GROUP 32 - PROJECT PROPOSAL

Aryan Hussain
Student# 1008968324
aryan.hussain.@mail.utoronto.ca

Ivan Chou
Student# 1008968967
ivan.chou@mail.utoronto.ca

Bala Kannan Murali
Student# 1009454494
balakannan.murali.utoronto.ca

Daniel Neaugu
Student# 1009306136
daniel.neagu@mail.utoronto.c

ABSTRACT

The following document is a final report concluding our team's efforts culminating in our audio classification project. Team032 has developed a convolutional recurrent neural network that is able to effectively classify different forms of media content including Music, TedTalks, Audiobooks, and Conversations based on a 10 second sample of their respective audio files. These audio samples were sourced from multiple datasets, converted into WAV files, and subsequently into MFCC spectrograms with uniform dimensions. Our final model achieved a high validation and training accuracy within the subset of training and validation data, and was able to achieve a similarly high testing accuracy given an entirely novel subset of testing data. This report summarizes and clearly outlines the progress of the group and its individual members until August 11, 2023. The purpose of this report is to present the model that Team032 created in detail, and discuss its functions, and its relevance.

—-Total Pages: 9

1 INTRODUCTION

Social media is a powerful tool that is capable of facilitating education, entertainment, personal experiences, and immediate access to any relevant information. With an increase in the popularity of social media services such as Instagram Reels, TikTok, and YouTube Shorts, users are exposed to an abundance of video and audio content in a very short amount of time. However, users are not always able to determine whether the information presented to them is accurate, misinformative, or of malicious intent.

Our project thus aims to classify audio data into categories so that users can determine the source of the content that they are consuming. These categories include TedTalks, audiobooks, conversations, and movies. Doing so presented a unique challenge to our group since our initial research involved deep learning models that were utilized to identify specific words, phrases, or speakers, however our project specifically aimed to more abstractly classify audio data into the aforementioned categories.

Our team tackled this problem using a deep learning approach, creating a model capable of organizing audio data using a Convolutional Recurrent Neural Network. Deep learning was the preferred approach for this goal due to its effectiveness at identifying the subtle relationships within large amounts of data. In contrast, manual organization would entail individuals listening to audio data one file at a time, unable to differentiate between subtle differences in the audio sound waves or spectrograms, rendering a non-machine learning method less efficient. An added benefit is that input audio data has the ability to be efficiently converted to sound waves, and subsequently spectrograms, so that deep learning models may take them as inputs.

2 ILLUSTRATION

Below is a depiction of the teams plan to classify an audio file into either a movie, audio book, podcast, ted talk, or song.

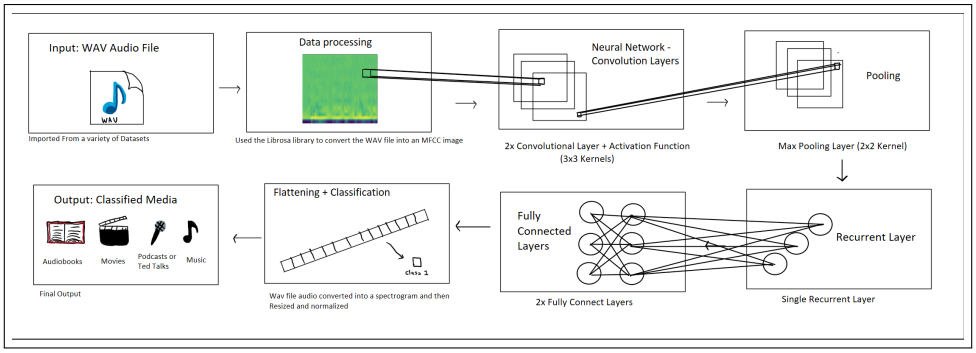

Figure 1: Project Plan Illustration

3 BACKGROUND AND RELATED WORK

Hackernoon recently published an article that explores a variety of techniques for audio classification. These methods cover a wide range of applications, including analyzing environmental sounds, classifying music genres, and utilizing AI to differentiate between different types of audio data. The article highlights how AI-based approaches are used for tasks like anomaly detection, understanding soundscapes, offering personalized music suggestions, and creating speech recognition systems. This underscores the practical applications of audio classification methods across diverse real-world scenarios.

The Papers with Code repository is a valuable platform for those working on audio classification. It compiles a comprehensive collection of research papers that delve into automatic audio signal categorization. These studies employ sophisticated machine learning models, including deep learning architectures, feature extraction techniques, and ensemble methods, to achieve highly accurate classification outcomes. This repository encourages collaborative efforts and knowledge exchange among researchers, thereby contributing significantly to the ongoing advancement of audio classification techniques.

Spotify employs machine learning algorithms to perform in-depth audio analysis for classifying music tracks. By extracting key information related to elements like rhythm, melody, and timbre, Spotify's system groups songs based on their acoustic attributes. This classification process enhances the platform's capability to offer tailored music recommendations, curate personalized playlists, and ultimately enhance user engagement.

Twitter also embraces AI and machine learning to automatically categorize diverse content types, including audio-based media like voice notes and podcasts. This technology aids users in discovering relevant content by organizing audio material according to their preferences and interests. Integrating audio recognition technology enhances the overall user experience and engagement on the platform.

Google Home makes effective use of voice recognition technology, powered by AI, to differentiate and respond to distinct user voices. This functionality enables personalized interactions, customized information delivery, playback of audio content, and control over smart home devices. The underlying AI techniques empower Google Home to accurately distinguish individual users, thus significantly enhancing the overall user experience within smart home environments.

4 Data Processing

Throughout the data collection phase, we obtained data from multiple sources, encompassing four distinct types: music, audiobook, tedtalks, and conversations. Our primary objective was to create a comprehensive and varied dataset to enhance the reliability and versatility of our models. The music dataset was sourced from the Library of Congress' website, providing a substantial collection of over 2000 audio files, spanning a wide spectrum of musical genres and styles.

For the audiobook and Ted Talk datasets, we discovered valuable resources on Openslr, an open-source platform known for its high-quality speech-related datasets. The conversations dataset, capturing authentic spoken interactions, was obtained from the reputable Meta AI platform, allowing us to delve into the intricacies and subtleties of human communication.

During the process, we encountered a challenge concerning the diverse audio file formats, including .mp3, .sph, .flac, and .mp4. To maintain consistency, we employed the FFMPEG software to convert these files into the universally recognized .wav format. Additionally, the Xrecode application was utilized to further optimize audio quality and ensure uniformity.

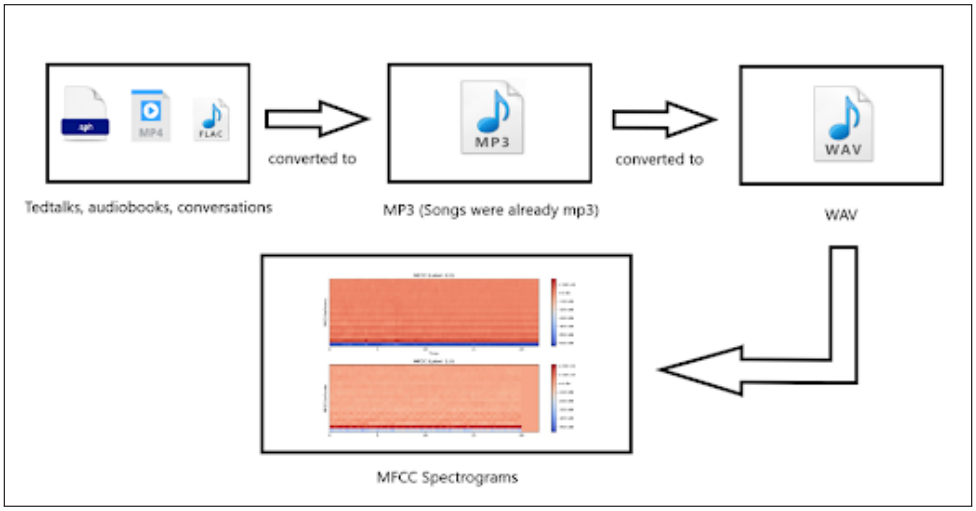

Figure 2: Audio Processing Diagram

Throughout training, we faced varying audio file lengths, which could potentially impact model performance. To mitigate this issue, we utilized the FFMPEG software once again to trim all audio files to approximately 10 seconds, ensuring uniformity and enabling efficient training. Data integrity was a top priority, and we meticulously performed data cleaning to validate that all audio files adhered to a consistent format. This crucial step ensured the accuracy and reliability of our results. During the data processing phase, we meticulously handled our diverse dataset, comprising 2592 conversations, 982 Tedtalks, 2382 music, and 2447 audiobook samples. To prepare the data for analysis, we skillfully utilized the user-friendly librosa library in Python, enabling the conversion of audio data from .wav files into spectrograms. The extraction of Mel Frequency Cepstral Coefficients (MFCC) features from these spectrograms laid the foundation for training our advanced models.

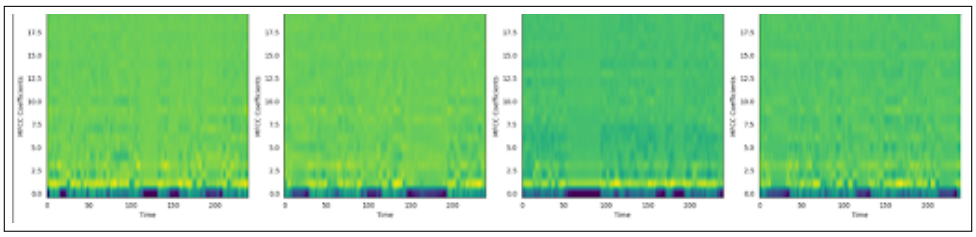

Figure 3: MFCC's from converted WAV files

This transformation allowed us to capture relevant acoustic characteristics, essential for effective audio analysis. Subsequently, these processed MFCC features were utilized in both our primary and baseline models, enabling us to develop robust and accurate systems for a range of audio-related tasks. In conclusion, our data collection process was meticulous, encompassing a diverse range of content. We proactively addressed challenges, ensuring data standardization and reliability. This carefully curated dataset and preprocessing efforts have been instrumental in driving the development of robust and accurate models for a wide array of audio-related tasks.

5 ARCHITECTURE

Audio-Cat uses a Convolutional Recurrent Neural Network (CRNN) due to its ability to recognize patterns in an image. Convolutional layers are used to detect image patterns such as colour intensity representing sound in the MFCC images while the recurrent layer is primarily used to detect more sequential data such as repeating waves which is ideal for music. Audio-Cat has 2 convolutional layers with 3x3 kernels, followed by a 2x2 max pooling layer, A recurrent layer, and 2 fully connected layers.

In terms of hyperparameters, the team meticulously different versions until the final model was reached with the highest recorded accuracy. Audio - Cat has a small learning rate of 0.0001 as increasing it caused drastic fluctuation of the accuracy. The adam optimizer was used as it has proven itself to be the most accurate for audio classification projects. The cross entropy loss function was used because the audio was being classified into one of four different categories. The team utilized 20 iterations or epochs due to it leading to a high accuracy for both training and validation without succumbing to overfitting. Finally, the team decided to use a batch size of 1 which did increase training time and lower efficiency, but also lead to the highest accuracy seen by the model.

6 BASELINE MODEL

The KNN (K-nearest neighbors) algorithm was chosen as the baseline model because of its straightforward design and widespread application in classification issues. KNN, a supervised learning method, is used to classify a new data sample based on the neighboring training instances that are available in the feature space. It does not fit utilizing any models and only relies on memory. The group that predominates near the sample to be classified is taken as the label.

When it is entered, one of the test data's k nearest neighbors is picked. There is no requirement to possess any prior understanding of the data's structure in the training set.

Predominantly, work was done to improve the primary model for Audio-cat whereas the baseline model remained untouched in terms of logic.

Below is an example diagram of how our baseline model would work with 3 categories (A,B,C):

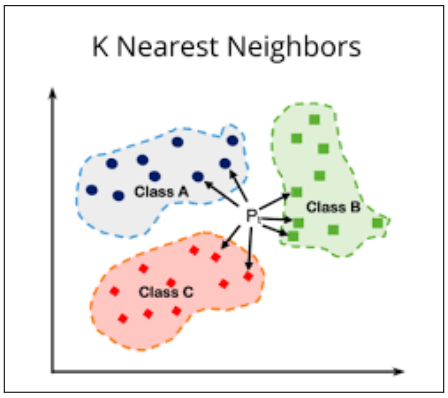

Figure 4: K-NN Classification Example

The Mfcc spectrograms were generated using the librosa module which allowed us to extract MFCC features. The features were converted to tensors and then cropped and normalized to a size of (20,240) using the torch module and then converted back into NumPy arrays. KNN Classifier function from the librosa model was used which performed the task of classification.

By arbitrarily selecting k=5 which we did not change so that the KNN makes predictions based on its 5 nearest neighbors, we were able to classify a subset of our audio sample data consisting of 800 audio samples, with 500 training samples, and 300 testing samples, with an accuracy of 79.33. The samples were taken from our main datasets which were used for our primary model with the training time considerably lower than the primary model.

Figure 5: Baseline Model Diagram

7 Quantatative Results

Our model's performance was tracked and measured by keeping track of the model's classification accuracy. Our final model performed with a validation accuracy of about 96% and a training accuracy of about 96% when trained to classify audio data into one of the four categories. This is an improvement from the baseline K-NN mode that performed with about an 80% validation accuracy, and shows similar results to the primary model which performed with a training accuracy of 97% and a validation accuracy of 95%.

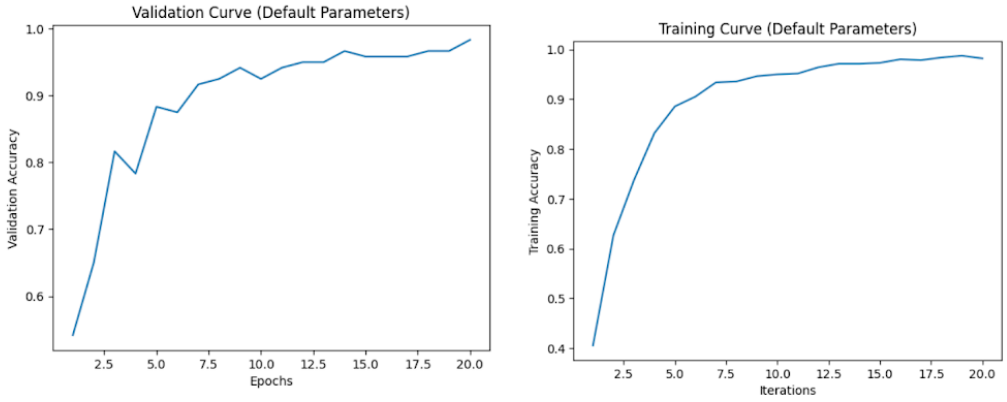

Figure 6: Quantitative Training and Validation Curves from primary model

Based upon the training and validation curves for our model above, the model's validation and training accuracies are steadily increasing, indicating that the model is able to both generalize the new data and the training data well. Since the validation curve seems to not have plateaued yet, it is also possible that given more epochs of training, that the validation accuracy of our model may further increase.

8 Qualatative Results

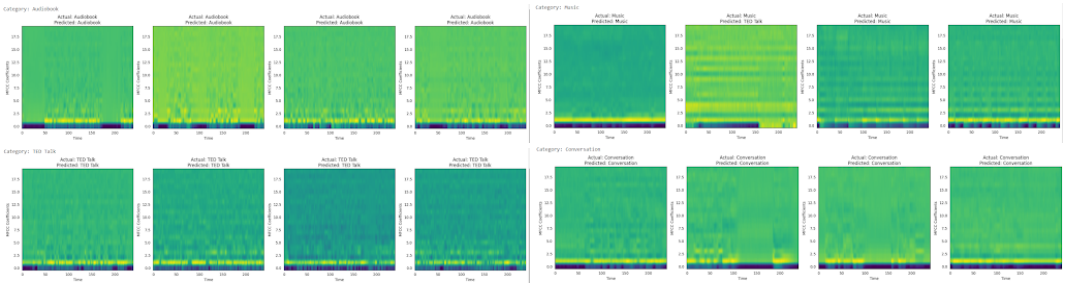

Figure 7: Here we can see some MFCC spectrograms for the 4 categories of audio samples with their assigned labels and ground truth values.l

These results reinforce the model's accuracy in classifying the inputs into their categories with similar performance observed in all 4 types.

With a validation accuracy of 96.29%, our sample predictions were correct (for the most part) for all 4 types of input audio data. The results make sense due to knowledge of our various dataset's categories and the labels being correctly matched. Furthermore the model learnt to classify rather than overfit as the training went on.

9 New Data Evaluation

We developed the model greatly after the training phase and progress report with modifications made to the hyperparameters including the learning rate, layers and epochs. The number of iterations increased from 10 to 20 with the change in learning rate to 0.0001. We also changed the layer architecture which now has 2 convolutional layers with 1 recurrent layer that utilizes 2 fully connected layers which were the major changes with the number of input and output channels along with pooling layers remaining the same. The number of parameters also changed from 123,746 since the progress report. These changes were primarily made based on the validation data so that the model can perform well with testing data that it has never been exposed to beforehand. The training took place using 2000+ samples from each categories with the accuracies being as follows:

```
Epoch 1/20: Train Accuracy: 59.62%, Val Accuracy: 65.67%
Epoch 2/20: Train Accuracy: 68.49%, Val Accuracy: 73.95%
Epoch 3/20: Train Accuracy: 77.72%, Val Accuracy: 82.48%
Epoch 4/20: Train Accuracy: 84.62%, Val Accuracy: 87.21%
Epoch 5/20: Train Accuracy: 88.79%, Val Accuracy: 87.21%
Epoch 6/20: Train Accuracy: 89.53%, Val Accuracy: 92.27%
Epoch 7/20: Train Accuracy: 91.46%, Val Accuracy: 93.37%
Epoch 8/20: Train Accuracy: 92.30%, Val Accuracy: 93.76%
Epoch 9/20: Train Accuracy: 93.15%, Val Accuracy: 93.13%
Epoch 10/20: Train Accuracy: 93.08%, Val Accuracy: 93.61%
Epoch 11/20: Train Accuracy: 93.86%, Val Accuracy: 93.53%
Epoch 12/20: Train Accuracy: 94.32%, Val Accuracy: 94.71%
Epoch 13/20: Train Accuracy: 94.49%, Val Accuracy: 94.40%
Epoch 14/20: Train Accuracy: 95.08%, Val Accuracy: 94.24%
Epoch 15/20: Train Accuracy: 95.18%, Val Accuracy: 95.19%
Epoch 16/20: Train Accuracy: 95.52%, Val Accuracy: 95.90%
Epoch 17/20: Train Accuracy: 95.96%, Val Accuracy: 95.82%
Epoch 18/20: Train Accuracy: 96.08%, Val Accuracy: 95.58%
Epoch 19/20: Train Accuracy: 96.31%, Val Accuracy: 95.03%
Epoch 20/20: Train Accuracy: 95.81%, Val Accuracy: 96.29%
Training finished!
```

Figure 8: Training Accuracies

For this reason, the testing data was only used a single time to ensure no changes could be made afterwards to remove any biases. The testing on the new data was done using audio clips randomly from never before used from multiple different datasets.

Team032 took this one step further by implementing data that the members themselves have recorded to see if the model works in real world applications. The following is just one of the classifications using these audio recordings.

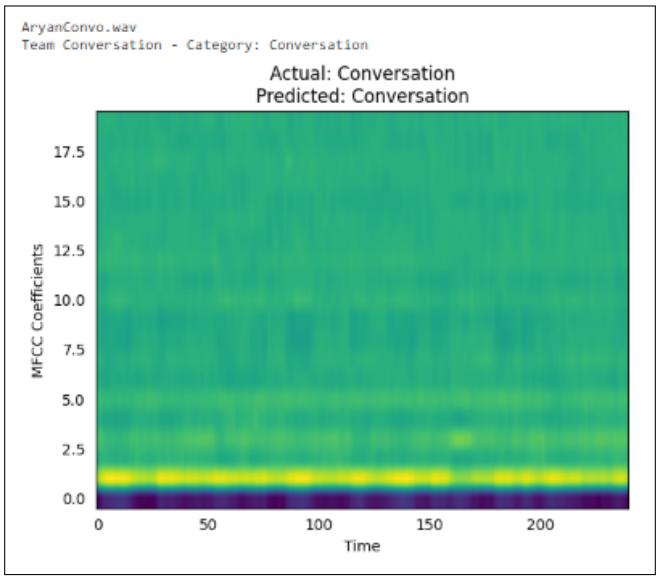

Figure 9: Prediction and Label of a conversation recording

Our model's accuracy is able to perform consistently above 90

10 DISCUSSION

10.1 MODEL PERFORMANCE SIMILARITY

A surprising result of our final model's performance was its similarity to the primary model's performance. Our final model was able to consistently classify all subsets of data including the training, validation, and testing subsets with an accuracy of about 96%. These numbers are very similar to the primary model which achieved a training accuracy of 97% and a validation accuracy of 95%. Although the model performance is quite high in both instances of the primary and the final model, there is not much difference in the validation accuracies of the models. The lack of a significant increase in performance between these models may be attributed to the sample size of each model's training and validation datasets. The primary model utilized 800 audio samples in both its training and validation, while the final model utilized 8403 audio samples. This is a large increase in the amount of data available for training and validation, and generally, an increase in the amount of data will amount to an increase in validation and training accuracy in deep learning models because the model will become more adept at generalizing new data. Although the final model was provided with a large amount of new data that the primary model was not provided, the nature of the new audio data supplied may explain why the increase in the amount of data did not significantly increase the validation accuracy of the model.

The increase in data provided to the final network may have varied and been a lot more diverse in nature, providing the model with a greater challenge in classifying the data. This is because the audio subsets that consisted of audiobook, tedtalk, conversation, and music samples were amassed from a diverse variety of audio sources. The Music category contained samples from an abundance of different artists, genres, melodies, and rhythms and the Audiobook, TedTalk and Conversation categories contained audio samples from an abundance of unique phrases, words, cadences, speakers, and intonations. These differences of subtle audio features due to the differences in the source

of each audio sample may have opposed the final model's ability to improve at generalizing new audio data as the new audio data became increasingly more diverse. Ultimately this effect may have resulted in the final model's similar performance to the primary model.

10.2 Audio Classification

Coming into this project, our team's inspiration largely originated from a curiosity into how audio classification works. Preliminary research had revealed that audio classification commonly utilized convolutional neural networks. Having just learned about how convolutional neural networks were used to classify images in our lectures, we wondered what similarities sound classification could possibly have to image classification. Further research revealed that sound classification was all along actually a specific case of image classification. To classify sound, audio files had to be converted into MFCC spectrograms whose values are stored in tensors and then passed as input into the deep learning model.

Upon manual analysis of the spectrograms generated for random audio samples and their respective labels, visible differences and patterns in the general form of the spectrographs pertaining to each category can be seen quite clearly through their colour compositions as specific patterns of different colours seemed to pertain to different audio categories. Although classifying the audio samples manually according to these differences does not often work, the simple nature of the visual differences that can be used to differentiate the spectrographs of audio samples from different categories indicated to the team that a deep learning approach could likely become very proficient at identifying both the visual patterns that humans perceive to visibly differentiate between spectrograms of different categories and other likely subtle differences that humans may not perceive but may also be just as useful to a deep learning model in classification.

When testing our model, and measuring its performance, we found that our hypothesis was correct. The model was able to very quickly generalize new audio data. The model started at a training accuracy of 59.62%, and a validation accuracy of 65.67% after the first epoch, jumping to a training accuracy of 68.49%, and validation accuracy of 73.95% after the second epoch. And by the 7th epoch of training, the training and validation accuracies had both passed accuracies of 90%. This confirmed that a deep learning approach was effective to achieving our project's goal of classifying audio data. Our model's performance thus exceeded our expectations, performing very well. Although we initially thought that a deep learning approach would be effective in processing and categorizing such large amounts of data, we did not realize until this point that the model could learn so quickly and achieve such high levels of accuracy in under ten epochs of training. By analyzing the spectrograms that were passed into the deep learning model ourselves, our group gained further understanding into how the MFCC features describe the type of audio that they represent, and why the deep learning model was able to do

11 Ethical Considerations

Table 1: Ethical Considerations

Consideration	**Why is it a problem?**	**How can we rectify it?**
Proper Consent from those involved (ie. using content).	All content and audio clips used must be obtained ethically. To be considered, the audio clip be free to use publicly to ensure no content is stolen and the parties involved are comfortable in its use	- Using existing datasets from both HuggingFace, Library of Congress, and Meta. All three provide public datasets that were ethically sourced. - All online resources, databases, and information used will be given credit by the team

Representation Bias	Having Bias towards certain collections or groups of data may misrepresent or underrepresented other groups and so diverse and balanced dataset should be used	The Datasets used have a diverse set of voices from all genders and races to ensure the model is not biased towards or against any group
Aggregation Bias	Aggregation bias occurs when a single model is applied to different groups assuming they have similar distributions, but in reality, they have distinct patterns. This can result in inaccurate or suboptimal outcomes for specific groups.	The model will not be marketed as a one model fits all but will specifically be used to classify between forms of media with English dialect regardless of speech pattern or accents
Evaluation Bias	Evaluation bias occurs when the model is tested on a non-diverse dataset but only on a very niche subset of the category that the model has been trained with. This may lead to high yet misleading performance metrics.	The model will be evaluated with a large quantity of never before seen test data from each of the datasets. The team will also test the model against audio samples recorded by the team.accents

12 PROJECT DIFFICULTY/ QUALITY

At the core of this project lies a multifaceted challenge resulting from the difficulties associated with sourcing the audio files, making them compatible with the model, and developing a model complex enough to classify them into one of the four categories.

The training, validation, and testing files were sourced from multiple Obscure datasets on the internet. Each dataset contained thousands of audio files with several audio formats which all had to be converted to WAV files, categorized, and trimmed to reach a consistent and worthy final dataset.

Once the data was mounted to the team's google drive, the team then had to convert the WAV files into MFCC spectrograms to be compatible with the CRNN. The difficulty in this lies within the limited documentation for the Libroasa library which made converting all the thousands of WAV files into MFCC's challenging. MFCC's also act a lot different to 3 channel images that convolutional neural networks typically use resulting in the team having to test a variety of transformation techniques.

Once the data was finally in a format the team could use, the real challenge began when trying to create a model to classify the data. MFCCs are hard to classify because of the complex and subtle variations they capture within audio signals that not even humans can classify properly like they can with regular images. It became painfully apparent to us that a simple CNN would not yield a high enough accuracy that the team could accept so the model was scaled up to also include a recurrent layer. Despite these difficulties, Team032 successfully created a model that can classify between forms of audio media using just the audio wave itself with a much higher accuracy then expected for this type of project.

REFERENCES

Aid, . "Speech recognition overview: Main approaches, Tools Techniques." (2023).

Ajao, Esther. "Spotify personalizes audio experiences with Machine Learning: TechTarget." (2022). https://www.techtarget.com/searchenterpriseai/feature/Spotify-personalizes-audio-experiences-with-machine-learning

Hugging Face, "The AI community building the future" https://huggingface.co/blog/audio-datasetsa-tour-of-audio-datasets-on-the-hub

Javatpoint, "Nearest Neighbor Alogrithm For Machine Learning" (2021) https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

Lim, Hengtee. "An introduction to 4 types of audio classification." (2020) https://hackernoon.com/an-introduction-to-4-types-of-audio-classification-nhg3zq3/

Nair, Pratheeksha. "The dummy's guide to MFCC." (2018). https://medium.com/prathena/the-dummys-guide-to-mfcc-aceab2450fd

Nandi, Papia. "Recurrent neural nets for audio classification." (2021). https://towardsdatascience.com/recurrent-neural-nets-for-audio-classification-81cb62327990: :text=This

Papeloto, . "Urban sound - feature extraction knn." (2019). https://www.kaggle.com/code/papeloto/urban-sound-feature-extraction-knn

Papers with Code, "Papers with Code" https://paperswithcode.com/task/audio-classification

Science 2019 " Deep learning for speech recognition, Medium (2019) https://odsc.medium.com/deep-learning-for-speech-recognition-cbbebab15f0d

Spotify Engineering, . "For your ears only: Personalizing spotify home with Machine Learning." (2022). https://engineering.atspotify.com/2020/01/for-your-ears-only-personalizing-spotify-home-with-machine-learning/

Writer, The Experimental. "Audio classification using CNN-coding example." (2019). https://medium.com/x8-the-ai-community/audio-classification-using-cnn-coding-example-f9cbd272269e