Training Report

The fine-tuned wav2vec2-large-960h-cv model did not successfully transcribe the audio file potentially due to mistakes I have made during fine-tuning.

One potential mistake might occur during encoding of the text label used to fine-tune the model. There seems to be a mistake as the encoded label did not align with the output from the ‘wav2vec2-large-960h-cv’ model.

Second potential mistake is the limited variation of hyperparameters and optimizer explored during fine-tuning. Due to time constraint, only 2 variations of batch size and learning rate were explored. To improve accuracy of the model, more values of the hyperparameters can be explored during hyperparameters tuning. Similarly, only AdamW optimizer was used during fine-tuning. Other types of optimizers such as SGD can be tested to determine if it leads to better model accuracy.

A third mistake is the limited amount of data used for fine-tuning. Due to resource and time constraint, the current model was fine-tuned on a very small subset of data. This can lead to a reduction in the accuracy the model can achieve. Fine-tuning on the full Common Voice dataset can lead to higher performance.

Apart from the mistakes made during fine-tuning, there are other methods that could be explored to improve model performance.

First, increasing variety in the training dataset. The pre-trained ‘wav2vec2-large-960h’ model was trained using the Librispeech dataset. The dataset is made up of recordings for 8000 English audio books from different genders [1]. The fine-tuned model is then trained using the Common Voice dataset and it contains recordings of different gender, age and accent [2]. Hence, the existing data has covered voices of different gender, age and accent. We can further increase variety by including recordings that have characteristics not present in the previous training dataset. For instance, we can include recordings of different emotions and pitches to allow the model to better perform in such cases.

Second, introducing noisy data to improve model performance. Fine-tuning model with recordings that has background noises is an example of ways to introduce noisy data. By training on such data, the model will be more robust to noise, thereby improving model performance. Alternatively, when a recording has background noises, it can be fed into a model that separates voices from the background noise before feeding the voice data into the model for transcribing. This can also ensure that the model performs well on noisy data.

Thirdly, we can also explore segmenting long audio into mutliple short audio before feed them into the model. This allows for better fine-tuning efficiency since the audio are now of shorter length and can be trained in parallel. It can also prevent overfitting as the components further away from each other in the audio might not be helpful in transcribing. However, it is important

to determine the right length to segment the audio to prevent removal of useful information from neighbouring component. Furthermore, it would not be wise to segment the audio midway through a word as it might lower the model's ability to decipher the word.

References

[1] V. Panayotov, G. Chen, D. Povey and S. Khudanpur, "LIBRISPEECH: AN ASR CORPUS BASED ON PUBLIC DOMAIN AUDIO BOOKS," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, South Brisbane, QLD, Australia, 2015.

[2] Mozilla Common Voice, "Common Voice Dataset," [Online]. Available: https://commonvoice.mozilla.org/en/datasets.