50.038 CDS Project

Fink Ployd
Claudia Chin 1004328
Arnold Lim 1004117
Koh Jun Hao 1004295

3 December 2021

1 Introduction

Music is very much a part of our daily lives. We hear it in movies, in shops, on social media, and all these sound clips are meant to illicit a certain mood from us. Music mood classification is hence a problem with a variety of uses, and our project aims to develop a model that can solve the problem.

While there are many different models for music classification, one of the more well-known models is the Valence-Arousal Circumplex Model. Developed by James Russell, this model suggests that emotions are distributed in a two-dimensional circular space, containing arousal and valence dimensions, with arousal on the vertical axis and valence on the horizontal axis. Our group has chosen this model for the purpose of categorising our data.

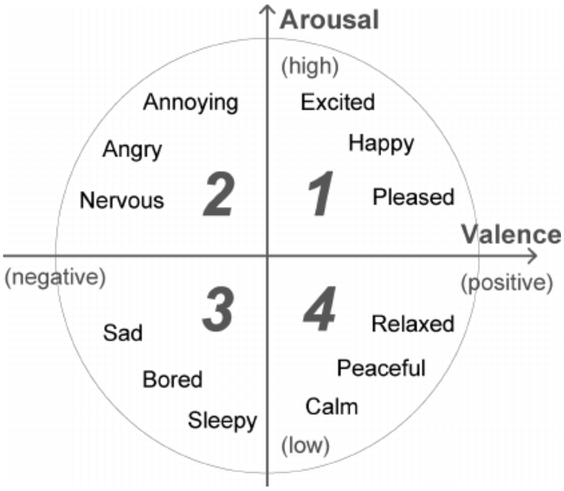


Figure 1: Valence-Arousal Circumplex Model

Arousal (or intensity) is the level of autonomic activation that an event creates, and ranges from calm (or low) to excited (or high). Valence, on the other hand, is the level of pleasantness that an event generates and is defined along a continuum from negative to positive.

2 Dataset and Collection

2.1 DEAM Dataset

The database for emotional analysis of music (DEAM) consists of 1802 excerpts and full music annotated with valence and arousal. DEAM obtained 260 low-level features (mean and standard deviation of 65 low-level acoustic descriptors, and their first-order derivatives) from non-overlapping segments of 500ms.

According to the DEAM dataset creators, it is a known issue that the annotators need some initial orientation time (IOT), before their continuous annotations become meaningful and reliable. Hence, they remove the first 15 seconds of the annotation from the benchmark data. As such, the sample data becomes relevant only from the 15 second mark, or in the terminology of the dataset, sample_15000ms.

2.2 MFCC (Librosa)

Mel-Frequency Cepstral Coefficients (MFCCs) is a feature set commonly used in signal analysis as a representation of the power spectrum of the signal wave.These features are not only used in music mood representation as in our case, they are also prominent in other audio-related machine learning tasks such as speech recognition. It is derived through a multi-step process. First the signal is framed into multiple short frames, before taking the Fourier transform of these short frames and mapping them onto the mel scale. The mel scale allows the audio feature to better mimic the human perception of pitch, since the human perception does not follow a linear scale – for example, to a listener, a difference between a 1000 Hz and a 1100 Hz sound wave is more perceivable than the difference between a 10,000 Hz and a 10,100 Hz sound wave, despite both cases having the same numerical difference in frequency. Afterwards, the logs of the powers are taken and passed through a discrete cosine transformation, giving us a set of amplitudes which we know as MFCCs.

For the project, we used the Librosa package to calculate the MFCC data from the DEAM audio files, giving us 20 MFCCs over 1920 frames for a 45 second audio clip. The data for all 1802 audio files is then saved in a 3D numpy array of shape (1802, 1920, 20) to be saved and fit into the model.

2.3 ComParE 2016 (OpenSmile)

ComParE 2016 refers to the INTERSPEECH 2016 Computational Paralinguistics Challenge. It addresses three different problems for the first time in research competition under well-defined conditions: classification of deceptive vs. non-deceptive speech, the estimation of the degree of sincerity, and the identification of the native language out of 11 L1 classes of English L2 speakers.

The ComParE 2016 dataset can be extracted on three different levels:

- Low-Level descriptors (LDDs)
- Low-Level descriptors deltas (LDDDs)
- Functionals

Both LDD and LDDDs provides 65 features in a time series dataset while the functional level provides 6373 features.

3 Data Pre-processing

3.1 Labelling the Data

Following the valence arousal scale, we label the music into 4 categories:

- Positive Valence & Arousal corresponds to label 0 or AP (Active-Positive).
- Negative Valence & Positive Arousal corresponds to label 1 or AN (Active-Negative).
- Positive Valence & Negative Arousal corresponds to label 2 or PP (Passive-Positive).
- Negative Valence & Arousal corresponds to label 3 or PN (Passive-Negative).

3.2 DEAM Dataset

For the DEAM dataset, we wanted to arrange the 1802 excerpts such that each of its 60 samples of 500ms would be labeled. In other words, for the feature data, the size would be (1802 excerpts, 60 samples, 261 features), and for the feature labels, the size would be (1802 excerpts, 60 samples, 4 labels). Hence, the following steps were taken to process the DEAM dataset:

1. **Prepare valence and arousal data.** After converting valence.csv and arousal.csv into a Pandas Dataframe, the Dataframes must be transposed such that each row contains the 261 feature data points, other than the first row which contains the song IDs.
2. **Get data for each sample in song.** For each song excerpt, read 30 seconds worth of samples, i.e. 60 samples. These data can be saved into a Python dictionary for easy retrieval.

3. **Get labels for each sample in song** For each sample in each song excerpt, categorise the sample based on the corresponding valence and arousal values read in Step 1. These labels should then be one-hot encoded (i.e. from label 1 to label [0,1,0,0]). These labels can be saved into a Python dictionary for easy retrieval.

4. **Build the numPy arrays X and y.** By looping through the keys in the dictionaries created in Step 2 and 3, add the data sequentially to create numPy arrays X and y to be used by the model for learning.

3.3 MFCC (Librosa)

The MFCC data generated using the Librosa package gives us 20 features per frame. The number of frames however differed slightly between each song, at around approximately 1930 frames. This is because the audio length is not necessarily exactly at 45 seconds long - rather, there were quite a number of songs that were a few milliseconds longer than the others. To simplify the data input, we truncated all the MFCC data to be 1920 frames to standardise the input shape. The resulting data shape comes up to be (1802, 1920, 20), where 1802 refers to the number of songs in the dataset.

To keep the MFCC data in line with the 0.5 seconds interval of the OpenSMILE features, we split the frames further into 60 pockets of 21 frames, since there is 21 frames for every 0.5 seconds of audio. This means that the CNN only processes the MFCC data as a (21, 2) Numpy array, in a time series of length 60. The final processed data shape comes up to be a 4D matrix of shape (1802, 60, 21, 20).

3.4 ComParE 2016 (OpenSmile)

Originally, we attempted to use the Functional level dataset which comprises of 6337 features for each song. However, after using this data to train the models, we realised that our models tend to overfit due to the large number of features and did not provide a very good test accuracy score (around 0.22). Instead, we opted to use the LDD level dataset which provides us with 65 features for each 500msec sample of the song.

Moreover, since we had extracted MFCC information from the songs through Librosa, we chose to exclude the mel-spectrogram features that are being provided in the LDD dataset. This is to avoid having two datapoints for the same feature.

4 Algorithm/Model

4.1 Regression

There are two different approaches to solving the issue of determining mood from music. The most straightforward way is solving a classification problem, where mood is split into classes. These classes could represent the four quadrants in the valence-arousal model. A model would be trained to classify a piece of music to one of these four quadrants given the feature set.

We also experimented with using the model to perform a regression task - first the model predicts the valence and arousal values, before being evaluated by determining which quadrant the resulting values belong to afterwards. For both approaches, the base model remains relatively the same as we experimented, with the only difference being the output layer and its activation function − softmax activation in the case of classification, and none for regression.

4.2 CNN Model

The nature of the spacial relationship between MFCC features is such that to process it as a sequence or a fixed set of numerical features may lead to overfitting, similar to training a dense neural network model on image data. Rather, we used 1-dimensional convolutional neural networks to learn the internal representation of the data. This led to a significant increase in validation accuracy, as compared to when MFCC data is fed into a dense neural network as part of an OpenSmile feature set. Our final model contains three layers of 1 -dimensional convolutional layers, each followed by a max pooling layer. The kernel size of the first convolutional layer is larger compared to the others, as we found the performance is better. This could be attributed to the first layer being responsible for extracting the surface level feature representations of the MFCC data first, whereas the rest of the layers extract the deeper features with more details.

We experimented with both using the CNN model as a prediction model by itself and as a component of the final model, producing a feature vector to best represent the MFCC data and combined with the given OpenSMILE features to feed into a LSTM model. Given that the input data is a time series, we wrapped each layer in a TimeDistributed wrapper to apply the convolutional layer on each individual time frame in the input.

4.3 RNN-LSTM Model

The input data into the LSTM model is a time series data consisting of a 256-long feature vector representing the MFCC features and 261 OpenSMILE features as provided in the DEAM dataset. These two data are then concatenated together. Since this concatenated data is already in a sequence with length 60 representing the number of time steps, we can pass it directly into a LSTM layer. We stack two LSTM layers on top of one another to create a deeper network, understanding that the additional LSTM layer would be able to combine the learned representations from the previous layer to create deeper representations of the data. However, we found that increasing the number of LSTM layers further did not lead to an improvement in performance.

The output of the second LSTM layer is then passed into a Dense layer, before being passed to the output layer, which changes depending on the task at hand - for regression, the output layer is a dense layer of 2 units with linear function; whereas for classification the output layer has 4 units with softmax activation function.

5 Evaluation Methodology

5.1 Train, Test, and Validation Splits

To ensure that there was ample training, validation and testing, we used a train-test split of (0.8, 0.2), and further split the training data into training and validation sets of train-validation split (0.8, 0.2).

5.2 Evaluation Metrics

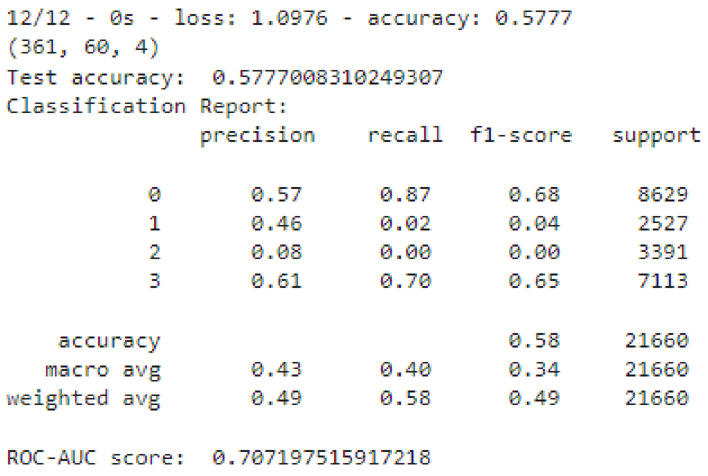
When discussing the results of the model, the following metrics will be used:

- Accuracy
- Area Under Curve (AUC)
- F1 score
- Mean Absolute Error (for regression)
- Confusion Matrix

6 Results and Discussion

6.1 Without Regression, Per Sample

The first variation we ran was without using regression, and classified the data per 500ms samples.



```
12/12 - 0s - loss: 1.0976 - accuracy: 0.5777
(361, 60, 4)
Test accuracy:  0.5777008310249307
Classification Report:
              precision    recall  f1-score   support

           0       0.57      0.87      0.68      8629
           1       0.46      0.02      0.04      2527
           2       0.08      0.00      0.00      3391
           3       0.61      0.70      0.65      7113

    accuracy                           0.58     21660
   macro avg       0.43      0.40      0.34     21660
weighted avg       0.49      0.58      0.49     21660

ROC-AUC score:  0.707197515917218
```

Figure 2: Results for "Without Regression, Per Sample" classification

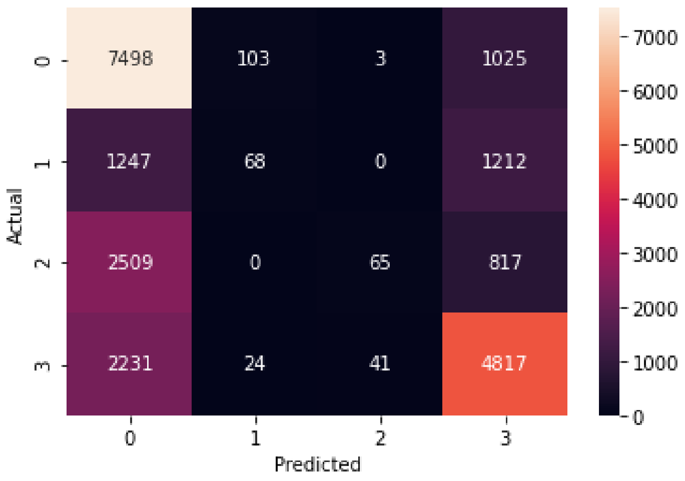
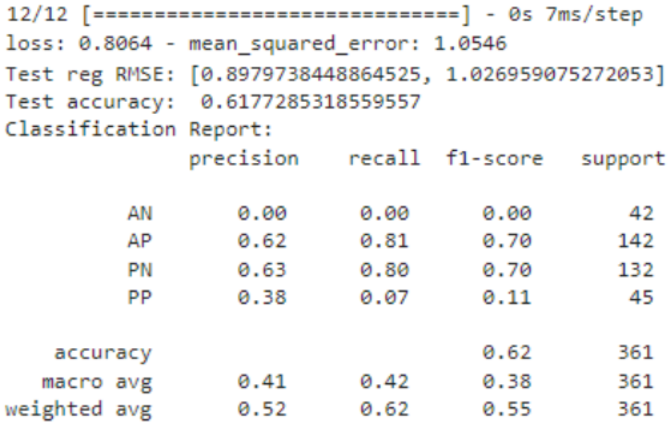


Figure 3: Confusion matrix for "Without Regression, Per Sample" classification

6.2 With Regression, Per Song

The next variation we ran used regression to categorise the data according to valence and arousal, and classified the data per the entire song. For regression per song, we did not have a ROC-AUC score as the model does not output a probability for the final prediction. In order to obtain a more detailed analysis of the output, we should look at the valence and arousal score.

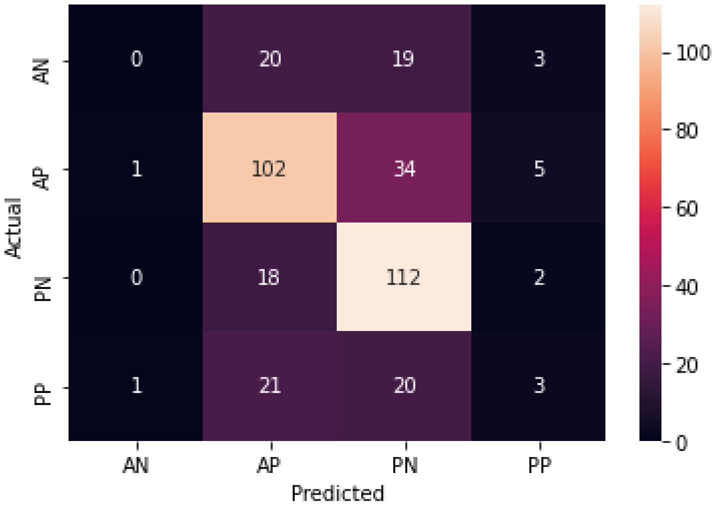


```
12/12 [==============================] - 0s 7ms/step
loss: 0.8064 - mean_squared_error: 1.0546
Test reg RMSE: [0.8979738448864525, 1.026959075272053]
Test accuracy:  0.6177285318559557
Classification Report:
              precision    recall  f1-score   support

          AN       0.00      0.00      0.00        42
          AP       0.62      0.81      0.70       142
          PN       0.63      0.80      0.70       132
          PP       0.38      0.07      0.11        45

    accuracy                           0.62       361
   macro avg       0.41      0.42      0.38       361
weighted avg       0.52      0.62      0.55       361
```

Figure 4: Results for "With Regression, Per Song" classification



Actual \ Predicted	AN	AP	PN	PP
AN	0	20	19	3
AP	1	102	34	5
PN	0	18	112	2
PP	1	21	20	3

Figure 5: Confusion matrix for "With Regression, Per Song" classification

6.3 Comparison with Existing Models

Our work on the DEAM dataset is largely inspired by the paper 'Music emotion recognition using convolutional long short term memory deep neural networks', given the close similarity between our objectives and training task. The main structure of our model follows the same approach as the paper, which is to first extract a feature vector representation of the MFCC data using a convolutional neural network, before combining with standard features to fit into a long-short term memory model that ultimately outputs a result at the end after a dense layer. Firstly, we opted to exclude the Log Mel Filterbank energies as part of the input into the CNN, given to close similarity in meaning between both MFCC and log-mel filterbank energies features. Furthermore, while the paper's approach is that of a classification task, we ultimately settled on our model as a regressor, first determining the valence and arousal from the audio features before using these predicted values as needed - in our case representing them in categories so we can evaluate the accuracy of the model. Refer to Annex 1 for comparison of models.

The DEAM dataset is significantly more varied as compared to the dataset used by the researchers, which was collected by the researchers themselves and the genre scope limited to Turkish traditional music. In contrast, the DEAM dataset comprises of different genres, which could have contributed in no small part to the different in featuress between the songs. As such, while the researchers were able to achieve an overall accuracy of 99.19%, in our case the larger and more varied dataset contributed to a lower accuracy in our final model.

7 Conclusion

The Regression-CNN-RNN-LSTM Model achieved 61.77% accuracy with a mean average error loss of 0.8064, making it the best variation for our model. Compared to the research papers referenced for the inspiration of our model, while we could not achieve similar results, we believe that our accuracy is justified based on the difference in dataset and model used.

In the future, things we could improve

- From the confusion matrix, we find that the 'Active Negative' and 'Passive Positive' classes to have a significantly lower performance as compared to the other two. This could be attributed to these two classes being underrepresented in the DEAM dataset. One possible improvement would be to append more songs from these two classes in our dataset. In the case where these classes do not have any impact in our collected audio features (for example, an 'Active Negative' song may look identical to a 'Active Positive' song when the MFCC data is plotted, given that both classes have high arousal), we could look at more in-depth features, such as Mel-Spectograms which gives a more complete picture of the audio data.
- The variation between genres and song information in the DEAM dataset is significant, which could have a influence on the extracted features. Furthermore, we did not have information on the song titles and names so that we could look at their metadata such as genre or artist. As such, the model is unable to differentiate whether the difference in features is attributed to the difference in mood or in other categories. One improvement would be to get a dataset that has the song metadata information such that they could be included in the model for a better prediction result, or limit the scope of the dataset to a specific genre or a set of artists.

Github: `https://github.com/Pillowkoh/Fink_Ployd/`

8 Citations

1. Bestelmeyer, P., Kotz, S. A., & Belin, P. (2017). Effects of emotional valence and arousal on the voice perception network. Social cognitive and affective neuroscience, 12(8), 1351–1358. https://doi.org/10.1093/scan/nsx059.
2. Aljanaki A, Yang Y-H, Soleymani M (2017) Developing a benchmark for emotional analysis of music. PLoS ONE 12(3): e0173392. https://doi.org/10.1371/journal.pone.0173392.
3. Serhat Hizlisoy, Serdar Yildirim, Zekeriya Tufekci, Music emotion recognition using convolutional long short term memory deep neural networks, Engineering Science and Technology, an International Journal, Volume 24, Issue 3, 2021, Pages 760-767, ISSN 2215-0986, https://doi.org/10.1016/j.jestch.2020.10.009.
4. Organisation and Chairing Special Session "INTERSPEECH 2016 Computational Paralinguistics Challenge (ComParE)", Björn Schuller, Stefan Steidl, Anton Batliner, Julia Hirschberg, Judee Burgoon, Eduardo Coutinho, INTERSPEECH 2016, ISCA, San Francisco, CA, 08.-12.09.2016. http://www.tangsoo.de/documents/Publications/Schuller16-TI2.pdf
5. Nair, P. (2018, July 27). The dummy's guide to MFCC. Medium. Retrieved December 1, 2021, from https://medium.com/prathena/the-dummys-guide-to-mfcc-aceab2450fd.

Annex 1

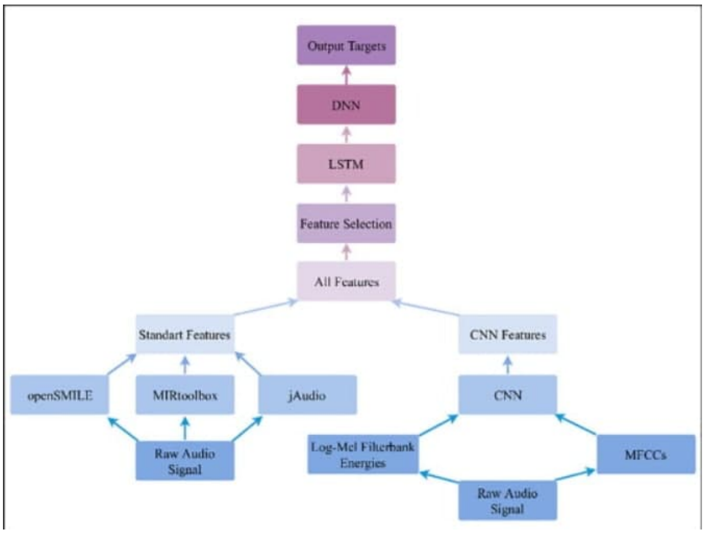


Figure 6: 'Music emotion recognition using convolutional long short term memory deep neural networks' model

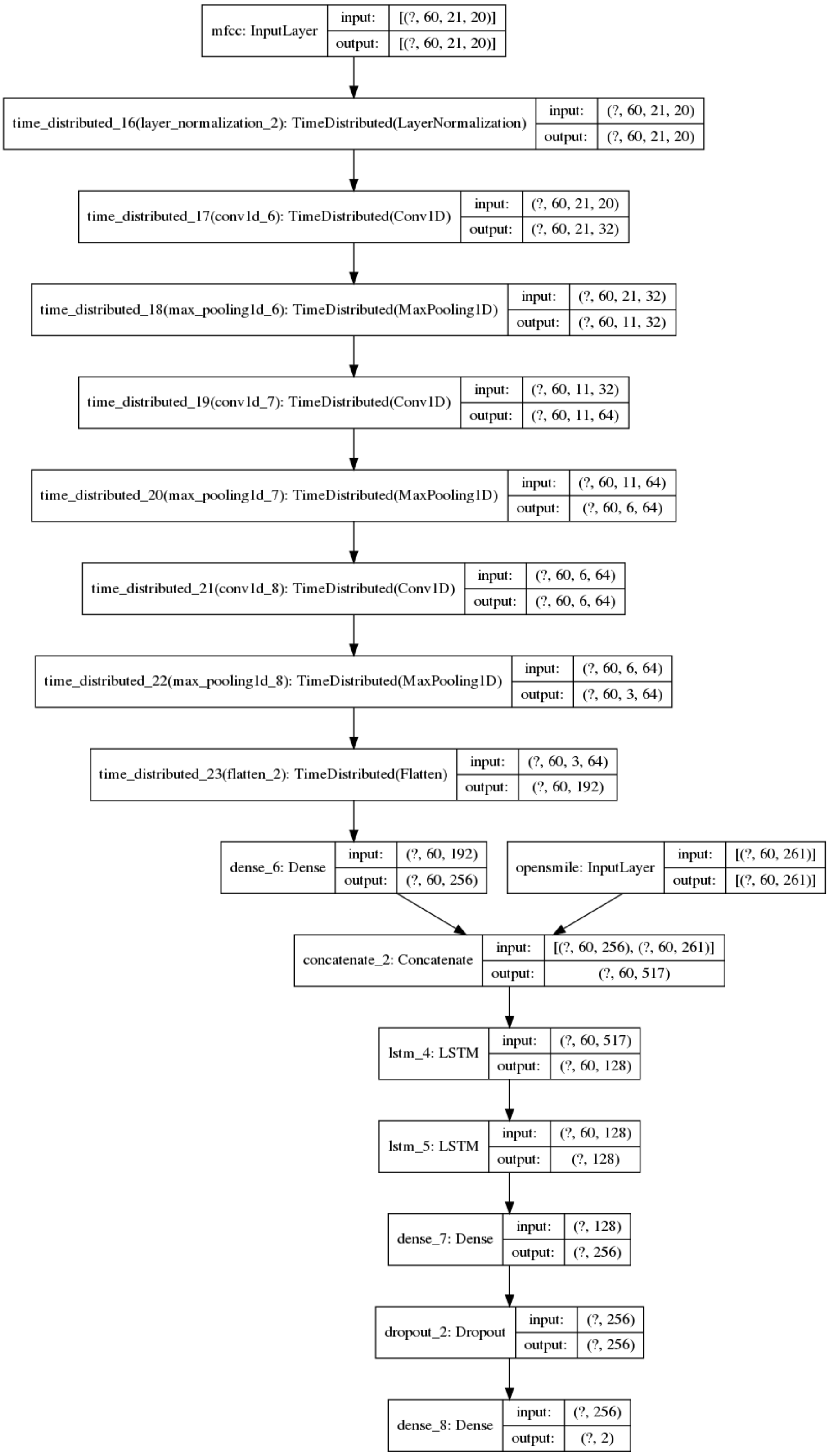


Figure 7: Our model