English Accent Detection: A Comparison of Audio, Spectrogram and Text Classification Method using Transformers

Candidate Numbers: **24155 31948 24692**

London School of Economics and Political Science
ST311 Group 11

Abstract

Accent provides additional speech information beyond content and speaker identity. Accurate accent classification can improve speech recognition through accent-specific models and improve speaker recognition. Understanding accents is also important for machine to understand human during interactions. Speech sounds may relate to different words across accents. This report uses three types of data input to classify eight English accents: audio arrays, spectrogram images, and text transcripts. The audio array model using wav2vec 2.0 has the best test accuracy from raw audio, closely followed by the spectrogram image model using Vision Transformer, whose misclassifications are mostly in classifying US accents. The text classification model using BERT performed relatively poorly. Text alone is insufficient for accent identification. The results demonstrate the effectiveness of transformer architectures for accent recognition when learning from audio data or spectrogram images. The developed models have applications in speech recognition, language learning assessment, and mitigating accent-based disparities in voice interface technologies. These models can improve speech systems for different speech patterns.

1 Introduction

In this project, we will use machine learning algorithms to classify the accents of English speeches. Accent classification can be helpful in multiple fields, for example, speech recognition systems and language learning. By identifying the accent of the speaker, the system can adjust its algorithms to better interpret the specific phonetic and international characteristics of that accent, improving overall speech recognition accuracy (1). Accent recognition algorithms can help to provide feedback to learners on their pronunciation compared to different native accents, helping them to refine their speaking skills and reduce their accent if that is their goal. Furthermore, A recent study presents that non-native English speakers and individuals with Southern accents experience reduced understanding rates by voice recognition systems like Amazon's Alexa and Google Home. Addressing these disparities through improved accent detection could significantly enhance the systems' global usability and accuracy for various accents. The dataset we use in the analysis has 7680 rows of data, each contains audio, audio transcripts and the label of accent. The accent labels contain eight different classes: "United States English" (0), "India and South Asia" (1), "England English" (2), "Scottish English" (3), "Irish English" (4), "Canadian English" (5), "Australian English" (6), "Filipino" (7). Severe class imbalance is an important challenge addressed in this research. The most frequent label is "United States English" with 3247 rows, and "Scottish English" being the least frequent class with only 119 rows.

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

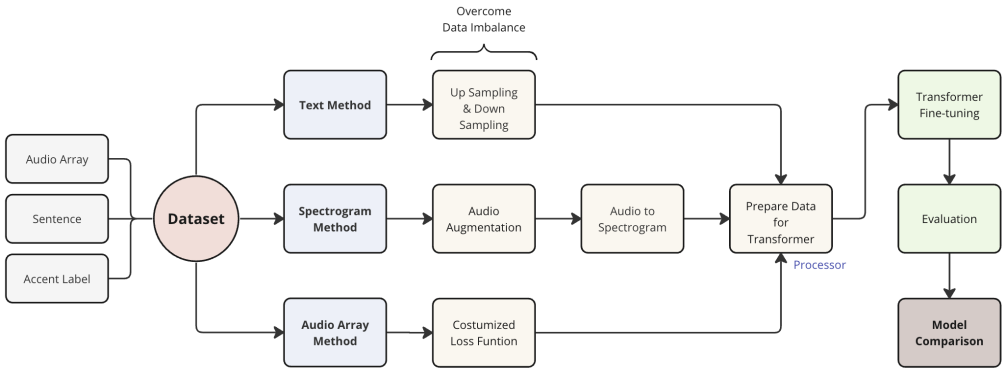


Figure 1: Algorithm Box

We aim to use three methods doing the recognition task respectively using Audio, Spectrogram, and Text Classification as in Figure 1. For Audio Classification, we will use the model (transformer) that is suitable for the input of Audio Arrays. In the spectrogram method, we will transform audio into spectrograms and use a computer vision model for classification. Lastly, we are also keen to explore whether text (transcripts of speech) can be useful in distinguishing the accent. We plan to fine-tune hugging face transformers and compare the result of three methods. Also, some insight we have discovered during the process will be displayed.

For each method, we will provide a detailed introduction about the method and the model we are fine-tuning as well as the data preparation process. We will also describe the training process and the evaluation results. Before we dive into the methods, we will firstly talk about what we have done for data preprocessing. This part also includes three methods, audio augmentation, Customized Loss Function, and down/up sampling.

2 Literature Review

Accent recognition is usually considered to be more challenging as there is a great similarity between accents of the same language (2). The initial deep learning approach on accent recognition introduced the traditional method of hybrid hidden Markov model (HMM) based on the phoneme features (3). The development of deep learning as HMM-Deep Neural Network (DNN) and end-to-end method then replaced this traditional method (4). The recent accent detection studies mostly focused on using multi-layer neural networks and Gaussian Mixture Models (GMM) in predicting the accent while doing speech recognition (5). Additionally, transformer-based end-to-end speech recognition systems were introduced for accent classification(6). Previous research also shows that the pre-trained techniques with automatic speech recognition (ASR) knowledge are found to be helpful in recognition performance (6).

Different modalities—text, spectrograms, and audio—are used in speech and accent recognition tasks. End-to-end speech recognition models have been proposed to recognise both accents and text, by adding accent IDs to the transcript (8). Furthermore, a two-stage pre-training approach has been suggested for sequence to sequence speech recognition, leveraging the rich linguistic information contained in transcripts for downstream ASR tasks (9). Spectrograms from audios are usually used with the Convolutional Neural Network (CNN) -based model for the accent classification (10). But with the occurrence of the transformer model in classification, the Vision Transformer model was introduced to analyse audio spectrograms for L1 identification (11). For audio signals-based accent recognition tasks, CNN and Recurrent Neural Networks (RNN) have been extensively applied and found their effectiveness in capturing accent-specific features (12).

From previous studies we found that most of the accent recognition tasks were built on Neural Networks and the approach of transformers is relatively new as it is less tested on this task. Moreover, the previous research only trained the models on single modality: either text, audio or spectrograms. In this research we will use transformers as the main methodology. We will also fine-tune the transformer with these three modalities and compare their classification performance.

3 Audio Array Method

3.1 Method Introduction

In this part, we use the wav2vec 2.0 framework. This model is self-supervised for speech representations (13) that distinguishes useful information from audio noises. It is pretrained on unlabeled speech data with a contrastive task, but we use labelled dataset for fine-tuning.

wav2vec 2.0 has a CNN feature encoder. The feature encoder takes raw audio as input and outputs latent speech representations in the mapping process. The encoder has seven blocks, each has a temporal convolution followed by a layer normalisation and a GELU activation function. The temporal convolutions use 512 channels with decreasing strides and kernel widths.

This encoder is then followed by a transformer network to build contextualised representations. The context network inputs the latent speech representations and outputs contextualised representations, including information from the entire sequence. 12 transformer blocks are in the context network, each has a multi-head self-attention module and a feedforward module 3,072 hidden units.

3.2 Data Pre-processing

Each accent label in the dataset is assigned a unique numeric identifier. We use a dictionary mapping to map each accent to an identifier and the other for the reverse mapping. The Wav2Vec2FeatureExtractor converted raw audio files into a structured format. It normalises the audio, as a result all input data has consistent volume and quality. It pads audio samples so that they're all the same length, allowing batch processing. After transforming the dataset format, we compute class weights from the training dataset. Unlike the data augmentation method used in the spectrogram method, this method uses weighted cross entropy loss to address the class imbalance issue. These help the model focus more on underrepresented data. In this way the model gives more attention to rarer accents like Scottish Accent during training. This is to prevent our fine-tuned model from being biased toward the more common accents, due to imbalances in the dataset especially for US accents.

3.3 Model Training

We used 10 epochs using the Adam optimiser with a learning rate of 2e-5, a batch size of 32 due to GPU choice of A100 with 40GB VRAM, and gradient accumulation over 4 batches. The learning rate was selected based on several trials and errors. The number of epochs and batch size were chosen to allow the model to converge to a good solution while fitting into available GPU memory. The validation set was evaluated every 50 steps to monitor performance relatively frequently. Early

Step	Training Loss	Valid Loss	Accuracy	F1	Precision	Recall	Mcc
50	1.8372	1.7642	0.7578	0.7463	0.7942	0.7578	0.6767
100	1.0019	0.8914	0.9733	0.9736	0.9752	0.9733	0.9652
150	0.5554	0.492	0.9954	0.9954	0.9955	0.9954	0.9939
450	0.1232	0.112	0.998	0.9981	0.9981	0.998	0.9974

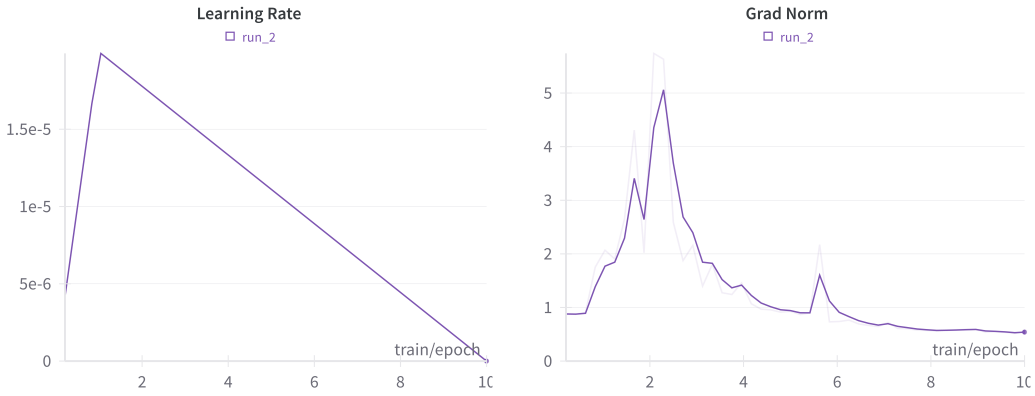
Table 1: Training Logs for Selected Epochs

stopping with a patience of 2 epochs was employed to prevent overfitting, because before training the optimum number of epochs is unknown. A weighted cross entropy loss was used. All hyperparameters were logged and performance metrics visualised using Weights & Biases.

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

The model converged quickly as seen in Table 1, reaching a validation accuracy of 75.9 after only 50 steps and a very high 97.3% accuracy after just 100 steps, converging at 99.5% by only step 150.



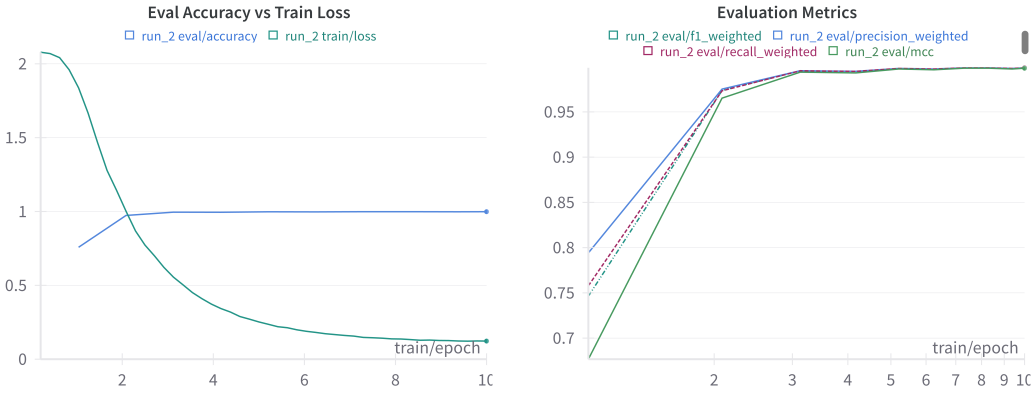
(a) Learning Rate (b) Grad Norm

Figure 2: Training Matrices

The model's learning rate firstly increased due to warmup settings and decreased linearly as seen in Figure 2 (a). The model's Grad Norm followed a similar pattern with a peak at the end of second epoch in Figure 2 (b). Further steps showed small gain in validation accuracy. Training loss decreased steadily from 1.84 down to 0.12, while validation loss decreased from 1.76 to 0.11. The exponential increasing evaluation accuracy shows that the model generalised well. No evidence for substantial overfit was found inferring from Precision and Recall of validation dataset. The F1 score (average of precision and recall) also closely followed accuracy. The model achieved good performance across all classes, not just the most common ones.

3.4 Evaluation

The final model achieved an impressive 99.8% accuracy on the test set. All performance metrics converge after the third epoch, referencing Figure 3. This indicates accurate prediction for all classes. The train loss curve moved the the opposite direction as the exponentially decreasing evaluation accuracy curve. he confusion matrix shows that only 2 data points out of 1536 were misclassified as



(a) Evaluation accuracy and Training Loss (b) Other Evaluation Metrics

Figure 3: Evaluation Matrices

in Figure 4, the fine-tuned model is an extremely accurate classifier for English accents. One English accent was incorrectly predicted as the US. Looking at the actual data point, it was the short sentence "As you wish." Possible reasons for the misclassification could be the audio is too short to provide enough distinguishing features, or that particular phrasing being more commonly associated with US English in the training data.

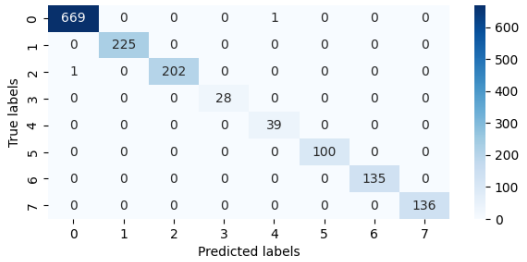


Figure 4: Confusion Matrix

One US accent was misclassified as Irish accent. The sentence was "They are on the ground." Again, the short length may not have provided sufficient information. It's also possible that pronunciations for these words similar to those of Irish. Two misclassifications are both related to US accent may be due to US English is more verbally diverse than other languages.

4 Spectrogram Method

4.1 Method Introduction

In this method, we will use the spectrogram of audios to predict the accent of an English speech. A spectrogram is a visual representation of the spectrum of frequencies of a signal as they vary with time. In the context of frequency-resolved optical gating (FROG), spectrograms are used to analyse the temporal and spectral characteristics of ultrashort laser pulses (14). Spectrograms have been widely used in audio classification, as evidenced by the increasing use of spectrogram-based models in various studies. For instance, the Audio Spectrogram Transformer (AST) has been introduced as a convolution-free, attention-based model for audio classification, achieving state-of-the-art results on various benchmarks (15) (16). Additionally, the use of spectrograms in training deep learning models, such as restricted Boltzmann machines and convolutional neural networks (CNNs), has demonstrated improved performance in audio classification tasks (17)(18).

We transform the audio data into spectrograms in image format, and then use an image classification transformer to complete the classification process. We will employ the 'librosa' package for spectrogram conversion with ViT image classification model. Hugging face API will be used for the training and evaluation process.

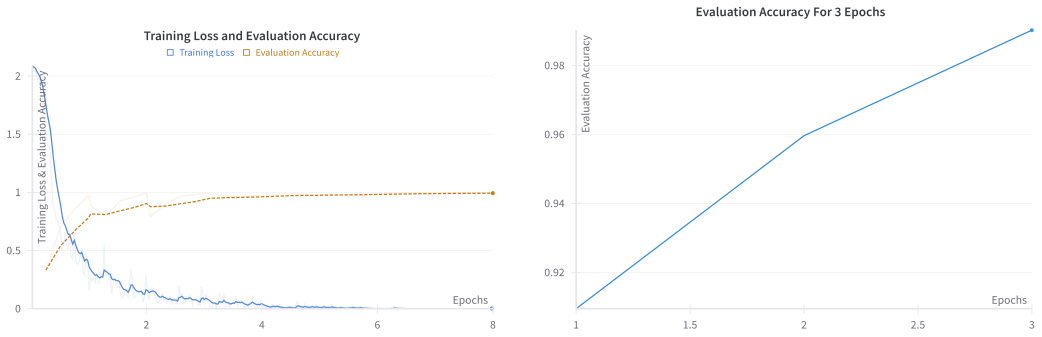
4.2 Data Pre-processing

The method we use to overcome data imbalance is doing audio augmentation. We employed the package 'librosa' to complete this task. More specifically, we achieving the audio augmentations by adding noise, changing pitch and changing speed of the audio. At the same time, we sub-sampled the major classes (the US, the UK, India and South Asia). After those steps, the dataset is generally balanced while each category has around 700-800 samples. To prepare the ready for the model. We have to first convert the augmented data into spectrogram images. We use the 'librosa' package to do that. After transforming every audio into spectrogram, we resize all the spectrograms into 244 * 244 standard image size for ViT model. Next, we normalised the image and turned the grayscale spectrogram into 3 channels that fit the model. Up to here, we have prepared all the image data and we output the image file into local devices with corresponding labels in a csv file. Then, we load all prepared images and corresponding labels to create a hugging face format dataset using the package 'datasets', both for training and testing.

Lastly, we import ViT Image Processor that aligns with the model we are going to use to get the final dataset. Now we are ready for the training phase.

4.3 Model Training

The model we use for training in this step is Google Vision Transformer (google/vit-base-patch16-224-in21k) that is downloaded from hugging face transformers. Google's Vision Transformer (ViT)



(a) Training with 8 epochs

(b) Training with 3 epochs

Figure 5: Training Matrices

is a pioneering model that applies the principles of transformers. The model is originally designed for natural language processing, but now also for image classification tasks. ViT divides an image into fixed-size patches, processes these patches sequentially using a transformer architecture, and effectively learns spatial hierarchies in image data. This approach has demonstrated remarkable performance on standard image recognition benchmarks, challenging the dominance of traditional convolutional neural networks in the field (19).

We fine-tune this transformer on all layers with training arguments of: 8 epochs, 0.0002 learning rate, batch size of 32. Two plots in Figure 5 show training loss and validation accuracy over the training process. As we can see in the plot, both training loss and evaluation accuracy curves start to converge around epoch 3. Also, at this time, the evaluation accuracy already reaches over 95%. Therefore, we chose to train the model again in 3 epochs as too many epochs might bring us the risk of overfitting.

The evaluation accuracy plot during the training process for 3 epochs is shown in Table 2, till the third epoch, the accuracy reaches 99%. We will keep this trained model for the evaluation phase. The final training metric is in the table below.

Parameter	**Value**
Epoch	3.0
Total FLOPs	1335722743 GF
Train Loss	0.0425

Table 2: Training Parameters

4.4 Evaluation

The evaluation metrics are in the table below. The evaluation accuracy on the test set is about 99.02% as in Table 3, achieving a very accurate result. Also, we can take a look at the confusion matrix of the classification as shown in figure n.

Metric	**Value**
Epoch	3.0
Evaluation Accuracy	0.9902
Evaluation Loss	0.0368

Table 3: Evaluation Metrics

As we can observe from Figure 7, the most misclassifications happen when classifying the accent is American Accent. There are 8 audios in American(0) accent that are misclassified into India and South Asia (SA, 1), England(2), Australian(6), Filipino(7), each of 2. There are also 2 Indian/SA Speeches, 2 England Speeches and 2 Filipino Speeches that are misclassified into the US Accent. There is one more England Accent that has been misclassified into Canadian (5) Accent. Those results

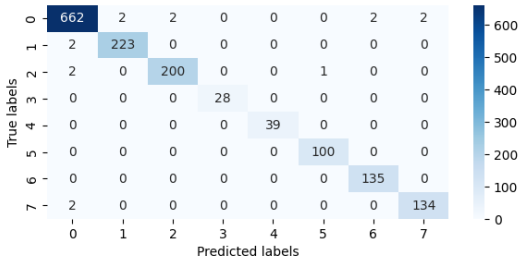


Figure 6: Confusion Matrix

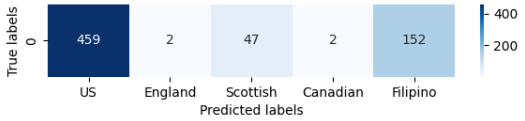


Figure 7: Confusion Matrix on extra US data

might indicate that identifying an American accent might be a more difficult task. There is a study that many English learners struggled to correctly identify the American accent, often misidentifying other accents as American, despite it being their "target" accent (20).

Based on the findings above, we did an extra evaluation specifically on a pure American accent dataset. The prediction shows a 70.9% accuracy which is much lower than the overall accuracy. The confusion matrix shows that the algorithm most likely misclassified the American Accent into Filipino Accent and then Scottish Accent.

5 Text Method

5.1 Method Introduction

The text classification of different accents is built based on the audio scripts of the dataset. We aim to classify the 8 different accents based on the features of text, and the transformer model we use in this task is BERT. BERT (Bidirectional Encoder Representations from Transformers) is a transformer model that leverages a vast corpus of unlabelled English text through self-supervised learning, meaning it requires no manually labelled data (21). The pretraining tasks of BERT enable the model to develop nuanced language representations that are useful for a variety of downstream tasks, such as feature extraction for sentence classification (21).

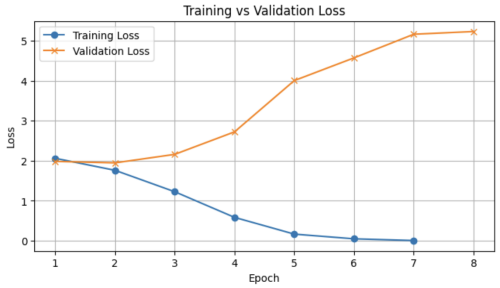
The analysis will start with the data pre-processing for text classification. Then we'll talk about the model training process based on the accent dataset, and the evaluation of the training results.

5.2 Data Pre-processing

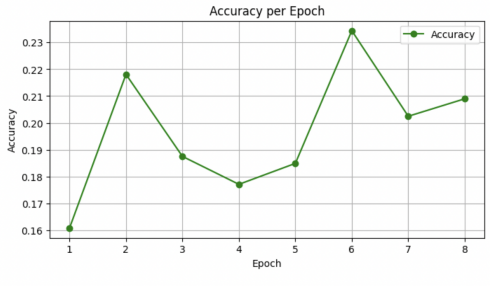
Due to the class imbalance problem of the data, we decide to upsample and downsample the text data. We divided the original dataset to training and testing data in the proportion of 8:2 and set the random state as 42 to ensure the reproducibility of the results. We then balanced the labels from the training dataset to roughly equal size. According to the class size of the training set, we decided to cut the top 3 classes to the size of 550 and upsampling the class of Irish English and Scottish English through duplication. The criteria of upsampling the Scottish English class is to triple the original size and for Irish English the standard is to double the original size, as the excessive upsampling of the minority class may lead to the problem of overfitting.

5.3 Model Training

We firstly fine-tuned the BERT model on all layers with a learning rate of 5e-5, batch size of 8 and epoch numbers of 8. The plots of training loss, validation loss and prediction accuracy by epoch below give the prediction performance of the model.



(a) Training Loss and Validation Loss

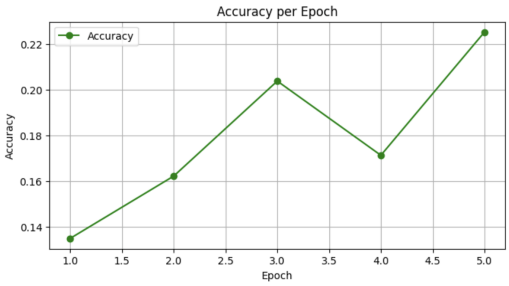


(b) Prediction Accuracy

Figure 8: Prediction Performance when epoch number = 8

The training loss in Figure 8 is decreasing consistently across epochs, which suggests that the model is learning and improving its ability to fit the training data over time. However, The validation loss continued increasing across epochs which is a clear sign of overfitting. The overall accuracy of prediction is extremely low, around 20%. The plot also shows that the accuracy rises and falls across epochs, suggesting that the model is not stable. As the training loss converges in around epoch 5, we decide to fine-tune the model again with epoch number of 5 as in Figure 9, keeping other parameters unchanged, and observe the prediction performance.

(a) Training Loss and Validation Loss



(b) Prediction Accuracy

Figure 9: Prediction Performance when epoch number = 5

After retraining, the validation loss follows the training loss closely until epoch 3, after which it begins to rise slightly. This rise is much more controlled compared to the first training run, which may indicate less overfitting. However, the prediction accuracy is still low, although this time it's more stable. This might suggest that the model or the audio scripts it is learning are not entirely suitable for the audio classification.

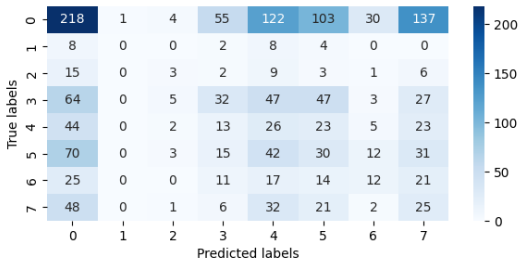


Figure 10: Confusion Matrix

5.4 Evaluation

Above is the confusion matrix for the re-trained model results, with the vertical labels as true labels and horizontal labels as predicted labels. The matrix indicates that the model classified most of the data to the class of US English(0). While the US English took the majority in the testing data set, only 32.5% of the input data is correctly classified. The classification results are also weak. For example in Figure 10, the amount of India(3) accents data that the model is classified as England(4) or Filipino(5) is even more than the amount that is correctly identified (47 and 47 over 32). These all indicated that the text classifier cannot identify the accent of the expressions contained in the audio.

6 Comparison

We used three methods for accent classification. All three methods used transformers. The audio method using wav2vec 2.0 had the best performance. The accuracy was 99.8%on the test dataset. It only made 2 mistakes out of 1536 examples. It used a CNN feature encoder and a 12-layer transformer to learn from raw audio. Imbalanced classes issue was mitigated by using a custom weight loss function. The spectrogram image method using ViT also did well with 99.02% accuracy. The method converted the audio to spectrogram images to fine-tuned the ViT model. Most mistakes involved the US accent. On the other hand, the text method fine-tuned on BERT had only around 20% accuracy, its training logs were also unstable and fluctuating. This method balanced the classes by upsampling and downsampling. The model's predictions were biased for US accent class. We can conclude reliable accent classification for accent is unachievable.

7 Limitation

An overfitting problem may exist in our fine-tuned models. The original full dataset has 818 parquet files, totaling more than 300 gigabytes of labeled voice data. Our dataset only concatenated the first seven parquet files, which is a small portion of the full dataset. The data in our subset of the dataset may not be entirely representative of the full dataset, resulting in overfitting on our specific dataset. Voices in each class in our subsetted dataset may be more similar to each other than voices in the full dataset due to grouping, or we may be very unlucky. In addition, our merged dataset was split into a train dataset and a validation dataset. As we discovered later, when a further out-of-sample parquet file was tested for the image data, the accuracy decreased to 71%, showing non-negligible out-of-sample performance degradation. More samples should be evaluated as further test data with all three models in the future.

Only eight English accents were in our dataset. These accents are regarded as the most common English accent in the world, but they do not represent the English population as a whole. Many more English accents exit, such as New Zealand English, South African English, Caribbean English, and many more regional dialects. We do not know the model's potential performance on theses accents. If our models struggle to accurately classify accents outside of the subsetted training set, the model may introduce bias against some marginalised groups. Adding some less common English accents to the exising dataset helps to mitigate the bias.

8 Conclusion

This report fine-tuned transformers to classify eight English accents. There were three methods regarding data input, which are audio arrays, spectrogram images, and text. Our results show that transformers can effectively learn features related to accents from audio data and spectrograms, while text data had very little accent related information. The audio model with wav2vec 2.0 had the best performance because it can capture rich, accent-sensitive information directly from the speech signal. The transformers' self-attention mechanisms can effectively model contextual information relevant to accent detection. The text-based approach using BERT didn't work well for accent classification. This suggests that accent differences might not be clear enough in text alone. It shows how important it is to use input representations that can capture the nuances of accent variation in speech. Very few previous studies used transformers for accent detect. Our study adopted transfer learning of existing models to a new domain. By recognising the special sounds of different accents, these systems can

understand and respond to the many ways people speak. In the future, we can make them even better by using both sound and picture clues.

References

[1] Jiao, Y., Tao, M., Berisha, V., & Liss, J. (2016). Accent identification by combining deep neural networks and recurrent neural networks trained on long and short term features. *Interspeech 2016*. https://doi.org/10.21437/interspeech.2016-1148

[2] Biadsy, F. (2011). Automatic dialect and accent recognition and its application to speech recognition (Order No. 3450188). Available from ProQuest Dissertations & Theses Global. (864672310). Retrieved from https://doi.org/10.7916/D8M61S68

[3] Liu Wai Kat, & Fung, P. (1999). Fast accent identification and accented speech recognition. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99* (Cat. No.99CH36258). IEEE. https://doi.org/10.1109/icassp.1999.758102

[4] Wang, D., Wang, X., & Lv, S. (2019). An overview of end-to-end automatic speech recognition. *Symmetry*, 11(8), 1018. https://doi.org/10.3390/sym11081018

[5] Deshpande, Y., Patel, S., Lendhe, M., Chavan, M., & Koshy, R. (2020). Emotion and Depression Detection from Speech. *Lecture Notes in Networks and Systems*, 257–265. https://doi.org/10.1007/978-981-15-8354-4_27

[6] Qiang, G., Wu, H., Sun, Y., & Duan, Y. (2021). An End-to-End Speech Accent Recognition Method Based on Hybrid CTC/Attention Transformer ASR. https://doi.org/10.1109/icassp39728.2021.9414082

[7] Shi, X., et al. (2021, June 1). The Accented English Speech Recognition Challenge 2020: Open Datasets, Tracks, Baselines, Results and Methods. *IEEE Xplore*. https://doi.org/10.1109/ICASSP39728.2021.9413386

[8] Li, S., et al. (2021). End-to-end multi-accent speech recognition with unsupervised accent modelling. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/icassp39728.2021.9414833

[9] Fan, Z., Zhou, S., & Xu, B. (2021). Two-stage pre-training for sequence to sequence speech recognition. *2021 International Joint Conference on Neural Networks (IJCNN)*. https://doi.org/10.1109/ijcnn52387.2021.9534170

[10] Mariia Lesnichaia, et al. (2022). Classification of Accented English Using CNN Model Trained on Amplitude Mel-Spectrograms. https://doi.org/10.21437/interspeech.2022-462

[11] Kishan Pipariya, et al. (2023). End-to-End Native Language Identification Using a Modified Vision Transformer(ViT) from L2 English Speech. *Lecture Notes in Computer Science*, 529–538. https://doi.org/10.1007/978-3-031-48312-7_42

[12] Kamal, M. B., Khan, A. A., Khan, F. A., Shahid, M. M. A., Wechtaisong, C., Kamal, M. D., & Uthansakul, P. (2022). An innovative approach utilizing binary-view transformer for speech recognition task. *Computers, Materials & Continua*, 72(3), 5547–5562. https://doi.org/10.32604/cmc.2022.024590

[13] Baevski, A., Zhou, H., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. ArXiv:2006.11477 [cs, eess]. Available at https://doi.org/10.48550/arXiv.2006.11477

[14] Yakovlev, V. S., Gagnon, J., Karpowicz, N., & Krausz, F. (2010). Attosecond streaking enables the measurement of quantum phase. *Physical Review Letters*, 105(7). https://doi.org/10.1103/physrevlett.105.073001

[15] Gong, Y., Chung, Y., & Glass, J. (2021). AST: audio spectrogram transformer. *Interspeech 2021*. https://doi.org/10.21437/interspeech.2021-698

[16] Gong, Y., Lai, C., Chung, Y., & Glass, J. (2022). SSAST: self-supervised audio spectrogram transformer. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10), 10699-10709. https://doi.org/10.1609/aaai.v36i10.21315

[17] Jaitly, N. and Hinton, G. E. (2011). Learning a better representation of speech soundwaves using restricted boltzmann machines. *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/icassp.2011.5947700

[18] Wu, Y., Mao, H., & Zhang, Y. (2018). Audio classification using attention-augmented convolutional neural network. *Knowledge-Based Systems*, 161, 90-100. https://doi.org/10.1016/j.knosys.2018.07.033

[19] Dosovitskiy, A., et al. (2020). An image is worth 16x16 words: transformers for image recognition at scale. https://doi.org/10.48550/arxiv.2010.11929

[20] Atagi, E., & Bent, T. (2013). Auditory free classification of nonnative speech. *Journal of Phonetics*, 41(6), https://doi.org/10.1016/j.wocn.2013.09.003

[21] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018, October 11). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *ArXiv.org*. https://doi.org/10.48550/arXiv.1810.04805

Individual Contribution: 31948 - 33.3333%, 24155 - 33.3333%, 24692 - 33.3333%
31948 coded spectrogram classification, 24155 coded audio classification, and 24692 coded text classification.
Introduction was mostly the contribution of 41948, Literature Review was mostly 24692, limitations and conclusion were mostly 24155.
All candidate contributed equally in terms of word count for the report and slides.