# MusicBud: A Music Recommendation System Based on Deep Learning algorithms

**Marius Andrei Negreț, Paul-Stefan Popescu**
University of Craiova,
13 A.I.Cuza Street, 200585 Craiova, Romania
andrei.negret@yahoo.com,
stefan.popescu@edu.ucv.ro

**Mihai Mocanu, Marian Cristian Mihaescu**
University of Craiova,
13 A.I.Cuza Street, 200585 Craiova, Romania
mihai.mocanu@edu.ucv.ro,
cristian.mihaescu@edu.ucv.ro

## ABSTRACT

This paper presents a system built for mood improvement based on a custom-designed sentiment analysis framework. The system uses the Fer2013 dataset to train a deep learning model used for mood detection, and then we match the mood with a specific playlist. The songs are extracted from Spotify API and can be easily upgraded to provide more relevant results for the recommender system. For system development, a mix of technologies and programming languages were used to facilitate user and system interaction and give accurate recommendations for lightweight resource consumption. Our focus was to build a system that would improve the mood by providing good music based on the current mood. The system validation revealed a reliable and easy-to-use system obtaining a good accuracy for our trained model and a trustable validation based on our real-life testing scenarios.

**Author Keywords**
Deep Learning; Sentiment Analysis; Recommender system

**ACM Classification Keywords**
Miscellaneous.

**General Terms**
Human Factors; Design; Measurement.

DOI: 10.37789/rochi.2022.1.1.22

## INTRODUCTION

Music tends to influence our mood and, by this, our entire life. It is also worth mentioning that music, as a relaxing environment, was a popular choice to express mood and sentiments. The choice when it comes to music is very subjective and follows the mood at the moment of choice. One crucial fact regarding music is that it must be correlated with our mood because we tend to listen to different types of music depending on our sentiments. The idea that motivated the development of such a project is that providing the proper music based on a person's mood can significantly increase the quality of life. It is also essential to have a lightweight application that will consume meager resources and run on many computers without special requirements.

This paper presents a project which focuses on providing the best music depending on our mood. The project was designed using a specific architecture to have clients on both web and mobile devices. So, there is a server-side module that computes sentiments and recommends music, and two clients take images and play the received music. In order to achieve the desired results, the system uses deep learning techniques for sentiment analysis and then a match between the computed sentiment and a list of songs that is proper for it. The system is also complemented with a feedback system which can be further used for improving the sentiment analysis module. It uses two signs: one thumbs up icon for validating the emotion and one thumb down icon for a wrong detected emotion.

Regarding the music recommender module, we considered several playlists from Spotify that were already classified as relevant for several emotions. Then, we made a match between the computed emotion and the playlist. After this match is done, a shuffle is made so users will not get the exact same songs for the same emotion, and we also offer the possibility to preview the songs and rerun the emotion detection process if the songs are not relevant enough for that mood.

The target audience for this project is composed of people who mainly work or spend their time on computers. The main requirement for using such an application is to have an enabled webcam and, of course, a web browser. Based on the system design, it is a lightweight application that can mainly run in the background and will consume meagre resources. The system can run correctly even if the webcam does not have to provide high-quality images because we use only low-resolution images. We also added a mobile module that can be used for mood feedback or in case the pc is not equipped with a webcam. In the usual case scenario, the application needs to run on a tab from a web browser and from time to time to update the mood in order to get relevant music. The method used for avoiding song repetition is a shuffle made on the list of songs available for a specific emotion so the person listening to music will not get bored.

From a technical perspective, the project was built on ASP.Net CORE and uses an MVC (Model-View-Controller) architecture, but there is furthermore; it also uses a TensorFlow model and a custom recommendation system based on Spotify API. For the music recommendation module, the Vue framework was used to decouple HTML, CSS and JavaScript components at a high-level view working great on MVC architectures. The most powerful characteristics of the Vue framework are the components that help us extend the essential HTML elements in order to encapsulate reusable code. For code development, two environments were used for mobile devices because of the IOS component for mobile devices: Xcode and Visual studio. For the deep learning module, Tensorflow was used, and the training was made in Python using Google Colab.

## RELATED WORK

This paper is relevant for more than one research area as it uses deep learning techniques, a custom recommender system and an HCI optimized interface.

Regarding the sentiment analysis component, there are plenty of approaches, some very recent as [1], which use a CNN deep learning model trained on the same Fer2013 dataset [2]. Despite the regular results, they obtained the best evaluation resulting in around 0,88, showing that the number of convolution layers, the batch size, the dropout and the epoch number have a significant impact on the results.

Even if there is plenty of work recently published in the area of sentiment analysis, some reviews were published in 2014, like [3], which tackles a comprehensive overview of the last update for that moment in this field. Many proposed algorithms' enhancements for that time and various Sentiment Analysis applications are investigated and presented briefly in this survey. Their analyzed articles are categorized according to their contributions to the various Sentiment Analysis techniques. They also tackle the paper-related fields of Sentiment Analysis (like transfer learning [4], emotion detection [5], and building resources [6]) that attracted researchers. The main target of the survey was to give a nearly full image of Sentiment Analysis techniques and the related fields with brief details. The main contributions of their paper include the sophisticated categorizations of a large number of recent articles and the illustration of the trend of research in sentiment analysis and its related areas.

Still in the area of sentiment analysis is a paper [7] that presents a combined approach. His paper combines rule-based classification, supervised learning and machine learning into a new combined method. The method described in [7] was tested on movie reviews, product reviews and MySpace comments. Their results show that a hybrid classification may improve classification effectiveness. They also use F1, which is a measure that takes both the precision and recall of a classifier's effectiveness into account. In addition, they propose a semi-automatic, complementary approach in which each classifier can contribute to other classifiers to achieve an excellent level of effectiveness. Other approaches present sentiment analysis performed on text as described in [8], which applies their techniques to twits.

Sentiment analysis based on facial expressions became a viral subject as deep learning techniques started to be used in more and more situations like [9] which also uses a CNN network with a specific architecture on the same Fer2013 dataset. Their results show that the results of our experiment have been a very encouraging 57% accuracy, and they state that this was an improvement in the domain of automated analysis of facial sentiments. Still, in this area, another paper uses CNN on Fer2013 (and other datasets) to compute facial emotions, but in this case [10], the analysis is made on groups of emotions. Their proposed framework uses the Haar filter to detect and extract face features. Then the convolutional neural network (CNN) is developed to recognize facial expressions and classify them into five primary emotion states, namely happy, sad, anger, surprise and neutral. Finally, the predicted group emotions are fed into an audio synthesizer to get audio output. The proposed model achieves an absolute accuracy of 65% for Facial Expression Recognition (FER)-2013 and 60% for custom datasets.

Despite the aim for better and better accuracy, some other papers aim for robust systems, like in [11], which built a system for robot interaction. They aim that ss robots are helping human beings; the robots need to understand human emotion and feeling in order to treat humans in a more customized manner. Predicting human emotion has been a complex problem that may be solved over a decade's time. In their paper, we have built a model which can predict human emotion from an image in real-time. Their built network was built on a convolutional neural network, which has reduced parameters by 90× from that of Vanilla CNN and also 50× from the latest state-of-the-art research carried out to the best of our knowledge. The network build was validated robustly using eight different datasets, namely Fer2013, CK and CK+, Chicago Face Database, JAFFE Dataset, FEI face dataset, IMFDB, TFEID, and custom datasets built in their laboratory having different setups.

Lastly, regarding sentiment analysis, a paper focuses on State-of-the-Art, Taxonomies and Challenges [12]. The paper [12] presents a systematic and comprehensive survey of current state-of-art Artificial Intelligence techniques (datasets and algorithms) that provide a solution to the aforementioned issues. It also presents a taxonomy of existing facial sentiment analysis strategies in brief. Then, the paper reviews the existing novel machine and deep learning networks proposed by researchers that are specifically designed for facial expression recognition based on static images and present their merits and demerits and summarizes their approach. Finally, the paper also presents the open issues and research challenges for the design of a robust facial expression recognition system

## SYSTEM DESIGN

The system design can be divided into more than one component because the whole system is composed of the deep learning module, the web application architecture and the mobile (IOS) app. All these three components have to merge and work together in order to offer the best results.

Figure 1 presents a use case diagram of the system. We have three types of users who can perform activities in the system: Logged User, Administrator and Unlogged user. Most of the functionalities are available for the logged user, which can change the profile picture, access all the pages, open the web camera, take a picture, then give feedback and also listen to or preview the recommended song. The Administrator has fewer functionalities but can view users' ratings, read messages from the users, and also access all the pages or even change info (or profile picture) for a user. The unlogged user can still use the application but without the benefits of the feedback system.

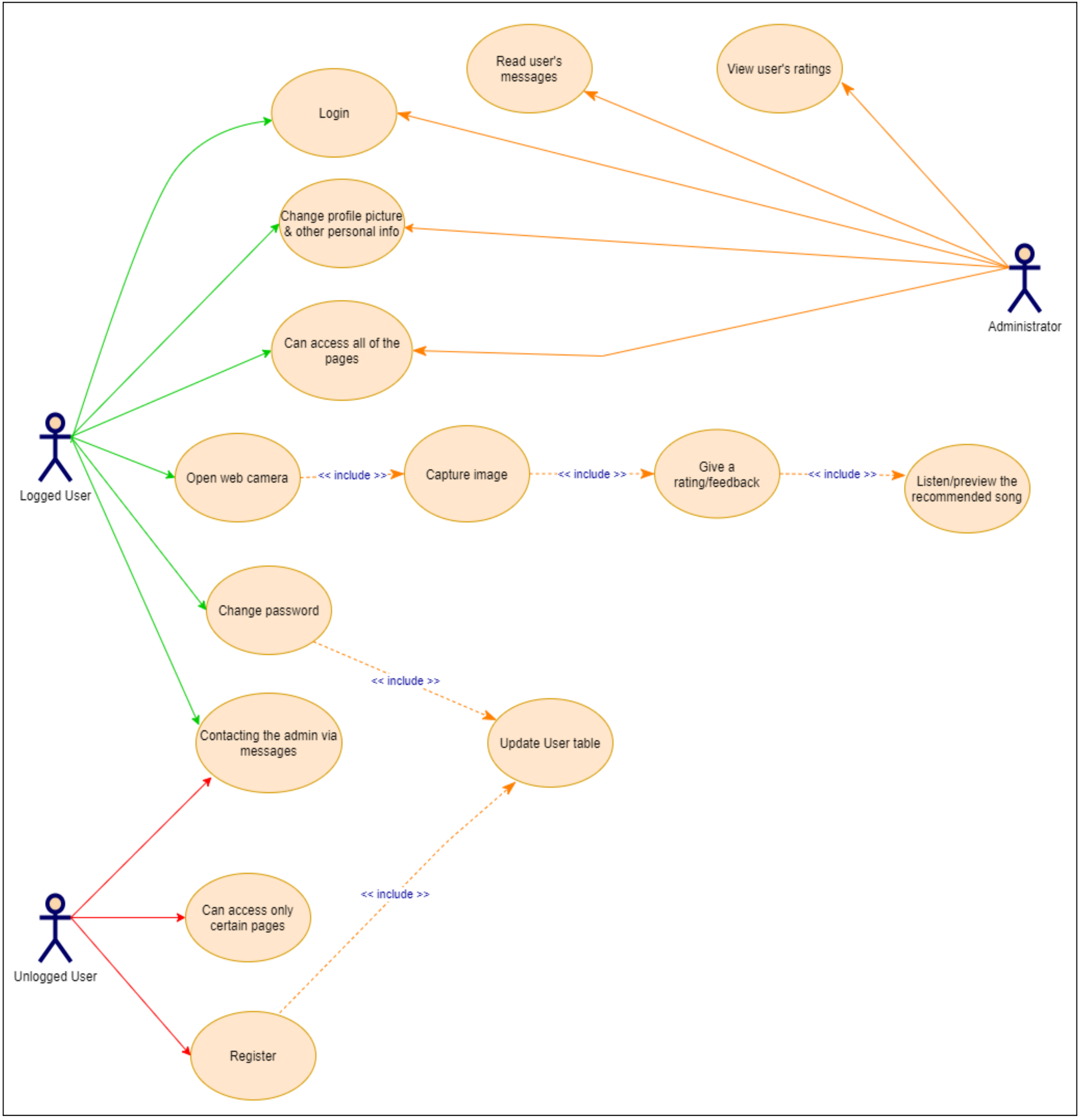


**Figure 1. Use Case Diagram**

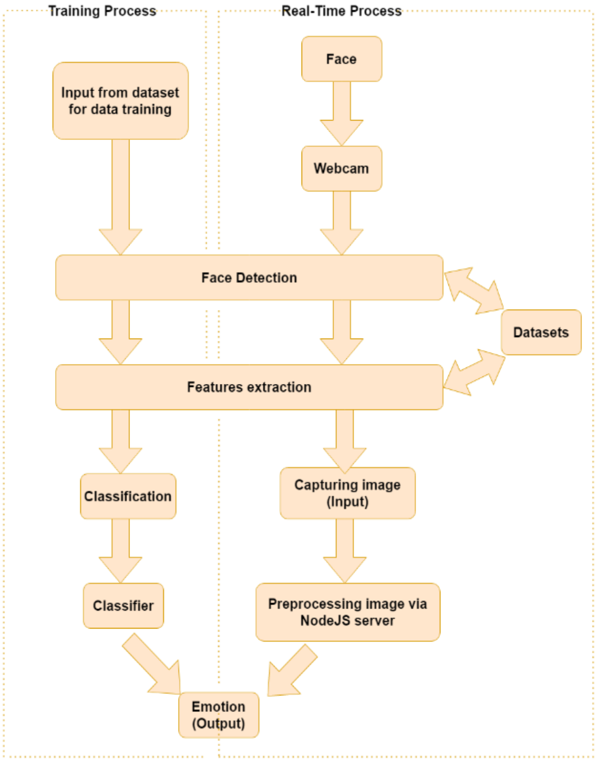
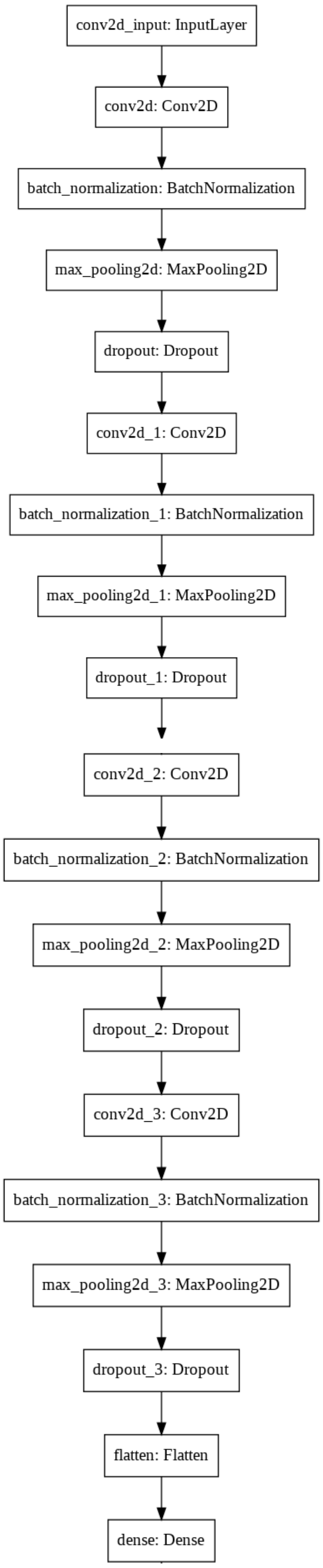


**Figure 2. Emotion Detection Diagram**

Regarding the emotion detection module, it is described in Figure 2, which captures both the training process and the real-time face detection process. Both processes are integrated into the same diagram because they share similar components. For the training process, everything starts from the input dataset and then for each epoch, the model is improved and validated going thru the face detection and features extraction phrase. For the real-time process, we have the model already trained, but the input comes from the webcam, it is processed using the JS module, and then an emotion label is outputted.

For replication purposes[1], Figure 3 represents the structure of the trained sequential model. The model starts from the input layer and goes down to the output layer; we have five Batch Normalization layers, three maxPooling2D layers and several dropouts and dense layers. This structure was used after exploring several configurations and validating each of them on both unseen data and real-life scenarios.

Regarding the music recommendation module and how the system works, we created Figure 4, which is an overview of the recommender system.


conv2d_input: InputLayer
conv2d: Conv2D
batch_normalization: BatchNormalization
max_pooling2d: MaxPooling2D
dropout: Dropout
conv2d_1: Conv2D
batch_normalization_1: BatchNormalization
max_pooling2d_1: MaxPooling2D
dropout_1: Dropout
conv2d_2: Conv2D
batch_normalization_2: BatchNormalization
max_pooling2d_2: MaxPooling2D
dropout_2: Dropout
conv2d_3: Conv2D
batch_normalization_3: BatchNormalization
max_pooling2d_3: MaxPooling2D
dropout_3: Dropout
flatten: Flatten
dense: Dense

[1] https://github.com/AndreiNegret/MusicBud

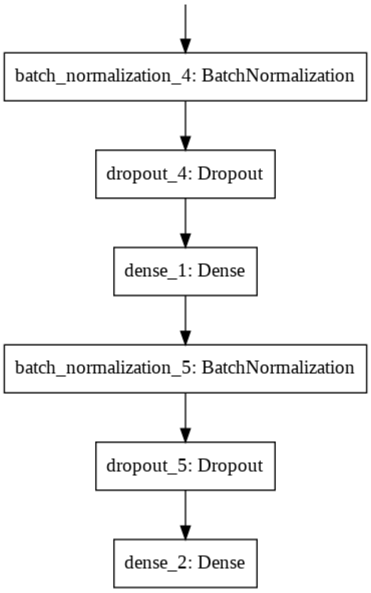

batch_normalization_4: BatchNormalization
dropout_4: Dropout
dense_1: Dense
batch_normalization_5: BatchNormalization
dropout_5: Dropout
dense_2: Dense

**Figure 3. Network structure**

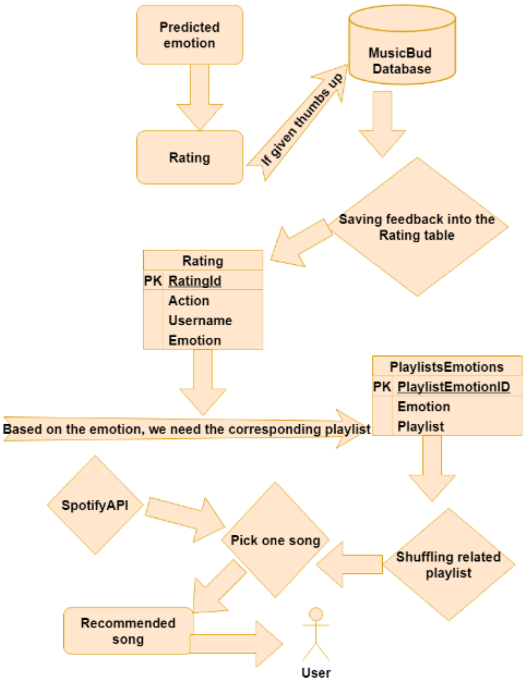
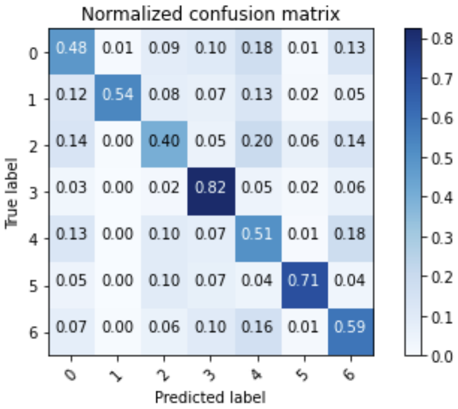


**Figure 4. Recommendation Overview**

Figure 4 explains the whole process built for the recommendation. It starts from the predicted emotion and is updated based on the rating; in the MusicBud database. Then the systems store the rating, and based on it; one playlist is selected. Going further, a shuffle is made on the songs obtained from Spotify API, and finally, a song is recommended to the user.

## EXPERIMENTS AND VALIDATION

Regarding the system validation, there are several ways to do it in order to get a good insight regarding the performance. Before going into presenting the app and how it was validated, we have to mention that the dataset was widely used, and even if the computed accuracy is not very good (around 0.6), in most of the cases, even if there is a small number of epochs, the results are pretty relevant. It is also worth mentioning that the number of instances for each emotion is imbalanced, and this will influence the confusion matrix, which is presented in Figure 4.



| | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| 0 | 0.48 | 0.01 | 0.09 | 0.10 | 0.18 | 0.01 | 0.13 |
| 1 | 0.12 | 0.54 | 0.08 | 0.07 | 0.13 | 0.02 | 0.05 |
| 2 | 0.14 | 0.00 | 0.40 | 0.05 | 0.20 | 0.06 | 0.14 |
| 3 | 0.03 | 0.00 | 0.02 | 0.82 | 0.05 | 0.02 | 0.06 |
| 4 | 0.13 | 0.00 | 0.10 | 0.07 | 0.51 | 0.01 | 0.18 |
| 5 | 0.05 | 0.00 | 0.10 | 0.07 | 0.04 | 0.71 | 0.04 |
| 6 | 0.07 | 0.00 | 0.06 | 0.10 | 0.16 | 0.01 | 0.59 |

**Figure 5. Normalized Confusion Matrix**

As we can see in Figure 5, which represents the normalized confusion matrix, because of the imbalanced classes, the accuracy for emotions is also different. For example, in some cases like Happy, we can obtain an accuracy of 0.82 or, in the case of Suprise, 0.71, but there are also other classes like fear which achieve only 0.4 accuracies. On Ox ad Oy labels of the confusion matrix, we have the emotions, which are coded as numbers, and their meaning is 0 for angry, 1 for disgust, 2 for fear, 3 for happy, 4 for sad, 5 for surprise and 6 for neutral.

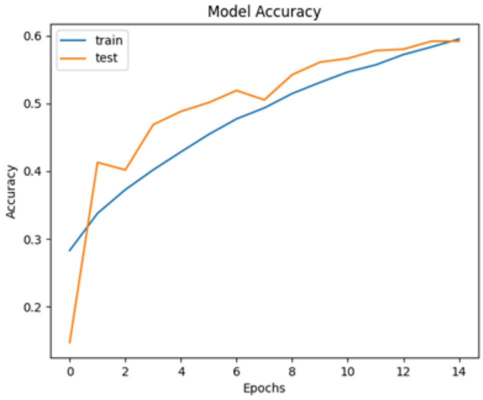
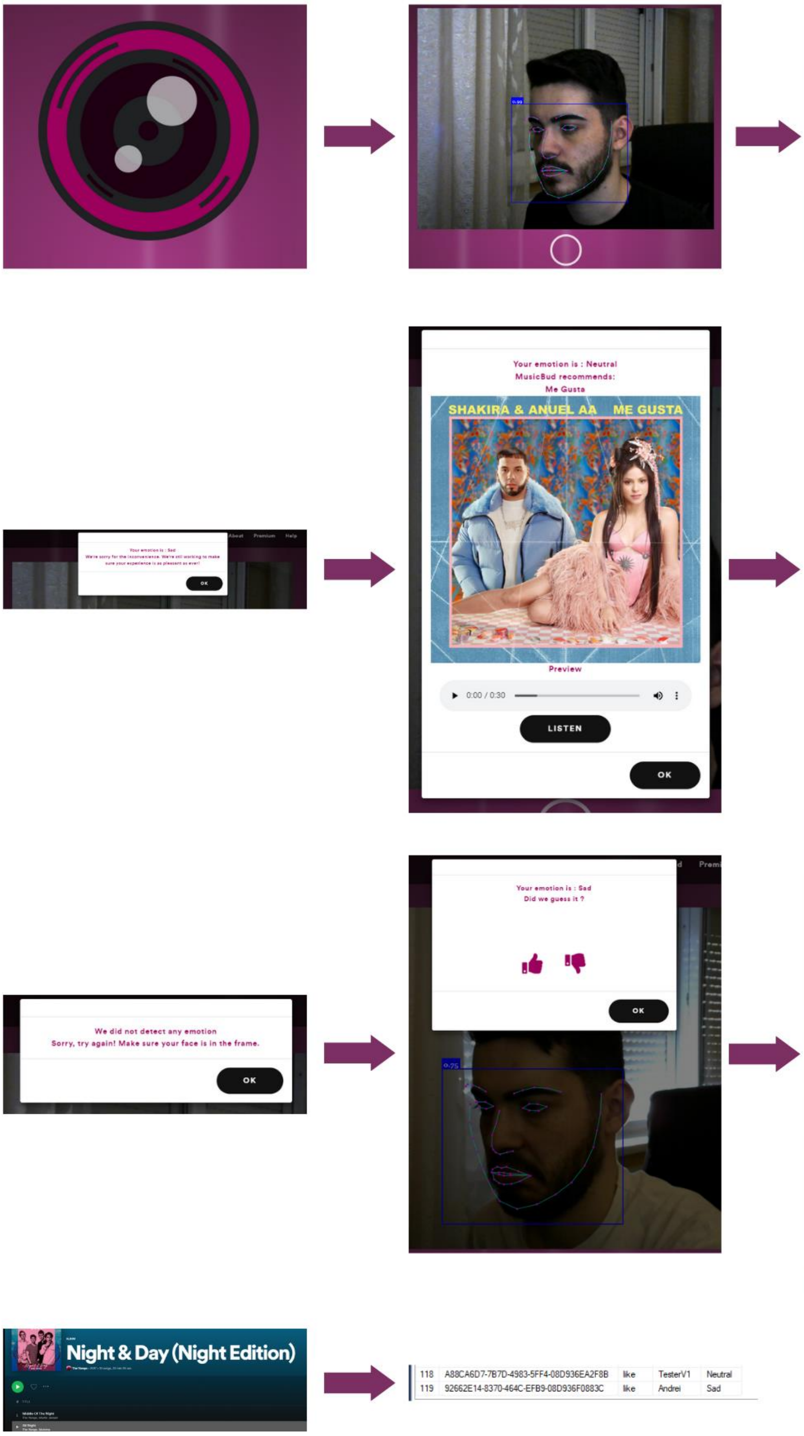


**Figure 6. Model's accuracy over 15 epochs**

Regarding the model evolution on training, we print the accuracy over 15 epochs in Figure 6, which is quite relevant for our case because the model achieves around 0.6

accuracies and tends to reach a plateau. In this case, we need to mention that we also tried to train the system on over 200 epochs, but we couldn't get relevant improvements. In Figure 6, the blue line represents the validation of the training data, and the orange line represents the validation of the test (unseen) data. Even if at the first epochs the accuracy tends to differ and on test data, the accuracy is not linear, over 15 epochs, both tend to reach the same values.


Your emotion is : Neutral
Did we guess it ?

OK

Your emotion is : Sad
We're sorry for the inconvenience. We're still working to make sure your experience is as pleasent as ever!

OK

Your emotion is : Neutral
MusicBud recommends:
Me Gusta

Preview

0:00 / 0:30

LISTEN

OK

We did not detect any emotion
Sorry, try again! Make sure your face is in the frame.

OK

Your emotion is : Sad
Did we guess it ?

OK

Your emotion is : Sad
MusicBud recommends:
All Night

Preview

0:00 / 0:00

LISTEN

OK

| | | | | |
|---|---|---|---|---|
| 118 | A88CA6D7-7B7D-4983-5FF4-08D936EA2F8B | like | TesterV1 | Neutral |
| 119 | 92662E14-8370-464C-EFB9-08D936F0883C | like | Andrei | Sad |

**Figure 7. System Demo**

Finally, for the desktop application, a system demonstration is presented in Figure 7, which takes the whole flow for getting a recommended song. Going from left to right for each row, we can see that first, we have to hit the button to take a snap, and then the face is detected along with a mood. After the mood detection, users can rate it and then the recommender system offers a song from the playlist waiting for the user's feedback. Depending on the user's choice, the system will start to improve the user's mood by playing the music or will offer another song.

## CONCLUSION

This paper presents a system for efficient music recommendations based on mood detection. Because of the type of application, during the system development, several challenges were faced, like building a robust and reliable mood detection system, building a lightweight system that does not consume a significant amount of resources and, of course, a user-friendly and easy-to-use system. Training and validating the deep learning module represented another challenge as the accuracy of the application is not having a fantastic value, but on our real-life tests, the system proved to be very reliable and trustable.

A future work, we plan to improve the recommender system using machine learning and deep learning techniques. At this moment, the playlists are already defined but having a generic method to classify new songs will easily keep the list of the songs up to date as songs tend to appear quite often, and a manual approach is a very time-consuming technique.

## REFERENCES

1. Meeki, N., Amine, A., Boudia, M.A. and Hamou, R.M., 2020. Deep learning for non-verbal sentiment analysis: Facial emotional expressions. GeCoDe Laboratory, Department of Computer Science, Tahar Moulay University of Saida.
2. Giannopoulos, P., Perikos, I. and Hatzilygeroudis, I., 2018. Deep learning approaches for facial emotion recognition: A case study on FER-2013. In Advances in hybridization of intelligent methods (pp. 1-16). Springer, Cham.
3. Medhat, W., Hassan, A. and Korashy, H., 2014. Sentiment analysis algorithms and applications: A survey. Ain Shams engineering journal, 5(4), pp.1093-1113.
4. Weiss, K., Khoshgoftaar, T.M. and Wang, D., 2016. A survey of transfer learning. Journal of Big data, 3(1), pp.1-40.
5. Gajarla, V. and Gupta, A., 2015. Emotion detection and sentiment analysis of images. Georgia Institute of Technology, pp.1-4.
6. Tsakalidis, A., Papadopoulos, S., Voskaki, R., Ioannidou, K., Boididou, C., Cristea, A.I., Liakata, M. and Kompatsiaris, Y., 2018. Building and evaluating resources for sentiment analysis in the Greek language. Language resources and evaluation, 52(4), pp.1021-1044.
7. Prabowo, R. and Thelwall, M., 2009. Sentiment analysis: A combined approach. Journal of Informetrics, 3(2), pp.143-157.
8. Agarwal, A., Xie, B., Vovsha, I., Rambow, O. and Passonneau, R.J., 2011, June. Sentiment analysis of twitter data. In Proceedings of the workshop on language in social media (LSM 2011) (pp. 30-38).
9. Yadav, Y., Kumar, V., Ranga, V. and Rawat, R.M., 2020, July. Analysis of Facial Sentiments: A deep-learning Way. In 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 541-545). IEEE.
10. Kaviya, P. and Arumugaprakash, T., 2020, June. Group Facial Emotion Analysis System Using Convolutional Neural Network. In 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184) (pp. 643-647). IEEE.
11. Jaiswal, S. and Nandi, G.C., 2020. Robust real-time emotion detection system using cnn architecture. Neural Computing and Applications, 32(15), pp.11253-11262.
12. Patel, K., Mehta, D., Mistry, C., Gupta, R., Tanwar, S., Kumar, N. and Alazab, M., 2020. Facial sentiment analysis using AI techniques: state-of-the-art, taxonomies, and challenges. IEEE Access, 8, pp.90495-90519.