Technische Universität Berlin
SS 2019

Music Mood Classification using Convolutional Neural Networks

Jan Stoltenberg (401445)
Duc Linh Tran (347498)
Simon Zoller (556606)

Advisors:
Athanasios Lykartsis,
Roman Gebhardt,
Dr. Charalampos Saitis,
Dr. Paul Luizard

August 17, 2019

Abstract

This paper presents a Convolutional Neural Network (CNN) for the use of emotion classification of music. Trained with the MediaEval 1000 songs Dataset, the network is based on a regression and outputs values for valence and arousal of each song sample. Mel-spectrograms, based on 15.0 audio-files excerpts, were used as input data. While the initial model of the network is based on the best results of prior research, this paper investigates how tweaking certain parameters and changing parts of the model's architecture impacts the overall performance.
Adapting the initial model with the respective best performing parameters, the final model and it's performance are conclusively presented.

Keywords : Music Emotion Recognition, Convolutional Neural Network, Mel-Spectrogram, Valence-Arousal Model

1 Introduction

In recent years, the use of music streaming services has greatly increased, making automatic classification and analysis of music a crucial subject area in the audio industry. Such classifications could e.g. automatically create playlists based on a certain mood or give mood-based recommendations.

Due to the subjective nature of the classification of emotions, this task poses great difficulty. However, research in recent years has shown growing success in the use of convolutional neural networks for classifying music and recurrent neural networks such as LSTMs (Long-Short-Term-Memory) [Malik et al., 2017] [Aljanaki et al., 2017]. Convolutional neural networks were originally developed for image classification tasks, but can also be used to process audio spectrograms (frequency content over time), and analogous to images are able to learn patterns in the music content.

As the mood of a song is interpreted differently by different listeners, it is hard to merge multiple opinions on a set of songs to a data set with ground truth. As emotions cannot be ordered, the emotions fall into the category of qualitative, nominal data types. Posner et. al. proposed circumplex model [Posner et al., 2005], which splits every emotion into two components: **valence**, which denotes the level of positivity or negativity, and **arousal**, which indicates the activeness of the emotion. While the model is still dependent on the subjective evaluation of the labeling users, the valence and arousal values are numbers, where calculations can be applied on. Since the valence/arousal values belong to the class of quantitative, continuous data types, it is possible to merge different opinions about a perceived emotions by calculating the mean of the valence and arousal values.

Therefore, in our implementation, instead of using classification, the authors chose to run a regression on the valence/arousal values. With the songs as input, we predict valence and arousal values, which then can be compared to the ground truth values by measuring the distance. Since we only evaluate the quantitized values, we are not bound to the opinion of a majority but rather, the resulting valence/arousal values are created through the influence of each labeler. For an end user, if a song is perceived in a specific emotion, another song with similar valence and arousal values will most likely be perceived in the same emotion, regardless of the perceived emotions of the labeler and other users.

Section 2 will give an overview of the dataset used, explain the applied pre-processing of the audio data and explain the model architecture. Moreover, the valence/arousal model, which was chosen as the dimensional psychometric model in this paper, is declared within this section. Section 3 displays the results of each change in the network, showing the overall residual mean average errors of the model when changing the input data as well as other parameters within the model. In section 4, some of the outcomes will be discussed and questioned. Finally, section 5 summarizes our work.

2 Methods

2.1 The Dataset

The dataset consists of 744 unique songs selected from Free Music Archive (FMA). For each song there is a 45.0 s excerpt with a randomly chosen starting point [Soleymani et al., 2013].
Each song got annotated 10 times by Amazon MTurk workers with a valence and arousal value ranging from 1 to 9 and each annotations was done separately for valence and arousal [Soleymani et al., 2013].
There exists a mean value and a standard deviation of those annotations for each song.
The mean value for each song is used as data for the network.

The mean value over all averaged song annotations is $\mu_a = 4.79$ with a standard deviation of $\rho_a = 1.38$ for arousal, and $\mu_v = 5.00$ and $\rho_v = 1.24$ for valence. This shows that the valence and arousal values are roughly centered around the center value 5.0.

It is to note that the mean value of all standard deviations of the averaged annotations are relatively high, they are $\mu_{\rho_a,i} = 1.64$ for arousal and $\mu_{\rho_v} = 1.56$ for valence.
This shows the relatively strong subjectivity of the annotations.

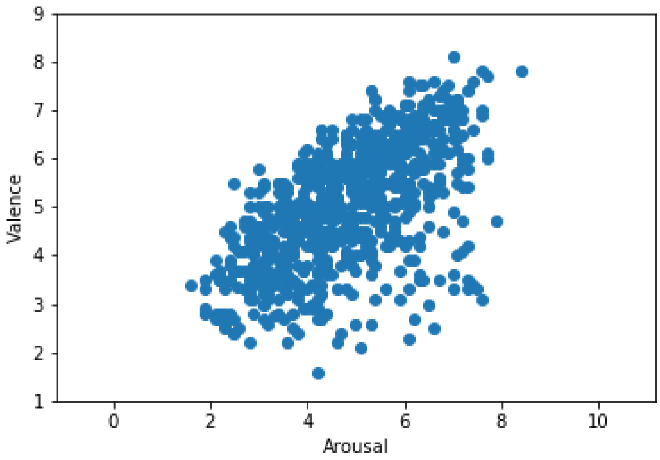

Figure 1: Distribution of mean values for valence and arousal in the dataset. The axis have equal dimensions. It is visible that the dataset is unequally distributed.

2.2 Pre-Processing the Data

For pre-processing, the Librosa python library was used [McFee et al., 2015], which contains additional music and audio analysis tools.

At first, the audio files are converted to mono and then resampled with a Sampling Frequency $f_s = 16000$ Hz to save computing time.
To avoid the association of loudness with emotion, all songs are normalized to a range from -1.0 to $+$ 1.0.
The songs are then converted to Mel-spectograms with the sample rate $f_s = 16000$ Hz and a number of 80 bins. The Power is converted to db.
Mel-Spectrograms in comparison to regular spectrograms have a logarithmic distribution of frequency bins and are therefore closer to our perception of music.

As for the valence/arousal, they are scaled to a range from 0.0 to 1.0, meaning that an arousal value of 7 equals 0.7. Throughout this paper, all values will be presented as scaled values.

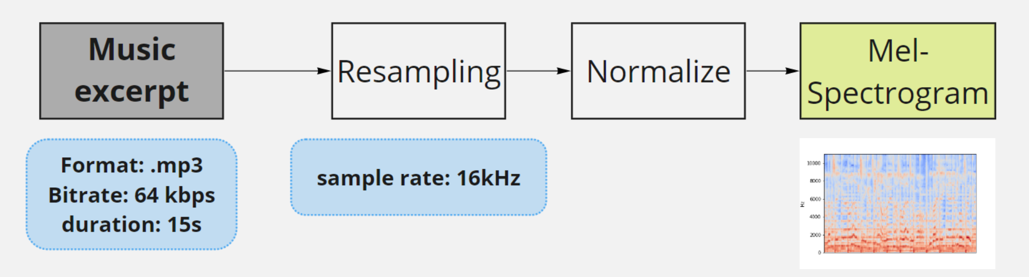

Figure 2: Pre-Processing Pipeline

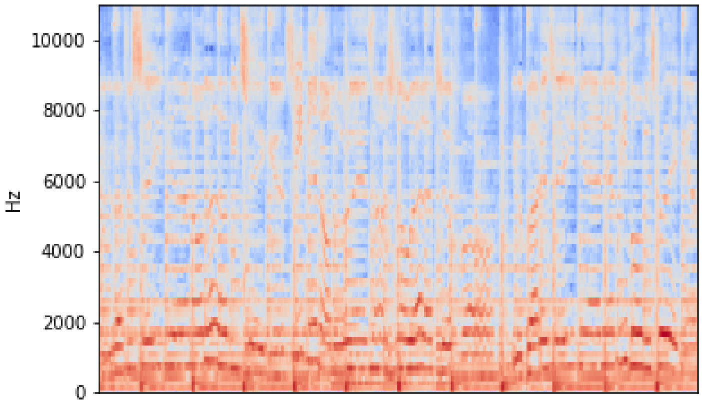

Figure 3: Mel-Spectrogram for Track ID Nr. 18 Black Rebel Motor - Beat The Devil's Tattoo (Live @ KEXP) over a range of 15.0 s.

2.3 Basic Network architecture

The network architecture was chosen based on research on similar datasets, such as [Malik et al., 2017]. Due to the relatively small dataset, it seems advisable to use a relatively small and simple network.
The network consists of a convolutional layer with varying filter and kernel sizes. It is followed by an activation function and then goes into a pooling layer.
After flattening there is a fully connected layer with an activation function, batch normalization and a dropout layer. At last, there are the two output neurons for valence and arousal with an activation function.

E.g., [Malik et al., 2017] reports good results on a similar dataset with one convolutional layer followed by some fully connected and recurrent layers.

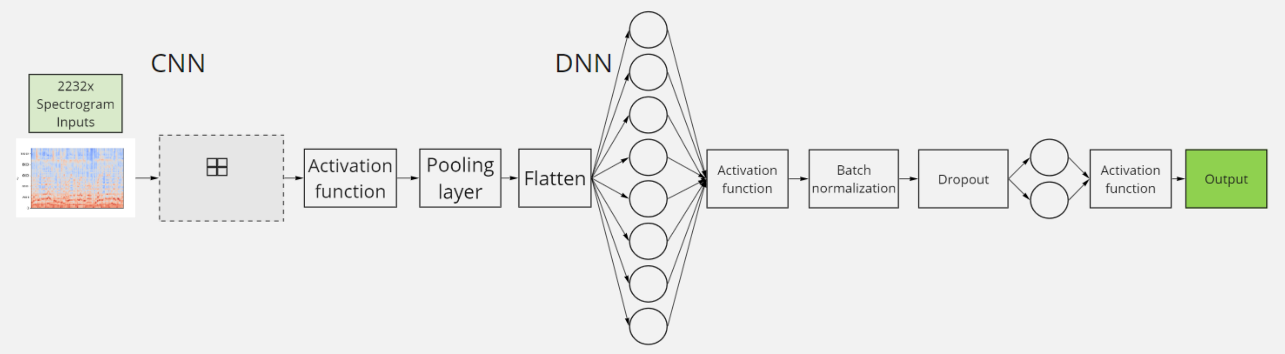

Figure 4: Schematic of the used network architecture. Spectrograms are used as input into a convolutional layer. The output are two values, valence and arousal.

2.4 Dimensional vs. categorical psychometrics - Valence/Arousal model

While emotions are labeled discretely in common parlance, different strategies have been developed to classify emotions using neural networks [Aljanaki et al., 2017]. While one approach would be to design a classification network that outputs one of a determined number of discrete emotions, a different approach is to use the two-dimensional emotion model proposed by [Posner et al., 2005], which suggests that emotions can be displayed on a two-dimensional valence/arousal scale, forming a circle within the proposed coordinate system.
Latter has been chosen as the approach for this model, as [Yang and Chen, 2012] refer to the two-dimensional regression approach as a "[...] sound theoretical foundation [that] exhibits promising prediction accuracy".

2.5 Optimizing the parameters of the network

In each subsection of section 3, the results of the respective different parameters were changed and the mean absolute error was evaluated.
For each changed value the network was trained for 10 epochs 10 times. The mean value and the standard deviation for those 10 mean average errors is calculated and then compared. In the end the best resulting network is chosen.
10 % of the original dataset was used as validation dataset. Those songs were randomly selected.

Each category start from the same base model 5 and changes only the respective parameters.
Based on this first prototype, different parameters have been tweaked to get the best accuracy.

2.5.1 CNN kernel size

In our model, instead of the raw audio data, the spectogram is used as a feature to take advantage of image processing techniques. When going through each pixel in the image, not only the current pixel is processed, but also surrounding pixel. The number of surrounding pixels is determined by the kernel size.
The kernel size defines the width of the 2D convolution window used for convolution. In order to find the best overall error for different kernel sizes, the different kernel sizes *2x2, 3x3, 5x5 and 10x10* were tested based on the above-mentioned prototype network. Since the network's biases and weights and the test and validation and test set distributions are set randomly each time the network is initialized (e.g. with new kernel sizes), the network has been initialized with these four different kernel sizes multiple times each, and the mean squared error was taken afterwards to ensure a more generalized comparison.

2.5.2 Optimizer functions

The CNN uses backpropagation, so after each epoch, the result will be transformed into weights for the new iteration. The weights are calculated through the partial derivation of the error function.
For our experiments, different optimization algorithms are used.

The stochastic gradient descent optimizer (**SGD**) an iterative algorithm using randomly selected samples to evaluate the gradient.

RMSProp [Ruder, 2016] is a popular unpublished algorithm, which was introduced for adaptive machine learning applications. It was introduced to handle mini batches for on-line types of applications and is useful in cases, where the learning rate drops fast. It operates on pre-scaled gradients.

ADADELTA was introduced as a "per-dimension learning rate for gradient descent" [Zeiler, 2012], which means that for each dimension or element position in a vector, a separate learning rate is applied to take advantage of differences. With the minimal computational overhead, it is considered a simple SGD extension.

Adam is described by its authors as "an algorithm for first-order gradient-based optimization of stochastic objective functions" [Kingma and Ba, 2014]. It is closely related to to RMSProp. By requiring a stepsize, exponential decay rates for the moment estimates, a stochastic objective function with parameters and an initial parameter vector, the Adam algorithm returns by updating moving averages a resulting parameter vector to solve an optimization problem. The moving averages, which are estimates of the mean and the uncentered variance of the gradient, are controlled by the decay rates.

AdaMax is a variation of Adam, which uses the infinity norm. Another related work to Adam is **ADAGRAD** [Duchi et al., 2011], which corresponds to Adam for specific decay rates. By extending Adam with Nesterov's accelerated gradient (NAG), the goal of **Nadam** [Dozat, 2016] lies in incorporating the advantages of NAG to the Adam algorithm.

2.5.3 Number of filters

When going through an image to analyze the content, it is useful to find different aspects of the image, from the detection of hard edges or finding a color map to abstract results, which might not make sense to the user. With the number of filters, we can set the number of different kernels, which will be applied on the image. A higher number of filter means more different types of probes in the image.

3 Results

This section shows the results of the proposed model with changes for different parameters.
To achieve the best comparability, a base model has been selected for this paper, that is shown in figure 5. While the following subsections display the change of accuracy for tweaking only one parameter at each time, the other parameters will always adopt to the base model's values.

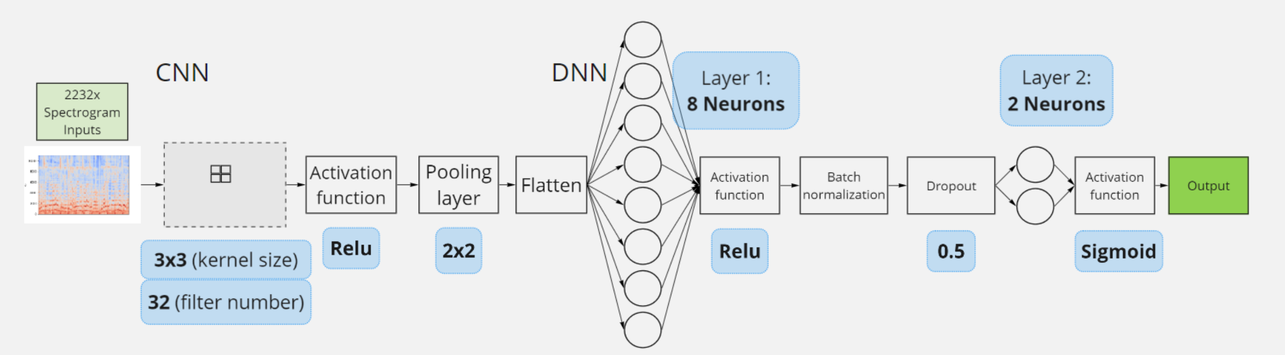

Figure 5: Base model used for optimizing network parameters.

At the end of this section, the authors will present the overall accuracy of the model with each parameter set to the value with the respective highest accuracy.

3.1 Original Data vs augmented Data

As described in the Methods section, the dataset's size has been tripled by augmenting existing data. This subsection briefly takes a look at the different accuracies and standard deviations of the standard model with different kernel sizes for some variation and better comparability. Both datasets, the original one and the extended one were used to see if the additional data make a difference in the outcome.

Table 1 shows the average mean absolute error and SD of the regression model for the old dataset with 1000 songs.
Table 2 shows the average mean absolute error and SD of the regression model for the new dataset with 3000 songs.

2x2	3x3	5x5	10x10
0.097	0.100	0.091	0.1300

Table 1: Original dataset, Rectangular kernel sizes with respective average mean absolute error *(SD = 0.004, 0.0234, 0.0125, 0.036)*.

kernel size	2x2	3x3	5x5	10x10
mean absolute error	0.097	0.095	0.096	0.096
standard deviation	0.007	0.007	0.009	0.008

Table 2: Augmented dataset, Rectangular kernel sizes with respective average mean absolute error and standard deviation

3.2 Kernel size

As already shown in section 3.1, table 2 shows that a kernel size of *3x3* produces the lowest error. However, the errors for the other kernel sizes are very close to this value.

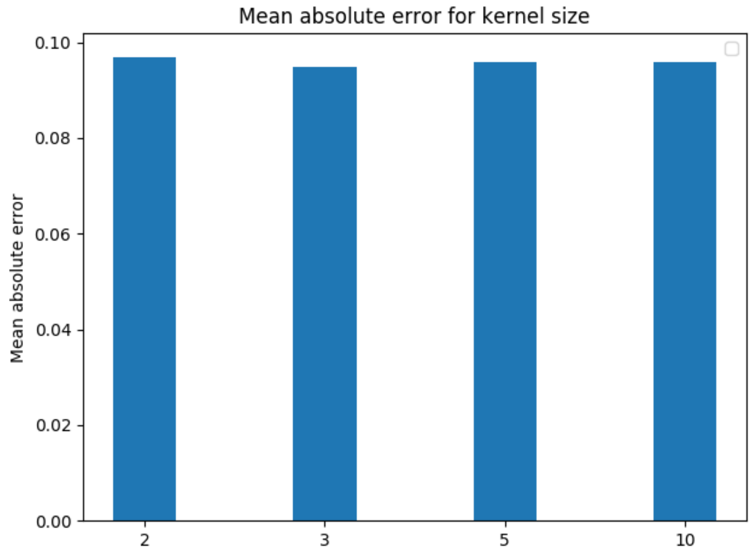

Figure 6: Average mean absolute error in regards to the kernel sizes on the augmented dataset. While the values are very similar, the minimum is at the kernel size of 3x3, which is the most commonly used kernel size.

On another note, the SD of the MAE is lower for the kernel sizes 3x3, 5x5 and 10x10.

3.3 Optimizer function

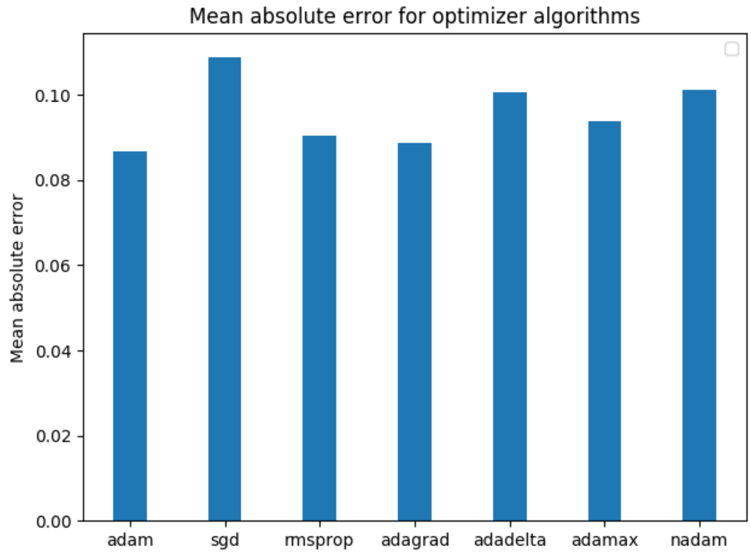

Figure 7: Average mean absolute error for different optimizer algorithms on the augmented dataset. The x-axis shows the different optimizers.

Another experiment was executed with different optimizer algorithm. The experiments did not have a great difference in performance in regards to the optimizer algorithms.

optimization algorithm	adam	sgd	rmsprop	adagrad	adadelta	adamax	nadam
mean absolute error	0.087	0.109	0.091	0.089	0.101	0.094	0.101
standard deviation	0.006	0.023	0.009	0.008	0.011	0.008	0.012

Table 3: Optimizer algorithms with respective average mean absolute error and the standard deviation.

The result is shown in figure 7 and table 3. The figure shows that Adam performs best with an average mean absolute error of 0.0869. The stochastic gradient descent performed worst with an average mean absolute error of 0.1089. Given that SGD is the most basic optimization algorithm, the outcome is reasonable. It is also interesting to note that the standard deviation for SGD is very big in comparison to all other experiments. This can be attributed to the random sampling without any additional tweaking.

3.4 Filter size

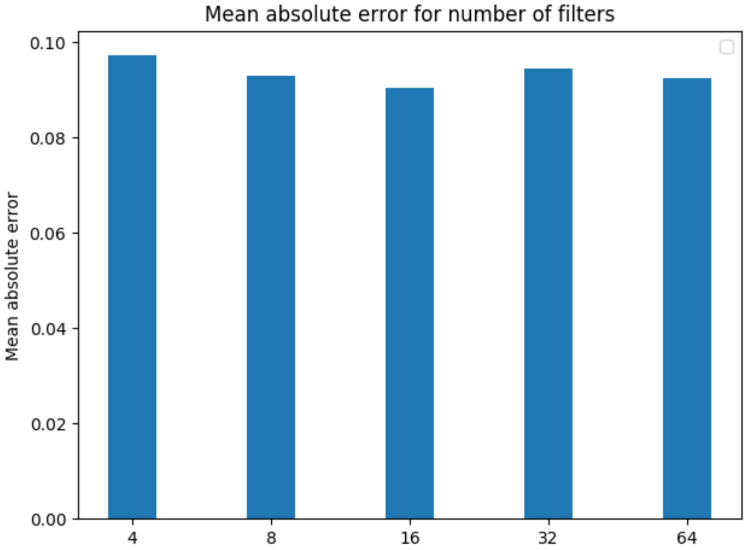

Figure 8: Average mean absolute error in regards to the number of filters on the augmented dataset. The minimum error is achieved with 16 filters.

number of filters	4	8	16	32	64
mean absolute error	0.097	0.093	0.090	0.095	0.092
standard deviation	0.008	0.009	0.006	0.010	0.008

Table 4: Number of filters with respective average mean absolute error and standard deviation.

Figure 8 shows that the model performs best with 16 filters. It also has the smallest standard deviation based on table 4, which means that it provides the most stable results as well.
If processing time must be considered in the cost evaluation, choosing 8 filters is also reasonable, although the error is lower for 16 and 64 filters.

3.5 Activation function

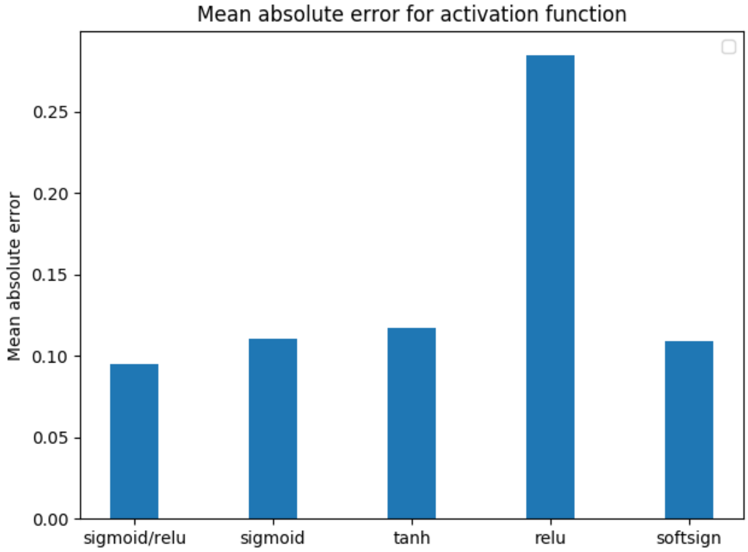

Figure 9: Average mean absolute error for different activation functions on the augmented dataset. For the most left entry, relu was used for the first 2 layers and sigmoid for the last one.

Table 5's columns 2-5 show the average mean absolute error for having the respective same three activation functions within the network.
Based on the research described in chapter 2.3, previous networks have been designed with a combination of activation functions. Therefore, column one is based on such a combination of the relu activation function after the CNN and after the first DNN layer with 8 neurons. A sigmoid activation function for the last layer with two neurons was chosen.

It's evident that indeed the combination of the two different activation functions achieves the best results. It therefore makes sense to try out different combinations of activation functions within one network. On another note, the model with only relu activation functions performed worst with a average mean absolute error of 0.285.

activation function	relu/sigmoid	sigmoid	tanh	relu	softsign
mean absolute error	0.095	0.110	0.117	0.285	0.109
standard deviation	0.010	0.008	0.009	0.006	0.010

Table 5: Activation functions with respective average mean absolute error and standard deviation.

3.6 Dropouts

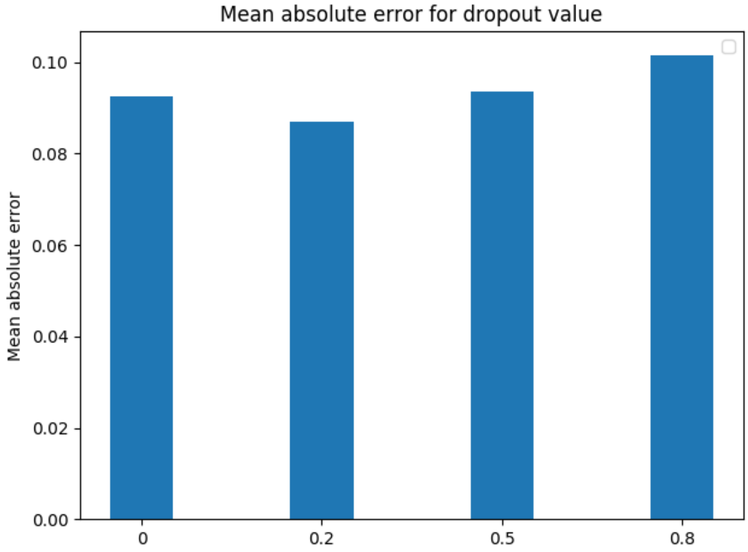

Figure 10: Average mean absolute error in regards to the dropout value on the augmented dataset.

The mea presented for each dropout value presented in table 6 shows a minimum mae of *0.087* for a dropout value of 0.2.

dropout	0	0.2	0.5	0.8
average mean absolute error	0.092	0.087	0.094	0.102
standard deviation	0.013	0.008	0.007	0.004

Table 6: Dropout values with respective average mean absolute error and standard deviation

3.7 Max pooling

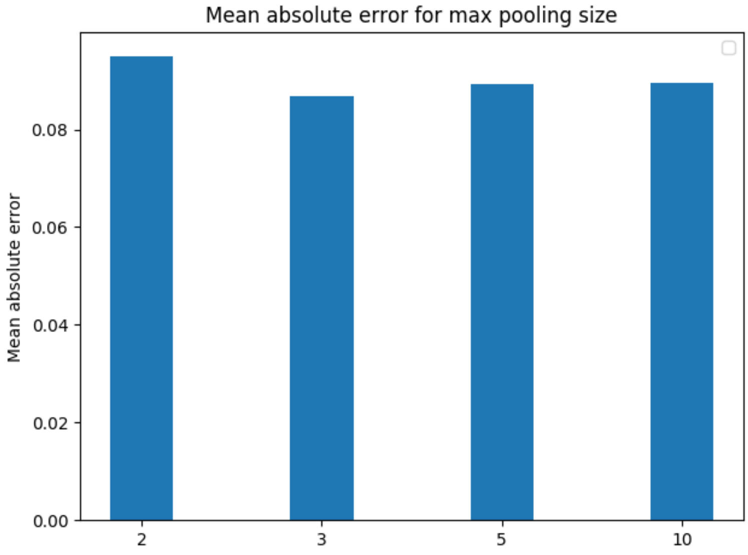

Figure 11: Average mean absolute error in regards of max pooling values on the augmented dataset.

As previously mentioned, max pooling has been applied to the model once within the CNN of the presented neural network. Table 6 shows, that the amount of max pooling doesn't strongly influence the overall mae of the model. However, the minimum mae values are observable for a max pooling size of 3x3.
As with the previous parameters, this value has been adopted for the final model proposed in this paper.

max pooling size	2x2	3x3	5x5	10x10
mean absolute error	0.095	0.087	0.089	0.089
standard deviation	0.008	0.005	0.007	0.011

Table 7: Max pooling sizes with respective average mean absolute error and standard deviation.

3.8 Proposed model with respective optimum values

Figure 12 shows the final outcome of this paper's investigation - Adam was chosen as the optimization function.

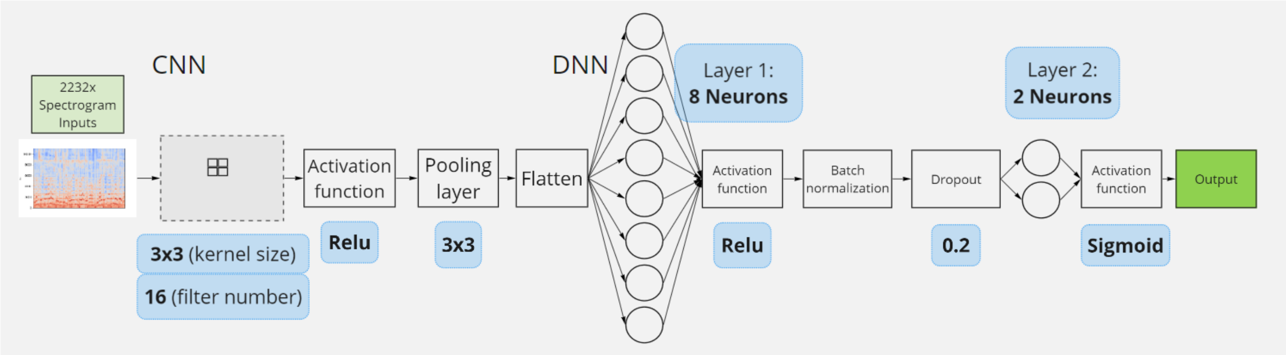

Figure 12: Schematic of the proposed network architecture with adjusted values.

On the validation dataset, a mean absolute error of mae $= 0.089 \pm 0.01$ was reached.
This means on our range of values from 0.1 to 0.9 that we have a relative error of $\frac{0.089}{0.8} \cdot 100 = 11.125\%$.

4 Discussion

4.1 Comparability

As evident in the review paper [Kim et al., 2010], these results are comparable to other results of past networks concerned with the regression-based models. [Schmidt et al., 2010], who reached used SVRs and Multiple Linear Regression, obtained an average error distance in the normalized valence/arousal space of only 13.7% with their highest performing system. Comparing these results, the model proposed in this paper performs 2,6% better than [Schmidt et al., 2010].
It is to be noted that other models, based on different approaches, combining multiple techniques and using different and richer datasets, achieved better performance [Kim et al., 2010].

4.2 On the valence/arousal model and the dataset

It is to argue if the valence and arousal model is a useful model to describe emotions and especially if it is useful to create a meaningful relation between songs on an emotional basis.
As already mentioned in 2.1, the dataset already shows a great deviation in the task of labelling tracks. The mean values for all standard deviations of the averaged values are relatively high compared to the range of possible values (1 to 9). Also, figure 1 shows that there is an unequal distribution of songs. Almost no songs with a low arousal and a high valence exist, and only a few with high arousal and low valence.

Furthermore, it is difficult to give certain emotions such as anger or joy a specific value or value-range inside the valence/arousal space. And even though emotions might be close to each other within the valence/arousal space, it doesn't mean that one perceives them as similarly related to each other.
For a specific task like music recommendations on emotion it might be argued that a dataset with values (e.g. ranging from 0.0 to 1.0) for each emotion (e.g. anger, joy, sadness, etc) might give a better representation of the underlying emotional perception.
The great power in the valence/arousal model though, lies in its simplicity and it might still be able to give useful relations between songs on an emotional basis. Additionally, the valence/arousal model offers flexibility concerning the final classification. Subsequent classification such as classifying the emotions in the coordinate system's quadrants or assigning them to an arbitrary number of discrete emotions can be done afterwards relatively easily. E.g., the smallest distance to a discrete emotion in the valence/arousal space can lead to discrete, specific emotion classification or simply the positive/negative position to the coordinate origin on each axis can allows for quadrant mapping.

4.3 Changes of parameters

For the number of filters, we achieved our best result with a low number of 16 filters in comparison with 32 or 64 filter. With these experiment results, we conclude that our spectrogram does not have many different characteristics to determine the valence/arousal values. We can predict that a higher number of filters will not improve our results or even causing overfitting.
While the Adam algorithm provided the best result in our experiment, RMSProp and AdaGrad also achieved relatively good results. Since Adam combines the advantages of RMSProp's on-line setting and AdaGrad sparse gradient handling, it can be assumed that our dataset might have both properties. We cannot confirm or prove these properties though. It is clear that much additional work will be required before a complete understanding of this phenomenon occurs.

Mel-frequency coefficients are adapted to human hearing, modeling the spacing of frequency bins more closely to musical perception [Hasan et al., 2004]. Since spectrograms are limited in size and resolution due to their discrete nature, the MFCCs are maximizing the contained information in relation of our hearing. This makes sense, as emotion classification is purely based on our cognitive, human assessment. Many other researchers have found MFCCs to work best in their studies too [Kim et al., 2010], stressing their usefulness even more.

Section 3 further showed the lowest mae at a dropout rate of 0.2. It seems reasonable that relatively large dropout rates close to 1.0, e.g. one of 0.8 (cf. table 6), will increase the mae at some point, as only very little or even no data are forwarded. It should also be mentioned that dropout rates tend to improve generalization of the data and, as dropouts should also be considered to prevent overfitting, further studies could investigate the relationship of dropouts and overfitting for different models more closely.

4.4 Augmented vs original data

Augmenting the dataset by taking 3 times 15.0 seconds excerpts didn't make a significant difference.
As a song's basic chord structure often doesn't dramatically change within the song, the added value only minimally affects (cf. table 12). Having completely different songs would most probably have a greater impact on the model's output than a simple data augmentation.

4.5 Less standard deviation

One of the great difficulties in training the network lies in the fact that all valence/arousal values are centered around 5.0. The network in many cases makes relatively conservative estimations closer to 5.0 than the original dataset's sample distribution.
There is a standard deviation for all arousal values of $\rho_a = 1.38$ and for valence $\rho_v = 1.24$ in the dataset. This paper's proposed model has a smaller one of $\rho_a = 0.8$ and $\rho_v = 0.7$. This phenomenon might occur due to the mathematical approach of the network to minimize the mae. Minimizing the overall mae might thus happen at the cost of outliers being very improbable as an output of the network. In other words, the network might not output solid predictions about songs featuring more extreme valence/arousal values.
On this occasion, future work could investigate whether a subsequent quadrant classification could be advantageous for this specific problem, since as much as a minimum deviation from the coordinate origin in both dimensions strongly impacts the final result and extreme values in a respective dimension wouldn't affect that result.

5 Conclusion

Throughout this paper, a Convolutional Neural Network was implemented, which is able to predict valence/arousal values for song excerpts for Music Mood Classification. Several experiments were carried out in which different parameters were optimized. The best results features a mean absolute error below 0.1, meaning that the predicted average valence/arousal values are only 1 value off when being compared to the ground truth. The proposed parameters, determined through experiments stated in this paper, performed better than the initial model, which was inspired by other papers.
For future work, a bigger data set with a representable distribution is needed to confirm whether the estimation around 0.5 is only a problem in this investigation's experimental setup or a more general problem in this field of research. A dataset with music excerpts featuring more extreme valence/arousal values could be an interesting starting point for such an investigation.

References

[Aljanaki et al., 2017] Aljanaki, A., Yang, Y.-H., and Soleymani, M. (2017). Developing a benchmark for emotional analysis of music. *PLOS ONE*, 12(3):e0173392.

[Dozat, 2016] Dozat, T. (2016). INCORPORATING NESTEROV MOMENTUM INTO ADAM. page 4.

[Duchi et al., 2011] Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. page 39.

[Hasan et al., 2004] Hasan, R., Jamil, M., and Rahman, G. R. S. (2004). SPEAKER IDENTIFICATION USING MEL FREQUENCY CEPSTRAL COEFFICIENTS. page 5.

[Kim et al., 2010] Kim, Y. E., Schmidt, E. M., Migneco, R., Morton, B. G., Richardson, P., Scott, J., Speck, J. A., and Turnbull, D. (2010). MUSIC EMOTION RECOGNITION: A STATE OF THE ART REVIEW. page 12.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*. arXiv: 1412.6980.

[Malik et al., 2017] Malik, M., Adavanne, S., Drossos, K., Virtanen, T., Ticha, D., and Jarina, R. (2017). STACKED CONVOLUTIONAL AND RECURRENT NEURAL NETWORKS FOR MUSIC EMOTION RECOGNITION. page 6.

[McFee et al., 2015] McFee, B., Raffel, C., Liang, D., Ellis, D., McVicar, M., Battenberg, E., and Nieto, O. (2015). librosa: Audio and Music Signal Analysis in Python. pages 18–24, Austin, Texas.

[Posner et al., 2005] Posner, J., Russell, J. A., and Peterson, B. S. (2005). The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology. *Development and Psychopathology*, 17(03).

[Ruder, 2016] Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv:1609.04747 [cs]*. arXiv: 1609.04747.

[Schmidt et al., 2010] Schmidt, E. M., Turnbull, D., and Kim, Y. E. (2010). Feature Selection for Content-based, Time-varying Musical Emotion Regression. In *Proceedings of the International Conference on Multimedia Information Retrieval*, MIR '10, pages 267–274, New York, NY, USA. ACM. event-place: Philadelphia, Pennsylvania, USA.

[Soleymani et al., 2013] Soleymani, M., Caro, M. N., Schmidt, E. M., Sha, C.-Y., and Yang, Y.-H. (2013). 1000 songs for emotional analysis of music. In *Proceedings of the 2nd ACM international workshop on Crowdsourcing for multimedia - CrowdMM '13*, pages 1–6, Barcelona, Spain. ACM Press.

[Yang and Chen, 2012] Yang, Y.-H. and Chen, H. H. (2012). Machine Recognition of Music Emotion: A Review. *ACM Transactions on Intelligent Systems and Technology*, 3(3):1–30.

[Zeiler, 2012] Zeiler, M. D. (2012). ADADELTA: An Adaptive Learning Rate Method. *arXiv:1212.5701 [cs]*. arXiv: 1212.5701.