Emotion Recognition in Music through Deep Learning techniques

Denis-Angel Moldovan

Abstract

Music, as many people say, a *universal language*, is a method that removes the language and cultural barriers between people throughout the globe. This paper dives into the subjective domain of emotion recognition in nowadays' trending music, aiming to decipher the relationship between musical elements and human emotions. Through deep learning techniques, I explore a small part of the rich emotional landscape embedded within music art. This study encompasses a diverse dataset regarding styles and emotions transmitted through each song. Leveraging state-of-the-art feature extraction methods and advanced deep learning models, I unveil patterns of correlation between regular features like tone, rhythm and more complex features such as the audio spectogram of a song and human emotions through artificial neural networks. Using a convolutional approach and turning the song into an image gave me an accuracy score of 80% in guessing the emotion transmitted by a song.

1 Classification

1.1 AMS Mathematical Subject Classification

Statistics and Data Analysis (AMS 62)

1.2 ACM Computing Reviews Categories and Subject Descriptors

- Information Search and Retrieval (H.3.3)
- Learning (I.2.6)

This paper aligns with the AMS classification of Statistics and Data Analysis (AMS 62), focusing on statistical methods and data analysis techniques. Furthermore, within the ACM Computing Reviews Categories, it can be categorized under Information Search and Retrieval (H.3.3) due to its use of search algorithms, and Learning (I.2.6) focusing on machine learning.

2 Introduction

Music, a profound word that underlines the expression of human creativity and emotion. It transcends linguistic and cultural boundaries, resonating on an unimaginable deep level within each individual on this entire planet. The capacity of it to evoke and convey emotions makes it a powerful medium for humans to express themselves and making others understand how they feel. Understanding and deciphering the emotional content embedded within music have aroused profound interest in recent years. Emotion recognition in music, the ability to discern and interpret the emotional nuances expressed through melodies and rhythms, holds substantial relevance in several domains. Primarily, the study of emotion recognition in music is crucial in unraveling the mechanisms underlying human response to auditory stimuli. Gaining a good understanding on how musical features trigger different parts of the brain that are responsible with emotions contributes t advancements in psychology and neuroscience and on top of that, it helps the development of intelligent systems capable of perceiving human emotions. Also, in different applications, recognizing emotions for each individual in music plays a crucial role for developers to bring a more comfortable and therapeutic time while being online. On top of that, on the media and entertainment side, producers would get a better understanding of how to correlate videos with the right music in order to intensify the emotions of people observing.

The challenge with evaluating music emotion recognition systems comes from the subjective nature of how people perceive emotions transmitted by a song. Due to the fact that emotions are deeply personal and are directly influenced by a variety of elements such as culture, preferences, past experiences that might bring memories from a person's life. Establishing a uniform classification is complicated by the fact that various listeners may perceive a song in different subjective ways. Therefore, there is no universal truth on how to categorize songs to emotions, which makes it a hard process of benchmarking such systems.

This paper dives into the domain of emotion recognition in music, aiming to find a grain of understanding of the subjective and complex interplay between musical elements and emotion recognition using regular *fully connected neural networks*, but also a *convolutional* approach trying to turn a song into an image.

In the following sections of this article, I will summarize previous works in this domain, walk you through the entire process of obtaining the results including the dataset, feature extraction, the experiments I have conducted and conclusions regarding this work.

3 Previous works

Compared to image classification where the ground truth is almost the same to every human being when given the prompt, music is way different. Some songs could easily be categorized in different classes of emotions by various people. A sad song could easily be categorized as a calm song for instance.

In [8], a regression approach is used to determine the emotion of the given song. To solve the dependency between arousal and valence and predict AV values, the paper formulates music emotion recognition as a regression problem. Regarding feature extraction two professional computer programs are used: "PSYSOUND"[1] and "Marsyas"[7]. The dataset consists of 195 popular songs selected from a number of Western, Chinise and Japanese albums. The features extracted by the programs are numerous: loudness, dissonance based on psychoacoustic models, timbral texture, rhythm, pitch, spectral contrast etc. Each song had to be uniformly distributed in each quadrant of the emotion plane. Support Vector Regression outperforms other algorithms, reaching 58.3% for arousal and 28.1% for valence, plotted as a point on the Thayer's arousal-valence emotion plane. This regression approach significantly improves emotion variation detection within music selections.

In [2], a similar AV prediction is used, however the Thayer's emotion plane is divided into more areas, big enough to store 11 emotions. The dataset is made up of 165 western pop songs. They have equally distributed each training song with each emotion and used various features such as scale, average energy, rhythm and harmonics, which they thoroughly mathematically describe each and every one. SVMs are used to try to predict the emotions of the songs. Their accuracy of prediction reaches 94.55% on the genre of music used for training.

In [6], on an instrumental music only database, where they use RNNs and similar features found in this paper (MFCC, CENS, Spectral centroid, bandwidth, rollof, ZCR, Chroma), all combined and then fed into an RNN to classify the songs into Happy, Sad, Neutral and Fear for many instruments, reach a performance of 89.2%.

In [4], a CLSTM architecture is used to find the emotion given in Turkish songs. They use a CNN to extract relevant features from 30s sequences of songs and then feed the output as an input to LSTMs and reach an accuracy of 91.93% on the full feature set when using LSTM+DNN for classification and a 99.19% with CFS applied. After obtaining an 88.70% accuracy on MFCC, they decided to increase the number of features used and apply CFS on them which boosted the model's overall performance to 99.19%.

4 Data collection & Dataset

A dataset of 1162 of popular trending songs labeled to four classes: sad, happy, energetic, and calm. These songs are uniformly almost distributed in these classes. Most of the previous works use western music/instrumentals as they are easier to classify into emotions and are less subjective than the top songs on the radio today. However I have chosen to work with these songs as they are used mostly listened to by the population. For the validation set, I use 120 songs off of the previously mentioned.

The dataset comes with some Spotify extracted features using a Spotify API called Spotipy. All of these songs are almost evenly distributed to the 4 classes. See Figure 1.

Due to the necessity of providing a CNN relevant inputs, it was necessary to somehow store the songs into a matrix. The methodology of downloading all the songs automatically was used through another python script. This script uses "youtube_search" API to search the songs on YouTube by the name of the song and retrieve the corresponding URL. The URL is later passed to a function that would download the video (using PyTube API and requests) and extract only the audio using MoviePy API. The audio was then truncated to use only 30 seconds of the song starting from $\frac{1}{4}$ of the song's duration and save it locally on the hard drive as an mp3 file. Each song took $\tilde{6}$ secoonds to download.

5 Feature extraction

Spotipy is able to provide some unique values for features like danceability, energy, key, loudness, speechiness, acousticness, instrumentalness, liveness, valence, tempo and time signature. Unfortunately the methodology of how these features are extracted is not publicly available.

Also, having the mp3 files for each song in the dataset, we would extract the following features that are thoroughly explained in [5]:

- MFCC (Mel-frequency cepstral coefficients)

  Represents the spectral envelope of the audio signal and capture the power spectrum characteristics. They're derived by applying a series of mathematical operations to the short-time power spectrum, widely used in speech and audio processing tasks.

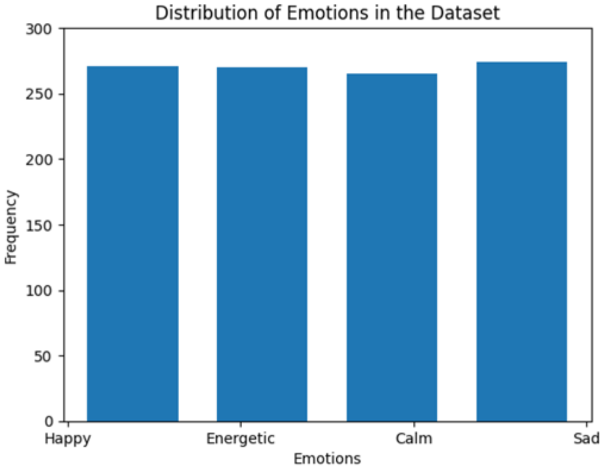

Figure 1: Dataset distibution

- Spectral Centroid

  It indicates the "center of mass" of the spectrum, providing insights into the spectral brightness or the average frequency content of the signal. Higher values correspond to brighter or higher-pitched sounds.

- Chroma Energy Normalized Statistics (CENS)

  CENS computes a chromogram representation, summarizing the distribution of energy in different pitch classes. It's useful for analyzing tonal content in music and is more robust than raw chroma features against variations in tempo and dynamics.

- Mel Spectrogram

  This representation is obtained by converting the linearly spaced frequency bins of a spectrogram into Mel-frequency bins using a Mel filter bank. It emphasizes frequencies relevant to human perception, facilitating tasks like audio classification or genre recognition.

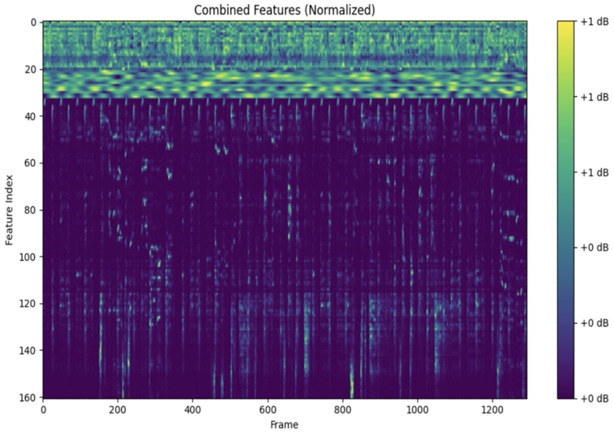

Figure 2: Image sample of a 30s song fed to the CNN

6 Architecture and training

As described in [4], CNN is a good way of extracting features from various inputs. I have decided to approach this problem as an "image classification" problem, where I try to convert a sequence of a song into an image. Just using a few of the features mentioned in the above section, features that were present in [4] aswell, I could easily assemble an image as a song by concatenating the songs into a matrix sequentially. See Figure 2. Each one of these images are fed into the CNN.

The models' performance is measured in accuracy. The training part uses a K-fold validation method with 5 folds, due to the fact that the dataset is small. I used the ResNet18[3] architecture due to the computation speed, size and performance regarding feature extraction. The first and last layers are modified according to the Figure 3. The training for the CNN was made on 10 epochs with 5 folds and it took 1 hour using an RTX 3050 laptop GPU.

- First convolutional layer is modified to use one channel instead of three
- Added a fully connected layer at the end to predict into the 4 classes the data is labeled to
- All layers are frozen except the last one, which is the one that is trained

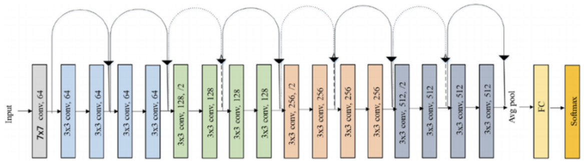

Figure 3: Modified ResNet-18 architecture

Also, having the features extracted with SpotiPy, I modeled an ANN with 4 hidden layers with a decreasing number of neurons: 128, 128, 64, 32 in this order. The training took a very short period of time ($\tilde{2}$ minutes) for 200 epochs.

7 Experiments

Training several different models with different features is the first step on a large understanding in the experiments I have conducted. The models I have experimented with are:

- Spotify features:
  - Logistic Regressor with a Standard Scaler as feature standardizer (for comparison purposes)
  - ANN with various hidden layer sizes and learning rate values
- CNN features:
  - ResNet18, finetuning parameters

The parameters of the best registered accuracies on the conducted experiments are provided below:

For the CNN:

- First convolutional layer using stride of 1, kernel size of 7 and a padding of 2
- Learning rate of 0.0009, SGD optimizer with 0.9 momentum and a learning rate scheduler of step_size=7, gamma=0.1, batch size of 4

After trying with various values for the learning rate, kernel size, stride and different optimizers, the above values gave the best results out of 10 epochs with 5

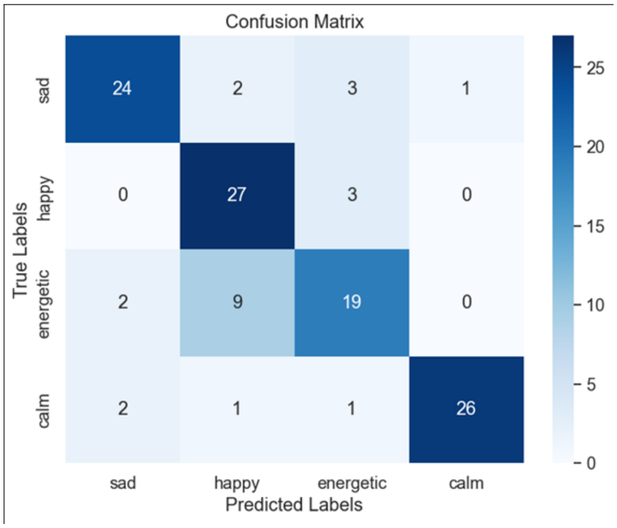

Figure 4: CNN confusion matrix

folds. The model outputs an accuracy of 80% and regular fully connected layers at the end for the classification problem. In [4], just using MFCC as input, they obtained an accuracy of 88.70% on turkish songs with LSTM+DNN classifier.

By plotting the confusion matrix (See Figure 4) with the results that were predicted by the CNN for the test dataset, we can see that songs that were labeled as energetic are often misclassified as happy songs. Which is in fact really hard to differentiate, as happy songs are most of the time energetic too. This is the perfect example where the subjectivity issue regarding this domain of research is highly observed.

A notable thing is that using a bigger kernel size and a relatively big padding, gave better results than using a smaller kernel size with no padding. Also, a bigger learning rate caused the model to oscillate the loss value on each epoch. Plotting the

For the ANN:

- Using a learning rate of 0.006, an Adamax optimizer with step_size=50 and a gamma=0.1

After trying with different values for the learning rate, other optimizers and even adding more hidden layers with more neurons, the above values seem to be the best results out of 200 epochs. The model outputs an accuracy of 84% in average with Spotify's features. Interesting to note is the fact that adding more hidden layers to the network would not only increase the training time, but also would decrease the model's performance dramatically

For the logistic regressor:

- Used only a standard scaler to standardize the features by removing the mean and scaling to unit variance

This approach got an accuracy of 82% in average with Spotify's features. After these conducted experiments, we can safely say that the features extracted by Spotify are slightly greater than the "image" approach of a song classification. However, a CNN could have a far better performance if a better architecture using RNNs with LSTM/GRU gates for learning sequences better than only regular fully connected layers. Compared to [6], where similar features were used, I managed to get to a fairly good accuracy almost close to an RNN (their 89.2% on instrumental music) on popular music.

Comparing the three approaches we can safely say that choosing the right features for this task gives an enormous boost in the performance of the system.

8 Conclusion

The exploration into the realm of emotion recognition in music using deep learning techniques has revealed both the challenges and the beauty of this domain. Music, as a universal language, is able to overcome linguistic and cultural barriers. Inspired by this fact, this paper aims to find a relationship between wonderfully aligned audio, named music and human emotions. The subjective nature of human perception and capacity of feeling different emotions on various songs represented a challenge in evaluating music emotion recognition systems. Emotions, which are deeply personal and directly influenced by past experiences of the listeners, has opposed to a universal classification. Other researchers decided to only classify a single genre of music such as Chinese music, western pop or Turkish music, achieving far better results. This shows the complexity of this area and how difficult it is to differentiate emotions transmitted by a song, even for humans.

By diving into previous works, this research outlined various methodologies and approaches: SVMs, RNNs with LSTM/GRUs, Regression. Notably, the utilization of CNNs, converting music sequences into image representations gave promising results achieving an accuracy of 80% in predicting emotions from the most popular songs that are being broadcasted on radios throughout the entire world.

The CNN's performance, encountered challenges in differentiating between energetic and happy classes, which rigorously traces the subjectivity issue when it comes to recognizing emotions in music. As energetic songs can denote a happy factor and vice versa. Despite the challenges, the study's findings underline the potential deep learning has in extracting emotions out of harmonic audio files.

The experiments conducted on a dataset of trending songs, uniformly labeled into four emotion classes outlined the importance of feature extractions. A regular ANN with 4 hidden layers beat the modified ResNet18 when it comes to emotion recognition hitting an average accuracy of 84%. However, even though the regular ANN approach obtained a fairly good accuracy the problem is that the scalability of it is limited. I thoroughly think that if enough computation power, combining the existing CNN with LSTM/GRUs would significantly boost the accuracy of the model.

In conclusion, this work contributes to the deepen the knowledge in music emotion recognition in popular radio songs, clarifying the complexity, but at the same time opportunities in this field. The combination of deep learning methods and music analysis opens doors to a better understanding of how music can impact the brain, ultimately waking beautiful and personal emotions. Gaining good understanding over this area, promise advancements in various areas, from entertainment to therapeutic activities.

References

[1] Densil Cabrera et al. Psysound: A computer program for psychoacoustical analysis. 24:47–54, 1999.

[2] Byeong-jun Han, Seungmin Rho, Roger B Dannenberg, and Eenjun Hwang. Smers: Music emotion recognition using support vector regression. pages 651–656, 2009.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[4] Serhat Hizlisoy, Serdar Yildirim, and Zekeriya Tufekci. Music emotion recognition using convolutional long short term memory deep neural networks. *Engineering Science and Technology, an International Journal*, 24(3):760–767, 2021.

[5] Renato Panda, Ricardo Malheiro, and Rui Pedro Paiva. Novel audio features for music emotion recognition. *IEEE Transactions on Affective Computing*, 11(4):614–626, 2018.

[6] Sangeetha Rajesh and NJ Nalini. Musical instrument emotion recognition using deep recurrent neural network. *Procedia Computer Science*, 167:16–25, 2020.

[7] George Tzanetakis and Perry Cook. Marsyas: A framework for audio analysis. *Organised sound*, 4(3):169–175, 2000.

[8] Yi-Hsuan Yang, Yu-Ching Lin, Ya-Fan Su, and Homer H Chen. A regression approach to music emotion recognition. *IEEE Transactions on audio, speech, and language processing*, 16(2):448–457, 2008.