AUDIO PROCESSING AND INDEXING FINAL PROJECT REPORT SPEECH EMOTION ANALYSIS AND PLAYLIST GENERATION

Xiang He
s3627136
@vuw.leidenuniv.nl

Dervis Onur Gurbuz
s3440524
@vuw.leidenuniv.nl

Xudong Shi
s3444538
@vuw.leidenuniv.nl

December 19, 2023

1 Introduction

The recommendation system has been a sophisticated business model with proven revenue-generating capabilities. Within vast range of application of recommendation system, music recommendation has become particularly prominent in the music applications. In the popular music applications, such as Spotify and YouTube Music, beside the user's personal preference towards a certain a singer or song that could be accessed through search function, the application would also recommend music based on a range of properties that potentially matches the user's preference. As this has greatly enhanced user experience by enriched the user's media library, and constructed a channel for music creators, thereby greatly diversified the music market.

Numerous elements have been taken into account when recommending music, including user profiling factors like age and region [1]. Despite this, from our experience, the use of speech emotion in music recommendation has not been widely implemented. Speech emotions can encapsulate a range of emotions reflecting the speaker's mood, potentially serving as a valuable criterion in music suggestion. When speaking, various moods such as excitement, sadness, or enjoyment may intertwine. Consequently, our project will focus on implementing music recommendation through the lens of speech emotion recognition.

For our final project we aim to apply leading audio analysis technologies to identify moods from user-generated sound inputs. The software will basically utilize user speech to create unique and customized playlists that correspond with the identified mood. With this approach we tried to reinvent music listening by fusing user preferences with state-of-the-art audio analysis to provide users with an unmatched playlist selection tailored to their moods.

This report unfolds as following: section3 will introduce the methods extracting emotions from audio. Next, section4 will discuss the evaluation of the methods. After that, section5 will demonstrate how to generate playlist of recommend songs. Finally, 7 will wrap up and give conclusions.

2 Dataset

We have used the crema dataset [2] for the training, validating and testing of the model. It is a large collection of audio files which consists sentences spoken by a variety of group of people, such as different gender, age. There are six emotion labels, however, as the scope of this project, we only use angry, happy and neutral audio files. Moreover, the audios have four levels of emotions, which is from weak to strong and unspecified. This is also one of the reason we have selected this dataset, as the dataset is greatly diversified by the emotion levels, which in turn contributes to the generalization capability of our model.

3 Emotion Detection

3.1 Feature extracting

We have utilized librosa [3] to extract audio features. Upon reading the entire audio file, there are a number of audio features that available for further analysis. In this project, we have decided to use the following features:

- **Zero Crossing Rate** (ZCR): This measures the rate of signal changing from positive to zero to negative or vice-versa. In the context of audio, it is one of the important features that can be used for determining if an audio is (un)voiced.
- **chroma**: It is a useful feature in musical information retrieval. It represents the harmonic and melodic properties of a piece of audio. In this project, the stft'ed chroma was being used.
- **MFCC**: It stands for Mel-Frequency Cepstral Coefficients. It is a feature related to the power spectrum of a signal. In speech recognition, it can be used for identifying property of speech.
- **rms** Stands for Root Mean Square. It is the average power of audio signal. It can be used for evaluating the loudness of sound.
- **melspectrogram** A spectrogram combined with Mel scale, which reflects human's awareness of pitch of sound. The melspectrogram can be used for capturing speech.
- **tempo** the speed of music piece, measured in BPM (Beats Per Minute)

3.2 Model Selection

We implemented a multi-layer perceptron (MLP) and a convolutional neural network (CNN) to encode and analyze the extracted audio features. These two strategies correspond to different feature addressing functions.

In the MLP, we flatten the extracted feature into a long vector as the model input. The model structure consists of three hidden layers that are activated by `relu`. Before activation, we apply batch-normalization techniques. The output is the probability distribution for three labels. The key idea of this approach is to use the complete feature directly, which remains the time sequence information.

We developed a more complex CNN architecture consisting of two convolutional layers and two fully connected layers. We introduce dropout and batch-normalization to reduce the probabilities of over-fitting. In the data preprocessing stage, we reshape the extracted audio features into a 9x18 matrix, where different features occur in different rows. We use the 3x3 convolutional kernels to scan the local regions of this matrix, each kernel can extract different combinations of those features. Therefore, the model can learn the relationships between diverse audio features without the limitation of time sequence.

Besides implementing the end to end process that is from feature extraction to determining the speech emotion, we have explored utilizing pretrained models from hugging face. The model which we have tried with was ehcalabres/wav2vec2-lg-xlsr-en-speech-emotion-recognition [4]. It is a fine-tuned model that is able to detect 8 different kinds of emotions. Initially, we have attempted to do another fine-tune on this model basing on other datasets so that the capability of the model can be generalized. However, we were unable to find the source code of the model, therefore it became unknown to us that how the features were being processed. Consequentially, we were unable to do secondary development, but rather use it as one of the candidates to look for a model of better performance.

4 Experiment

We completed the MLP and CNN experiments on a Windows PC with CUDA 11.8, both models are implemented through PyTorch. We utilized random seeds 42 to ensure the models are trained on the same data. We set the hyper-parameters for MLP and CNN as batch_size = 32, epochs = 1000, and learning_rate = 0.003.

Figures 1 and 2 show the Accuracy and Loss variations during the training process. The curves in the plots are smoothed by averaging the data over 55 epochs. The MLP converge slower than CNN and displays an unstable upward trend. The MLP trained on complete audio features seems affected by the noise from the time sequence and less considers the contribution of different features. In contrast, the CNN learned local relationship patterns within the features, helping to identify the importance of multi-feature blocks, which enhances the model's robustness.

Both the CNN and the MLP show over-fitting, where the accuracy increases significantly on the training set but is stuck on the validation set after 500 epochs. On the testing set, the MLP achieved 0.54 accuracy and the CNN reached

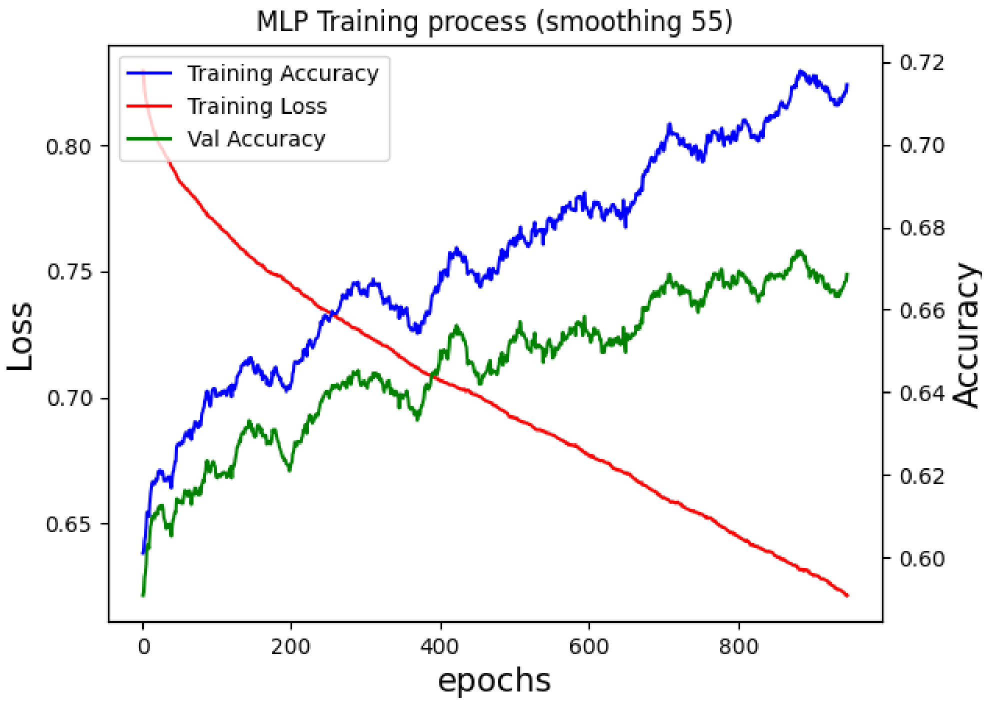

Figure 1: Accuracy on testing set: 0.5494

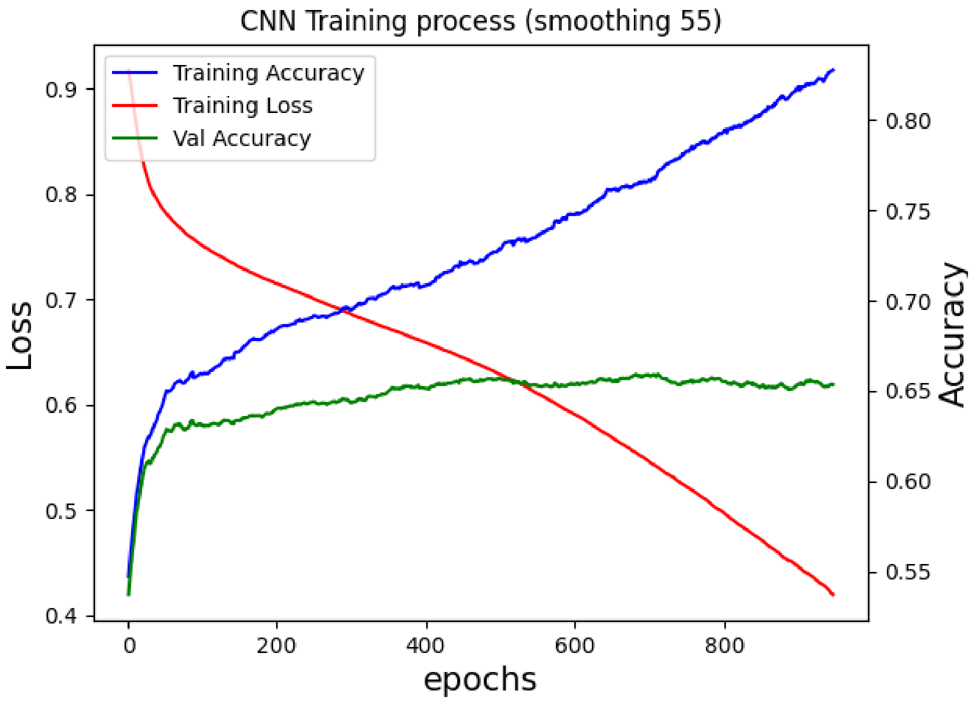

Figure 2: Accuracy on testing set: 0.6483

0.64, which aligns with their performance on the validation set. The reason for overfitting and low accuracy is that distinguishing between Sadness and Neutral emotions is challenging for these models. The audio features of these emotions are similar, which requires more complex approaches to find their potential relationship and differences.

Furthermore, We also have tested the aforementioned model from hugging face. However, the model had a poor performance, as the accuracy only reached 8.5%. We believe this is due to the significant dataset difference. The dataset used for fine-tuning the hugging face model is RAVDESS [5].

5 Playlist Generation

The playlist generator is designed as a back end Node.js server which listens port:3004. In addition, the back end server is using express framework for routing and handling HTTP requests. The app is build for receiving GET requests to its main route and return a playlist to the client or python emotion detection application.

5.1 Architecture

As shown Figure 3 after the emotion has been detected from the emotion detector the playlist Node.js application receives two important input (Tempo and Emotion Tag).

5.1.1 Data Set

The song list is an csv file called `data_moods.csv` which is a Spotify music list from an open source github repository [6]. The data set was a perfect fit for our implementation because it consists of songs labeled with emotions and tempo.

5.1.2 Search CSV

This function unique function is created to search and iterate through csv file to push music objects to a data structure. It basically receives emotionTag and tempo to return an array of music objects. In addition the tempo is used for defining the order of playlist he users with lower tempo expectations will have a playlist according to ascending tempo order.

As a result the array of music list has been send as a response to the request and it has also saved in a `music_list.json` file.

6 Running Application

In order to run Node.js server first the dependencies and form js directory node modules should to be installed and then it is ready to compile:

```
npm install
```

```
npm run server
```

As soon as the server is live, it will be listening port:3004. After that the python dependencies should be installed for speech emotion detection.

```
pip3 install -r requirements.txt
```

In order to demonstrate our application `python/predict_demo.ipynb` has been used. It basically receives speech wav files to make the predictions according to trained CNN model. Finally the algorithm compiles `call_server()` function to send HTTP request to backend server which will be used for playlist generation. The backend will send a response with a playlist.

7 Discussion and Conclusion

We have built a music recommender based on the speech emotion. It is an integration of speech emotion recognizer and a playlist generator. In the speech emotion recognizer, we have developed two models with MLP and CNN. Meanwhile, we also attempted to use pre-trained models from hugging face. As the result, we found that the pre-trained had poor performance which might due to low generalization capability across different datasets. Among our models, the CNN model performed better than the MLP. Therefore, we have determined to use CNN for the recognizer.

For the playlist generation the emotions tags have been efficiently used to provide a well structured playlist. The data set has been successfully transformed into the proper data structure, enabling efficient utilization in playlist generation

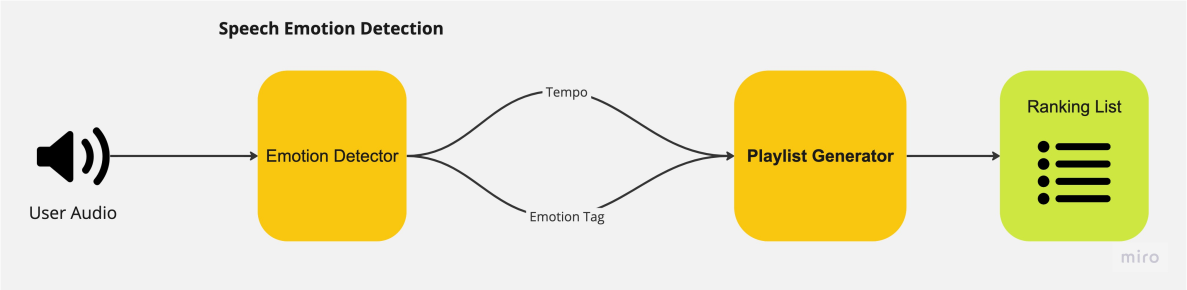

Figure 3: Playlist Generator

and facilitating mood based searches. In addition, the user specific tempo feature has been used to define the order of the playlist.

Speech emotion recognition is still an active field involving intensive research. There have been a plenty of methods that have been investigated. Among these, deep learning method has stood out. Moreover, combining insights from neural science seems to be a promising method [7]. In our project, we have performed preliminary attempts of building more complex model to detect emotions more accurately. Looking ahead, besides the accuracy of recognition, the speed also plays a crucial role by taking the user experience of mobile app into consideration.

The public Github repository of the project:

`https://github.com/dervisonurgurbuz/Audio-Processing-Project/tree/submission`

References

[1] Yading Song, Simon Dixon, and Marcus Pearce. A survey of music recommendation systems and future perspectives. In *9th international symposium on computer music modeling and retrieval*, volume 4, pages 395–410. Citeseer, 2012.

[2] Houwei Cao, David G Cooper, Michael K Keutmann, Ruben C Gur, Ani Nenkova, and Ragini Verma. Crema-d: Crowd-sourced emotional multimodal actors dataset. *IEEE transactions on affective computing*, 5(4):377–390, 2014.

[3] Brian McFee, Matt McVicar, Daniel Faronbi, Iran Roman, Matan Gover, Stefan Balke, Scott Seyfarth, Ayoub Malek, Colin Raffel, Vincent Lostanlen, Benjamin van Niekirk, Dana Lee, Frank Cwitkowitz, Frank Zalkow, Oriol Nieto, Dan Ellis, Jack Mason, Kyungyun Lee, Bea Steers, Emily Halvachs, Carl Thomé, Fabian Robert-Stöter, Rachel Bittner, Ziyao Wei, Adam Weiss, Eric Battenberg, Keunwoo Choi, Ryuichi Yamamoto, CJ Carr, Alex Metsai, Stefan Sullivan, Pius Friesch, Asmitha Krishnakumar, Shunsuke Hidaka, Steve Kowalik, Fabian Keller, Dan Mazur, Alexandre Chabot-Leclerc, Curtis Hawthorne, Chandrashekhar Ramaprasad, Myungchul Keum, Juanita Gomez, Will Monroe, Viktor Andreevitch Morozov, Kian Eliasi, nullmightybofo, Paul Biberstein, N. Dorukhan Sergin, Romain Hennequin, Rimvydas Naktinis, beantowel, Taewoon Kim, Jon Petter Åsen, Joon Lim, Alex Malins, Darío Hereñú, Stef van der Struijk, Lorenz Nickel, Jackie Wu, Zhen Wang, Tim Gates, Matt Vollrath, Andy Sarroff, Xiao-Ming, Alastair Porter, Seth Kranzler, Voodoohop, Mattia Di Gangi, Helmi Jinoz, Connor Guerrero, Abduttayyeb Mazhar, toddrme2178, Zvi Baratz, Anton Kostin, Xinlu Zhuang, Cash TingHin Lo, Pavel Campr, Eric Semeniuc, Monsij Biswal, Shayenne Moura, Paul Brossier, Hojin Lee, and Waldir Pimenta. librosa/librosa: 0.10.1, August 2023.

[4] ehcalabres/wav2vec2-lg-xlsr-en-speech-emotion-recognition · Hugging Face — huggingface.co. `https://huggingface.co/ehcalabres/wav2vec2-lg-xlsr-en-speech-emotion-recognition`. [Accessed 18-12-2023].

[5] Steven R. Livingstone and Frank A. Russo. The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS), April 2018.

[6] Spotify-machine-learning song list. `https://github.com/cristobalvch/Spotify-Machine-Learning/blob/master/data/data_moods.csv`. [Accessed: December 15, 2023].

[7] Gang Liu, Shifang Cai, and Ce Wang. Speech emotion recognition based on emotion perception. *EURASIP Journal on Audio, Speech, and Music Processing*, 2023(1):22, 2023.