun to little children as that is the age when everything seems fascinating. This project is based on an app which will narrate children's stories to deaf and dumb children by taking in stories in form of text as input and giving images of sign language gestures as output.

3. Analysis

The proposed is based on ISL (Indian Sign Language) dictionaries from different sources, such as the Indian Sign Language Research and Training Centre (ISLRTC) initiative, and GIFs of hand gestures presented in an ISL dictionary launched by ISLRTC.

3.1 Objectives

1. To build an app which narrates children's stories to hearing impaired children.
2. To take in stories in form of English text as input.
3. To giving images of sign language gestures as output.

3.2 Scope

1. The android application will be based on java.
2. The concept of NLP is used.
3. Target audience is not limited to children with hearing impairment.

3.3 Challenges

1. The speech and the images of gesture should be in sync.
2. The system should be able to handle large database.
3. The system should be able to translate the words, not having gesture, letter-by-letter.

3.4 Natural Language Processing

Natural language process (NLP) may be a subfield of linguistics, engineering science, data engineering, and computing involved with the interactions between computers and human (natural) languages, specially the way to program computers to method and analyze massive amounts of linguistic communication knowledge. Challenges in linguistic communication process often involve speech recognition, linguistic communication understanding, and linguistic communication generation. Tokenization is the process by which a string of text is divided into smaller parts called tokens. Paragraph is a token of large text, sentence is a token of paragraph, and word is a token of sentence. Tokenization also removes all the punctuations from the text. For instance, Input: My name is Palavi. Output: ['My', 'name', 'is', 'Palavi']. Removal of stop words: Text may

contain stop words like 'the', 'is', 'are'. Stop words can be filtered from the text to be processed. NLTK module contains a list of stop words.

3.5 Data Model

We will also be using a data model like Naive Bayes data model to handle our datasets. Naive Bayes may be an easy technique for constructing classifiers. There's not one algorithmic program for coaching such classifiers, however a family of algorithms supported a typical principle. We might need an optimized data mining model for accurate and fast mapping of the images to the system.

4. Design

This section discusses the design of the project. Two algorithms are used: one for text to speech conversion and other for speech to sign conversion. For text to speech conversion speech APIs are used. For text to sign conversion, tokenization algorithm will be used, which will be done using NLTK which is a toolkit used for natural language processing.

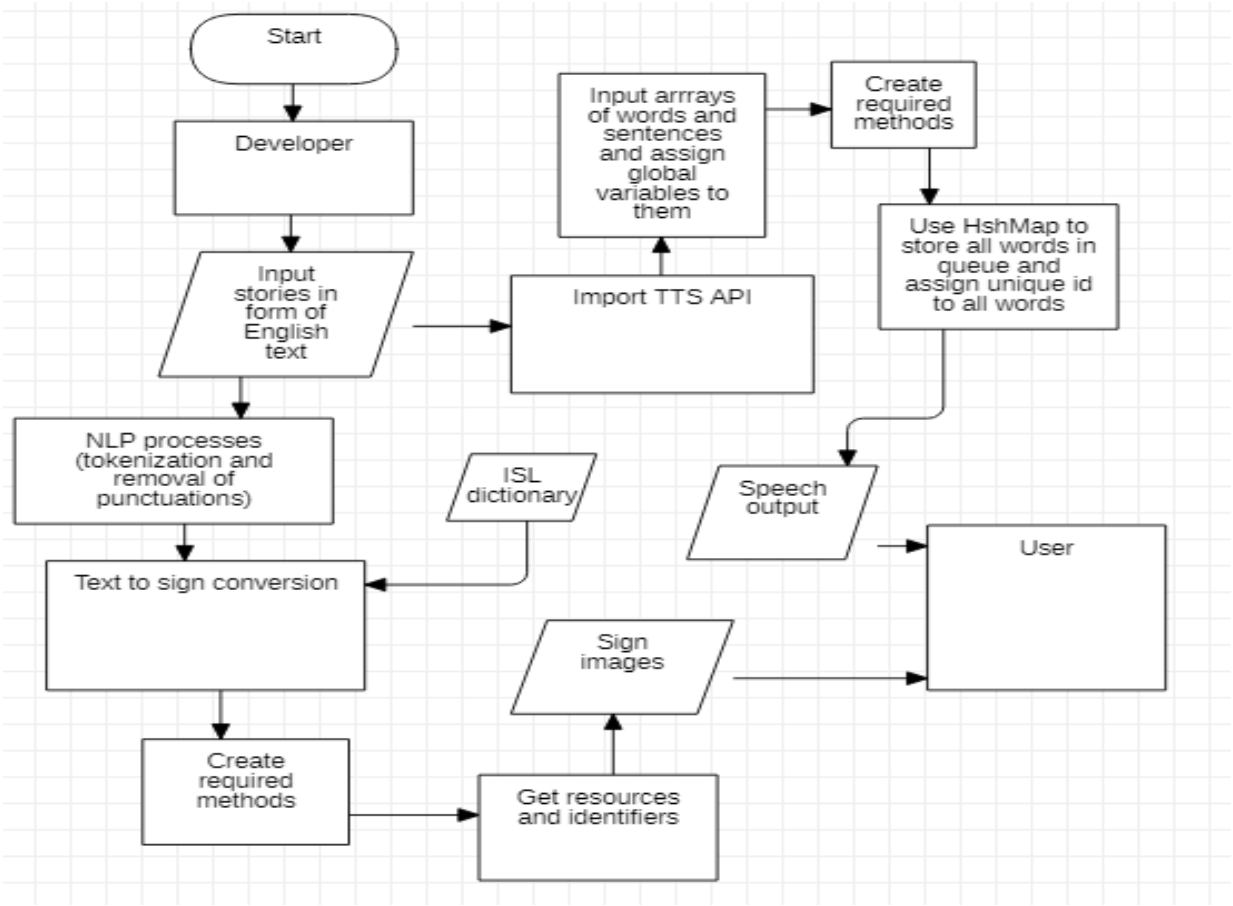

Fig -1: System Architecture

4.1 Tokenization and Punctuation Removal Algorithm

Tokenization algorithm is used to split the text into words to provide word to word translation of text into sign language. Punctuations are also removed using this algorithm.

1. Import NLTK, regular expression and word tokenization packages.
2. Input text.
3. Tokenize text.
4. Compile regular expression.
5. for words in tokens:
6. word = punct.sub("", words)
7. if len(word) > 0:
8. post_punct.append(word)
9. with open('tokenoutput.txt', 'w') as f:
10. print(post_punct, file=f)
11. end

4.2 Text to speech conversion (HashMap algorithm)

Text to speech makes an android device read the text and convert it to audio out via the speaker. Android Text to speech supports multiple languages. TTS is a simple but powerful feature. It can also be effectively used in mobile

APPs dedicated to visually impaired people [5]. Text to speech conversion can be done by using speech APIs like Google speech API or can be imported from Android Studio's inbuilt TTS function. HashMap algorithm is used for text to speech conversion.

1. Input two arrays: one for tokenized sentences and another for tokenized words.
2. Declare two global variables, i.e., i and j; where i is for array of words and j is for array of sentences.
3. Initialize i=0 and j=0.
4. Import android TextToSpeech TTS API.
5. Create Speak method with inputs string array and index.
6. If speak button press encountered, then call speak method.
7. All words get stored in HashMap function in queue format.
8. Provide each word with a unique ID.
9. Return speech output.
10. Create OnStart method with input HashMap word key.
11. Return OnStart method output: This highlights word and speaks the same word simultaneously.
12. Create OnDone method.
13. Increment i and input new i in Speak method to create a loop.
14. End.

4.3 Text to sign language conversion

Following algorithm is used for text to sign conversion:

1. Input three arrays: one for tokenized sentences, other for tokenized words and one for stop words and words with no sign.
2. Declare three global variables, i.e., i, j and k; where i is for array of words, j is for array of sentences and k is for array of stop words and words with no signs.
3. Initialize i=0, j=0 and k=0.
4. Import android TextToSpeech TTS API.
5. Create Speak method with inputs string array and index.
6. If speak button press encountered, then call speak method.
7. All words get stored in HashMap function in queue format.
8. Provide each word with a unique ID.
9. Return speech output.
10. Create OnStart method with input unique ID
11. Get resources.
12. Get identifier.
13. Match the ID with the location id from the folder which contains the signs.
14. Return OnStart method output: This gives sign output of the word according to the TTS and highlighted word.
15. End.

4.4 Alphabetization of words

Some words don't have sign languages. Such words are to be shown using sign language letter-by-letter. This done using the following:

1. Input arrays of words with no sign language.
2. Convert word string to character array.
3. End.

5. Result

Fig -2: Splash screen

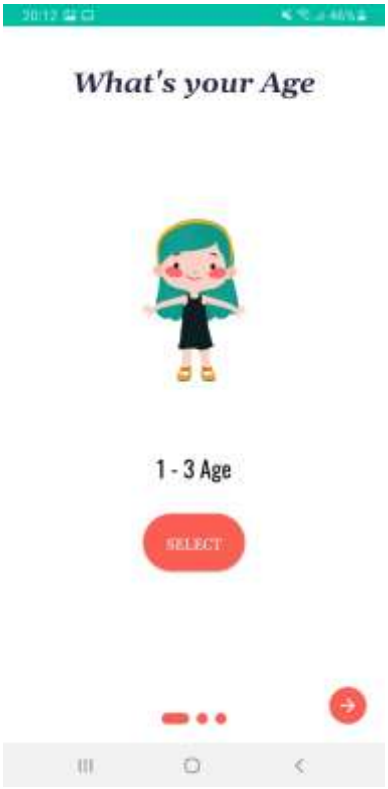

Fig -3: Age group selection page

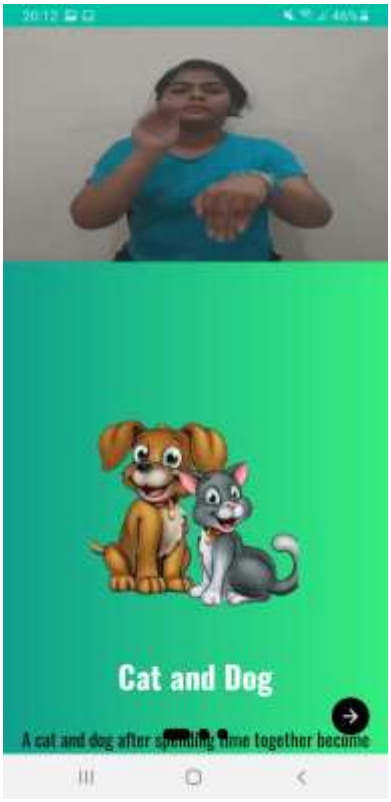

Fig -4: Story selection page

Fig -5: Read now page

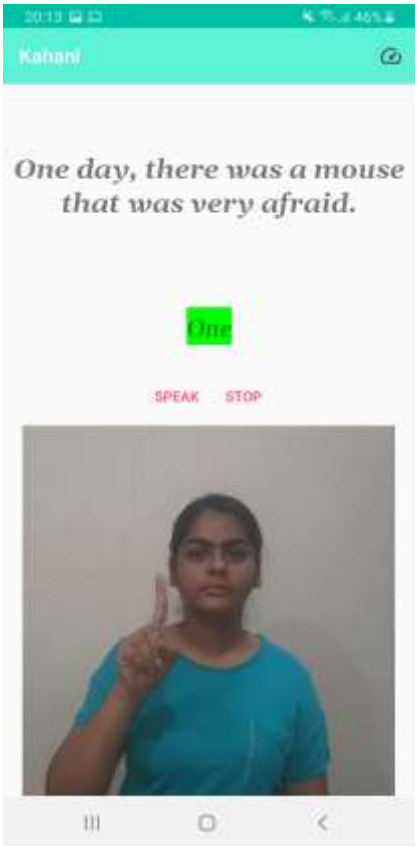

Fig -6: Story page

Fig -7: Speed meter

6. CONCLUSIONS

We have successfully implemented whatever was planned in the proposed. This is an innovative project based on an app which will narrate children's stories to hearing impaired children by taking in stories in form of text as input and giving images of Indian sign language gestures and speech as output. In this paper we present a platform, which translates written English into Indian sign language.

In future, the proposed system can be used at railway stations, bus stations, for higher education etc.

REFERENCES

[1] I. Almeida, L. Coheur and S. Candeias, "Coupling natural language processing and animation synthesis in Portuguese sign language translation," Proceedings of the 2015 Workshop on Vision and Language (VL'15), pages 94–103, Lisbon, Portugal, 18 September 2015.

[2] https://en.wikipedia.org/wiki/Natural_language_processing

[3] https://consumer.huawei.com/en/campaign/storysign/

[4] https://the-inkline.com/2017/04/21/hand-talk-groundbreaking-app-for-the-deaf-community/

[5] https://en.wikipedia.org/wiki/Spiral_model

[6] https://www.simplilearn.com/feasibility-study-article

[7] https://en.wikipedia.org/wiki/Cost%E2%80%93benefit_analysis

[8] https://en.wikipedia.org/wiki/System_requirements

[9] http://nlpprogress.com/english/dependency_parsing.html

[10] M. A. Covington, "A Fundamental Algorithm for Dependency Parsing," in Proceedings of the 39th annual ACM southeast conference, 2001, Citeseer, pp. 95–102.

[11] https://www.geeksforgeeks.org/project-idea-audio-sign-language-translator/

[12] H. Rangoonwala, V. Kaushik, P. Mohith and D. Samiappan, "TEXT TO SPEECH CONVERSION MODULE," Proceedings of the 2017 International Journal of Pure and Applied Mathematics, Volume 115 No. 6, 389

[13] https://www.hindawi.com/journals/wcmc/2018/1398191/

[14] M. Collins, "Tagging Problems, and Hidden Markov Models," (Course notes for NLP by Michael Collins, Columbia University)

[15] https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

[16] https://en.wikipedia.org/wiki/Stemming#Algorithms

[17] A. K. Melby, "Lexical Transfer: A Missing Element in Linguistics Theories," Brigham Young University Dept. of Linguistics Provo, Utah 84602 USA

[18] Cornell University "Binary Search / Dictionary Search," Lab 12 April 18–19, 2006