|  |
| --- |
| **GRAMATICALLY CORRECT SPEECH RECOGNITION** |

|  |
| --- |
| Ashish Chamanthula N969N592  Sruthibindu Vijjigiri E679Y429  Yuva Kishore Gurujala P793X363 |

**Motivation**

Voice is an important source of communication between people. There are around 7,000 languages in the world and many more dialects. There will be a problem if people with two or more different languages encounter each other and more than 20% of the world population suffer from various disabilities. Developing a Speech recognition system (speech-to-text) which allows computers to translate voice request and dictation into text. Speech recognition is the process of converting an acoustic signal which is captured using a microphone to a set of words. But just Speech recognition wouldn’t suffice. To improve the communication furthermore we even need the text to be more appropriate. This will be helpful for better understanding between people interacting over online.

**Specific problem**

In this project, we are trying to study the approach for better speech recognition by using two models i.e., one model for speech to text and the other for grammatical error correction.

**Challenges**

Despite a lot of real time applications of speech recognition most of these models have a lot of errors or defects. The solution should satisfy the requirements such as working against different accents, help those who have problems with speech or hearing communicate better and better understanding between people interacting over online.

**Literature Review**

All the papers which we have come across approach a new method for better ASR (Automatic speech recognition). In paper [1](https://www.cambridge.org/core/journals/apsipa-transactions-on-signal-and-information-processing/article/learning-from-past-mistakes-improving-automatic-speech-recognition-output-via-noisyclean-phrase-context-modeling/0025A4B2DF4F33B90FB090A195D304ED), they model ASR as a phrase-based noisy transformation channel and propose an error correction system that can learn from the aggregate errors of all the independent modules constituting the ASR and attempt to invert those. In Paper [2](https://www.sciencedirect.com/science/article/abs/pii/S0885230805000252),peaker adaptive acoustic modeling is investigated by using a novel method for speaker normalization and a well-known vocal tract length normalization method. In paper [3](https://arxiv.org/pdf/2004.04438.pdf), they proposed a novel NLP task called ASR post-processing for readability (APR) that aims to transform the noisy ASR output into a readable text for humans and downstream tasks while maintaining the semantic meaning of the speaker. In paper [4](https://arxiv.org/pdf/2001.03041.pdf), the goal of the contestants is to develop a method that improves the result of speech recognition process based on the (erroneous) output of the ASR system and the correct human-made transcription without access to the speech recordings. In paper [5](https://ieeexplore.ieee.org/abstract/document/9257188), they proposed lightweight approach that generally improve either ASR or end-to-end ST models. They leveraged continuous representations of words, known as word embeddings, to improve ASR in cascaded systems as well as end-to-end ST models.

**Summary of previous works shortcomings**

Most of the papers we came across were approaching the problem (efficient speech recognition) with corrected text to make a better ASR. There was no study done on using a second model for better speech recognition.

**Data description**

There are two models which are going to be used, Speech recognition and Grammatical Error Correction. For speech recognition we are going to use LibriSpeech. The LibriSpeech corpus is a collection of approximately 1,000 hours of audiobooks that are a part of the LibriVox project. For GEC, we are using JFLEG dataset. JFLEG (JHU FLuency-Extended GUG) is an English grammatical error correction (GEC) corpus. It is a gold standard benchmark for developing and evaluating GEC systems with respect to fluency (extent to which a text is native sounding) as well as grammaticality. For each source document, there are four human-written corrections.

**Exploratory data analysis results**

For exploratory data analysis, initially we try to determine the data types of the datasets, later we checked for the type of data present in the datasets. The dataset of libri speech and JLFEG have been imported from hugging face library datasets. Given that the datasets which we are using has already been clean cleaned and have been made readily available such that it can be used in models for implementation as such the data present in these datasets do not have missing values, outliers, and duplicates. We observed that the dataset is in Data dictionary format. For libri speech we removed the unwanted columns (speaker\_id, chapter\_id and id) and removed special character from the text of audio. For Jfleg we removed unwanted spaces and special character as these would confuse the model as noise. To increase the training data of Jfleg have taken and 248 data values from eval data.

Table

Description automatically generated

Figure Libri\_Speech Dataset

Graphical user interface, application, Teams

Description automatically generated

Figure JFLEG dataset

**Model description**

In machine learning, self-supervised learning has emerged as a paradigm to learn general data representations from unlabeled examples and to fine-tune the model on labeled data.​This has been particularly successful for natural language processing and Speech Recognition. As such we have chosen T5 for Gec and Wav2vec2 for ASR (Automatic Speech Recognition).​GEC task can be thought of as a sequence-to-sequence task where a Transformer model is trained to take an ungrammatical sentence as input and return a grammatically correct sentence.​ T5 is a text-to-text model meaning it can be trained to go from input text of one format to output text of one format. ​Wav2Vec2 learns powerful speech representations from more than 50.000 hours of unlabeled speech. Similar, to BERT's masked language modeling, the model learns contextualized speech representations by randomly masking feature vectors before passing them to a transformer network. Wav2Vec2 is fine-tuned using Connectionist Temporal Classification (CTC), which is an algorithm that is used to train neural networks for sequence-to-sequence problems and mainly in Automatic Speech Recognition and handwriting recognition.

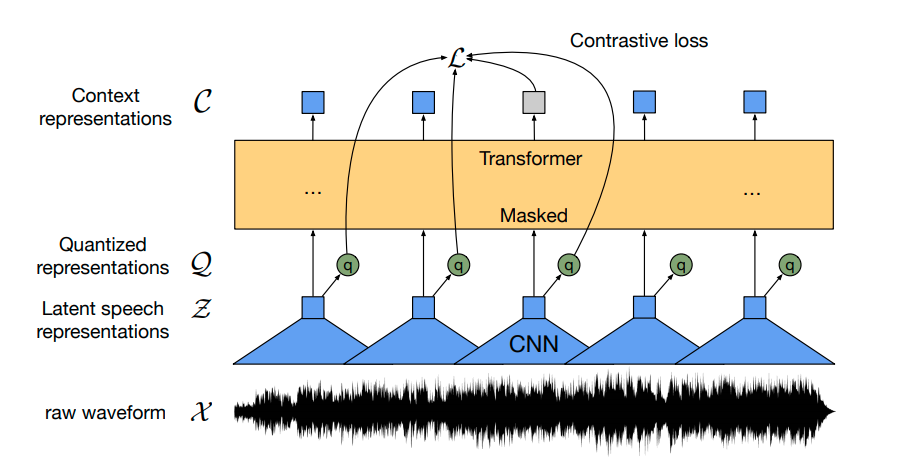


Figure Architecture of Wav2vec2 model

Diagram

Description automatically generated

Figure T5 Model

Diagram

Description automatically generated

Figure Data flow in Proposed model (predictive model design)

**Results**

The results we obtained are as following

* Wav2vec2- We trained the model with varying parameters with a small subset of librispeech (train.100 which is 100hrs of audio data) with a base wav2vec2 model. Due to shortcomings of computaions the batch was reduced to 8. After training we obtained
  + Test Wer- 0.084
  + 'train\_loss': 0.248
  + 'eval\_loss': 0.182
  + 'eval\_wer': 0.090
* T5- We trained the model with varying parameters with JFLEG dataset and to increase the traing data I added some of the validation data to train data. After training we obtained
  + Training loss-0.256
  + Loss eval(before train)-1.507
  + Loss eval(after train)-0.490

**Graphical user interface, text

Description automatically generated**

Figure Output Produced by GEC

**Results analysis/discussion**

## GEC

The observations which we get from the model GEC is that the data which we have provided for its training was insufficient and we observe the model efficiency to increase as we reduce the learning rate also the model which we used is the base model so if you had used a larger model with a larger data set we would have produced a better GEC model but this would require us to have a huge computational power to train the model.

## ASR

we observe that the results produced by the model was good which was improved by using a low learning rate. Due to the limitations of computational power, we were only able to use train.100 data set which is relatively small so I believe that if we used the larger data set, we would have produced a more better model.

**Conclusions**

We combined speech recognition and GEC to produce desired output i.e., GRAMATICALLY CORRECT SPEECH RECOGNITION.​ We belive our proposed model shows better performance than other existing models. This ​Introduces a new transfer learning method. ​

**Future work**

Just as mentioned in result analysis of our models by increasing the models’ size, dataset size we can get better result.​ The latency time of the system is also another issue which needs to be tackled.​ We believe that our work would have a lot of applications in different fields to improve communication.

**Contributions**

All the members of the team were part of Assignment’s, Coding and Documentation. Ashish and Sruthi played a key role in presentation and Documentation. Yuva was the technical head.

​

**Paper references**

* <https://www.cambridge.org/core/journals/apsipa-transactions-on-signal-and-information-processing/article/learning-from-past-mistakes-improving-automatic-speech-recognition-output-via-noisyclean-phrase-context-modeling/0025A4B2DF4F33B90FB090A195D304ED>
* <https://www.sciencedirect.com/science/article/abs/pii/S0885230805000252>
* <https://arxiv.org/pdf/2004.04438.pdf>
* <https://arxiv.org/pdf/2001.03041.pdf>
* <https://ieeexplore.ieee.org/abstract/document/9257188>