Machine Learning based Sign Language Interpreter

Aditi Mangal
MS in Software Engineering
San Jose State University

Heli Shah
MS in Software Engineering
San Jose State University

Mohmmadsalman Mal
MS in Software Engineering
San Jose State University

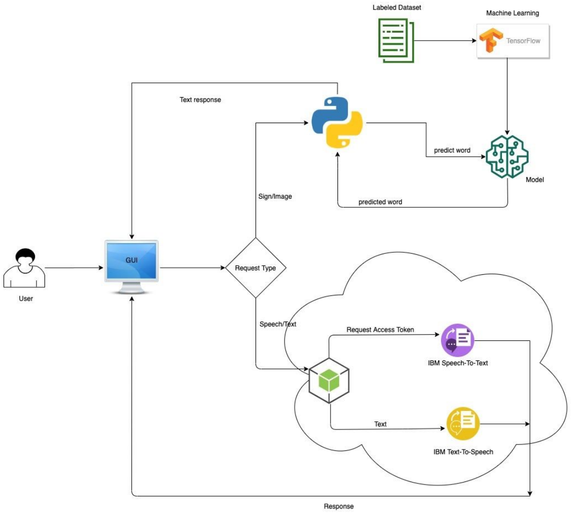
Sourabh Garg
MS in Software Engineering
San Jose State University

***Abstract*— Normal people need to get special training in sign language to communicate. This application can help people to communicate with them without having prior knowledge of sign language. It will be helpful to both groups of people. This application will translate signs into text and speech to help people who cannot speak to communicate with other people. In addition to that, this application will also support speech to text conversion for people who don't have ability to hear.**

Keywords*— Tensorflow, Keras, OpenCV, IBM Watson Speech to Text, IBM Watson Text to Speech, React, Redux, Python, Flask, Google Firebase Authentication, Sign Interpreter, Sign to Speech, Machine Learning, Gesture Recognition, Express.

I. INTRODUCTION

II. PROJECT FOCUS AND PROCESS FLOW



III. TECHNOLOGY STACK IMPLEMENTED

A. OpenCV - Python

In this project, we have utilised OpenCV(Open source computer vision) library to collect images for labeled dataset and to manipulate images coming from the user to make it ready for prediction.

B. Tensorflow - Keras - Python

Keras is a high level neural network api. In our project it is running on top of Tensorflow. We have used Keras's sequential model and different layers on top to train according to our labeled dataset.

C. Flask - Python

Flask is a micro web framework written in python. We have used flask to create an api endpoint on which the user sends requests with image payload to predict words using earlier trained model.

D. IBM Watson

Watson Text to Speech is a cloud service offered by IBM. We use that to convert interpreted sentence(s)/word to speech. Watson Speech to Text listens to a microphone stream using websocket and gives text response.

E. React - Redux

React is a javascript library to develop highly modular and interactive graphical user interfaces. Redux is an application state management library. We have utilised both to create applications user interface.

F. Google Firebase - Authentication

To prevent accessing services without signing up, we have implemented google's firebase authentication.

IV. APPLICATIONS

A. Attending any seminar or lecture, live Speech to Text for people with hearing disability.

B. At POS stations for order, Sign to speech for employees to understand customers and speech to text for customers with hearing disability.

C. At Enquiry counters, Sign to speech for employee and speech to text for customer with hearing disability.

D. At Ticket counters, Sign to speech for employee and speech to text for customer with hearing disability.

E. At the Bank teller desk, Sign to speech for employee and speech to text for customer with hearing disability.

V. FUTURE ENHANCEMENTS

Right now, the Sign to Speech interpreter works on user input like clicking for capturing signs and clicking on the speak button to convert the result to speech. In future, we would want to make it more user friendly by monitoring changes in signs to capture and converting to speech on stop. Predict more accurately not depending more on lighting conditions. Create a mobile application, where users can store regularly used sentences's audio(Text to Speech) like coffee order for drive through, bus ticket, etc.

VI. CONCLUSION

This prototype and potential applications shows how this can be helpful to many industries and people by opening doors for clear communication with specially abled people.

VII. ACKNOWLEDGEMENT

We would like to thank Professor Rakesh Ranjan(Innovation Leader, Director, Emerging Technologies at IBM) and IBM to provide us free access to IBM cloud services including and not limited to IBM Watson Speech to Text, IBM Watson Text to Speech, Watson Studio and Cloud Foundry services. And Thank Professor for introducing these emerging technologies and giving us a chance to implement/ integrate to make something innovative to help our society.

VIII. GIT REPOSITORY

A. Frontend : https://github.com/SJSUSpring2020-CMPE272/Sign-To-Speech

B. Backend : https://github.com/SJSUSpring2020-CMPE272/Sign-To-Speech/tree/master/backend

IX. REFERENCES

A. OpenCV : https://www.pyimagesearch.com/2018/07/19/opencv-tutorial-a-guide-to-learn-opencv/

B. Keras : https://keras.io/preprocessing/image/

C. IBM Watson Speech to Text : https://speech-to-text-demo.ng.bluemix.net/

D. IBM Watson Text to Speech : https://text-to-speech-demo.ng.bluemix.net/

E. Google Firebase Authentication : https://firebase.google.com/docs/auth/web/phone-auth