Beyond Happy or Sad: Music Emotion Recognition with Multi-Label Classification

Saeedeh Javadi
s301405@studenti.polito.it

Faezeh Saeedian
s301308@studenti.polito.it

Vida Ahmadi
s301905@studenti.polito.it

Xiyang Zu
s288740@studenti.polito.it

Reza Barati
s301309@studenti.polito.it

ABSTRACT

Recent progresses in deep learning sped up the development of content-based automatic music tagging systems. Music information retrieval (MIR) researchers advance a proposal for various architecture designs, mainly based on convolutional neural networks (CNNs), that achieve state-of-the-art results in this multi-label binary classification task. However, due to the differences in experimental setups followed by researchers, such as using different dataset splits and software versions for evaluation, it is difficult to compare the proposed architectures directly with each other. To facilitate further research, We present a music emotion recognition algorithm using Vggish and mel-spectrogram representations which then are classified using a shallow CNN architecture and a transformer on MTG-Jamendo dataset, and provide reference results using common evaluation metrics (ROC-AUC). For reproducibility, we provide the Keras implementations with the pre-trained models.

KEYWORDS

Music emotion recognition, CNN, Transformer, MTG-Jamendo dataset

1 INTRODUCTION

James Russell classified human emotions in 1980, using a model called the circumplex model as shown in Figure 1 [18]. This model represents emotions in a two-dimensional circular space that consists of the arousal and valence dimensions. Emotional states can be positioned at any point within this space, including different levels of valence and arousal, or a neutral level for one or both of these factors. In the case of music, this two-dimensional scale can also be used to classify songs emotionally. With the explosion of vast and easily-accessible digital music resources over the past decade, the retrieval of music by emotion is becoming an important task for various applications, such as song selection in mobile devices, music recommendation systems, TV and radio programs, and music therapy[19].

In order to achieve emotion-based music retrieval, it is often necessary to label the emotion of a musical work, which is not only a huge workload but also inefficient if the exploding volume of digital music is labeled manually. For this reason, it is important to study music emotion recognition (MER) techniques to achieve automatic emotion recognition of music works. Among them, MER aims to automatically recognize the effective content of a piece of music,

Figure 1: Human-Emotion diagram by James Russel.

which is the key component of the above-mentioned music emotion applications[14]. MER constitutes a process of using computers to extract and analyze music features, form the mapping relations between music features and emotion space, and recognize the emotion that music expresses[15]. Music features are often extracted from the audio signal, symbolic music scores, lyrics texts, and even biological features like EEG (Electroencephalogram). Emotion space can be represented by a finite number of discrete categories or an infinite number of points in a continuous multidimensional space[11].

Historically, due to limitations in classification techniques and computing power, MER was basically modeled as a single-label or multi-class classification task, using machine learning classifiers or deep learning models to produce results. However, the diversity of music leads to the fact that there may be many different kinds of emotions contained in the same piece of music. Therefore, this paper will focus on the multi-label classification problem for the emotion recognition of audio signals.

The major contribution of this paper will be presented in two parts:

- Extracting audio representations from two different approaches.
- A comparison between different multi-label classification techniques.

The paper is organized as follows. Section 2 related works on the MER field. Section 3 elaborates on the methodology and theory. We report model performances and their generalization capabilities,

,
,

and overall experimental results in Section 4. Finally, Section 5 concludes the paper.

2 RELATED WORKS

In this section we will provide an overview of two main facets:

- music emotion recognition.
- multi-label classification.

Hizlisoy et al. proposed an approach for MER based on convolutional long short-term memory deep neural network (CLDNN) architecture. They utilized features obtained by feeding convolutional neural network (CNN) layers with log-mel filter bank energies and mel frequency cepstral coefficients (MFCCs) in addition to standard acoustic features[20]. Wu et al. exploited a Hierarchical Music Emotion Recognition model (HMER) - a novel hierarchical Bayesian model sing sentence-level music and lyrics features. It captures music's emotion dynamics with a song-segment-sentence hierarchical structure. HMER also considers emotion correlations between both music segments and sentences[14].

Liu et al. used a deep convolutional neural network (CNN) on the music spectrograms that contain both the original time and frequency domain information[9]. Chen et al. applied CNN-LSTM (convolutional neural networks-long short-term memory) combined network in the field of music emotion classification and proposes a multifeatured combined network classifier based on CNN-LSTM which combines 2D (two-dimensional) feature input through CNN-LSTM and 1D (single-dimensional) feature input through DNN (deep neural networks) to make up for the deficiencies of original single feature models. The model used multiple convolution kernels in CNN for 2D feature extraction, and BiLSTM (bidirectional LSTM) for serialization processing and was used, respectively, for audio and lyrics single-modal emotion classification output[16].

Mehmet Bilal Er et al. presented a method for MER by using the AlexNet deep learning architecture as the pre-trained network model with the chroma spectrograms extracted from music recordings. They used the extracted deep features to train and test the Support Vector Machines (SVM) and the Softmax classifiers. They compared different layers of the VGG-16 deep network model in order to figure out the effective power of pre-trained deep networks in MER. The best result was obtained from the VGG-16 in Fc7 layer as 89.2%[1].

In Multi-label classification domain, much research has been done. Li et al. leveraged l2,1-norm and l1-norm regularizers to learn common and label-specific features simultaneously. The authors used a regularizer to constrain label correlations on label outputs instead of the coefficient matrix. Finally, they considered the k-nearest neighbor mechanism with the instance correlations[12]. Liu et al. announced an approach that utilizes Transformer decoders to query the existence of a class label. The built-in cross-attention module in the Transformer decoder offers an effective way to use label embeddings as queries to probe and pool class-related features from a feature map computed by a vision backbone for subsequent binary classifications. The approach outperformed five multi-label classification data sets, including MS-COCO, PASCAL VOC, NUS-WIDE, and Visual Genome[13].

Wang et al. focused on a label graph superimposing framework to improve the conventional GCN+CNN framework developed for multi-label recognition[10]. Yu et al. proposed a wrapped learning approach, they learned one kernelized linear model for each label where label-specific features were simultaneously generated within an embedded feature space via empirical loss minimization and pairwise label correlation regularization[21]. Compared to the studies mentioned above, this paper focuses more on large datasets with more than 50 labels and 18,000 thousand music tracks, while comparing the performance of various algorithms on multi-label datasets.

3 METHODOLOGY

Based on the medium size of the dataset selected, feature-based machine learning and shallow classifiers were employed. In this section, we explain the extracted features and the methodology we followed to train robust classification models. We analyzed two different types of features: handcrafted features, and features obtained through transfer learning. We tested classifiers such as Transformers, Convolutional Neural Networks, and Traditional machine learning classifiers. results can be found in the results section.

3.1 Hand-Craft Features

We extracted two different sets of features from our dataset using different methods. First, we used the same approach as Brown C. et al. in [5]. The feature extraction process is illustrated in Figure 2.

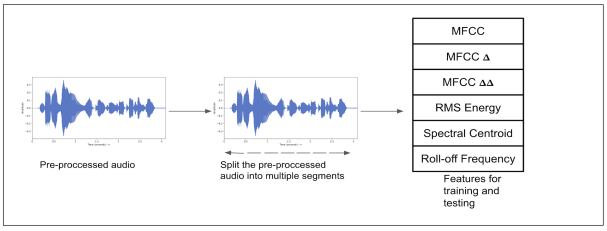

Figure 2: Hand-craft Feature Extraction: pre-processed audio recordings,are split into individual segments after which features such as MFCCs,MFCC velocity, MFCCs acceleration,RMS Energy, Spectral Centroid, and Roll-off Frequency are extracted.

The original audio recordings in MTG-Jamendo are sampled at 44.1 KHz. To make them compatible with standard audio processing tasks, we resampled them to 22KHz. We employed Librosa [3]as our audio processing library and extracted features from the audio at both the frame and segment level to encompass frequency-based, structural, statistical, and temporal attributes. A segment refers to a complete instance of one audio recording, whereas a frame is a subset of the audio data within a segment. We extracted time series features such as mel-frequency cepstral coefficients (MFCCs) and their variations, RMS Energy, Spectral Centroid, and Roll-off Frequency. We also obtained several statistical features to capture distributions beyond the mean, such as median, standard deviation, root-mean-square, maximum, minimum, 1st, and 3rd quartile, interquartile range, skewness, and kurtosis.

MFCCs have been commonly utilized as audio analysis features, particularly in automatic speech recognition applications [6]. We

implemented the conventional MFCC extraction method along with velocity (first-order difference) and acceleration (second-order difference), which have been demonstrated to enhance classifiers in previous studies [2]. The zero-crossing rate (ZCR) of an audio frame is the frequency at which a signal changes polarity, transitioning from positive to zero to negative or vice versa. Its value is particularly helpful in speech recognition and music information retrieval problems. RMS Energy is the root-mean-square of the magnitude of a short-time Fourier transform that provides the signal's power. Roll-off Frequency is the center frequency for a spectrogram bin such that at least 85% of the signal energy is below that frequency. Spectral Centroid refers to the average (centroid) extracted per frame of the magnitude spectrogram.. Overall, we extracted a total of 477 handcrafted features, including the first four segment-level features, four frame-level features represented by their statistics, and three variations of MFCCs with each component represented by its statistics ($4 + 4 \times 11 + 3 \times 13 \times 11 = 477$).

3.2 VGG-ish

Features from Transfer Learning: In addition to handcrafted features, we employ VGGish to extract audio features automatically [4]. The architecture of this network is inspired by the famous VGG networks used for image classification. The network consists of a series of convolution and activation layers, optionally followed by a max pooling layer. This network contains 17 layers in total (Figure 3).

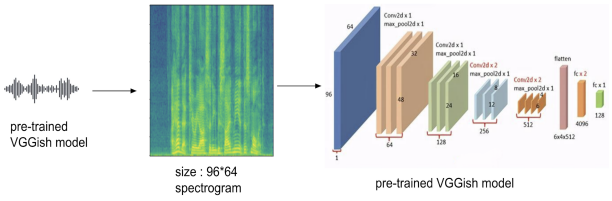

Figure 3: Features Extracted through VGGish, which returns a 128-Dimensional feature vector.

The VGGish model was trained using a large-scale YouTube dataset and the learned model parameters were released publicly. VGGish can be used in two ways:

- As a feature extractor: VGGish converts raw audio input features into a semantically, high-level 128-D embedding which can be fed as input to a downstream classification model. Since the VGGish embedding is more semantically compact than raw audio features, the downstream can be shallower than usual.
- As part of a larger model: In this case, we treat VGGish as a "warm start" for the lower layers of a model that takes audio features as input and adds more layers on top of the VGGish embedding. In this case, it will be trained to do the task of classification.

We employ it as a feature extractor to transform the raw audio waveforms into embeddings (features). It first resamples the data into 16KHz, then divides data samples into 0.96-sec non-overlapping sub-samples, and for every 0.96 seconds, it returns a 128-dimensional feature vector. After extracting features using this network, we can pass them to a shallower classifier in order to be trained. The data processing and feature extraction steps that we just did are the most unique aspects of our audio classification problem. From now on, the model and training procedure are quite similar to what is generally used in a standard image classification problem and are not special to audio deep learning.

3.3 Convolutional Neural Network

We use a CNN plus Linear Classifier model, and produce predictions about the class to which the audio belongs. Our model has four convolutional blocks which generate the feature maps. That data is then reshaped into the format we need so it can be input into the linear classifier layer and finally we use a sigmoid after the linear layer, which finally outputs the predictions for the classes. First, A batch of features is input to the model with shape (16, 1, 128, 1407). Then, Each CNN layer applies its filters to step up the depth of the features. By applying strides and kernels we reduce the width and height of the features. Finally, after passing through the four CNN layers, we get the output feature maps. After that, This gets pooled and flattened to a shape of (16, 128) and then input to the Linear layer. Finally, The Linear layer outputs one prediction score per class.

3.4 Transformers

The Transformer model was first introduced by Vaswani et al[17]. in 2017 for natural language processing tasks, and it quickly became a popular choice due to its ability to capture long-range dependencies in the input data.

The Transformer model is composed of a series of self-attention layers and feed forward layers. The self-attention layer allows the model to focus on different parts of the input data, while the feed forward layer transforms the output of the self-attention layer into a new representation.

In the case of a Transformer model that accepts 2D arrays, the input is a 2D array, such as an image, instead of a sequence of tokens. The 2D array is divided to patches, which is then processed by the self-attention and feed forward layers.

One important aspect of the Transformer model is that it does not rely on recurrent connections, which can be computationally expensive and limit the model's ability to capture long-term dependencies. Instead, the self-attention layer allows the model to directly attend to any part of the input sequence, making it more efficient and flexible.

3.5 Traditional Machine Learning Classifiers

- Label Power Set
  A Label Power Set (LPS) classifier is a type of multi-label classification algorithm used in machine learning. In multi-label classification, a single instance can be assigned multiple labels simultaneously, as opposed to traditional binary classification where a single instance is assigned a single label. LPS classifier works by transforming the multi-label classification problem into multiple binary classification problems. It does this by generating all possible label combinations

,
,

for the given set of labels, and then training a separate binary classifier for each label combination. The LPS classifier predicts the label set for a given instance by combining the predictions of all binary classifiers. This approach allows for efficient computation and effective handling of multi-label classification problems. However, it can become computationally expensive when the number of possible label combinations is large.

- ML-kNN
  The ML-kNN classifier is a multi-label classification algorithm that extends the traditional k-Nearest Neighbour (kNN) algorithm to handle multi-label classification problems. In this approach, the kNN algorithm is applied to each label independently, and the resulting predictions are combined to generate the final prediction for each instance.
  The ML-kNN classifier uses a voting system to determine the final prediction for each instance. The algorithm calculates the distance between a test instance and all training instances, and selects the k nearest neighbours. For each label, the algorithm then counts the number of positive and negative instances among the k nearest neighbours, and assigns the label to the test instance based on the majority vote. The final prediction for the test instance is then generated by combining the predictions for all labels. ML-kNN is known to be computationally efficient and has been shown to perform well on many multi-label classification tasks. However, it can be sensitive to the choice of k and can suffer from the curse of dimensionality when dealing with high-dimensional data.
- RAKEL
  The RAKEL classifier is a multi-label classification method that combines the Random Forest and k-Nearest Neighbour techniques. It transforms the multi-label problem into multiple binary problems by creating random label sets for each instance in the training data and training separate RF or kNN classifiers for each set. The predictions from all binary classifiers are then combined to predict the label set for a given instance. RAKEL is effective for multi-label classification and has been shown to produce results equal to or superior to other multi-label algorithms. One of its key benefits is its ability to deal with imbalanced datasets and produce consistent results with varying label combinations.

4 EXPERIMENTAL RESULTS

Dataset: The MTG-Jamendo Dataset was used to evaluate the proposed systems. The MTG-Jamendo Dataset [7] contains audio for 55,701 full songs and is built using music publicly available on the Jamendo 6 music platform under Creative Commons licenses. The dataset used for this task is the autotagging-moodtheme subset of the MTG-Jamendo dataset, this subset includes 18,486 audio tracks with mood and theme annotations. In total, there are 59 tags, and tracks can possibly have more than one tag. The minimum duration of each song is 30s, and they are provided in MP3 (320 Kbps bitrate). All tags were originally provided by the artists submitting music to Jamendo, but they were preprocessed with the goal of tag cleaning by the creators of the dataset. Multiple splits of the data are provided for training, validation, and testing. In this work, we use the split-0 and 59 most frequent tags. As this dataset has been released recently, not many studies are reporting the performance of the models using it yet. It is a useful addition to the evaluation methodologies followed by researchers in order to better assess the generalization of their models.

In this paper we tried to do the prediction of moods and themes conveyed by a music track, given the raw audio. The examples of moods and themes are: happy, dark, epic, melodic, love, film, space etc. Each track is tagged with at least one tag that serves as a ground-truth. The numbers of tags for this task are 59, which are very imbalance through the dataset, and as the reason of that, the classifiers perform very poorly. As a solution to this problem, we reduced the number of labels from 59 to 10, 8, and 5 label sets. In order to solve the mentioned problem, first we categorized the 59 labels into smaller groups based on their semantics. to do so, we used glove transformer and KMeans algorithm to cluster the label into 5, 8, and 10 label sets. We clustered the labels using euclidean distance. You can see the distribution of labels with different number of classes in Figures 5. As a result of this label reduction is improvement in performance (Table 2).

The length of the audio in the dataset are different from each other, since VGG-ish divides each audio into 0.96s segments and generates a 128-D feature vector for each segment, different lengths of the audio cause different sizes of the representations. As a result, the audio in the dataset was trimmed as 29.1s clips (the shortest signal in the dataset). The final representations of VGG-ish are of size 1407*128 for each audio.

The results of the proposed CNN architecture with 100 epochs using learning rate = 0.001 and adam optimizer are summarised in Table 2. We get the best performance using CNN when the number of labels is 5.

Performance measurement is a fundamental task in Machine Learning. A popular evaluation metric of binary decision problems is the area under receiver operating characteristic curve (ROC-AUC). this area, can be more instructive for evaluation on highly skewed datasets [8]. Hence, we use macro ROC- AUC to evaluate all considered music tagging models. Moreover, we use F1 score (macro and micro) which the former is the unweighted mean of the F1 scores calculated per label and the later is the normal F1 formula but calculated using the total number of True Positives (TP), False Positives (FP) and False Negatives (FN), instead of individually for each label.

As you can see in Table 1, by comparing other metrics we observe that classifiers that use features obtained through VGG-ish network are performing better in comparison with classifiers that use hand-craft features and features that are a combination of both VGG-ish and hand-craft features, which shows that VGG-ish can be a viable option for extracting audio features. So we were encouraged to go further and do more experiments on VGG-ish features.

Table 2 illustrates that reducing the number of labels is an effective way to improve performance. We get the best performance for the transformer (ROC_AUC = 72.8) when the number of labels is 8. although, number of labels is a hyper parameter that should be tune for each model.

One of the challenges of multi-label classification tasks is finding the threshold for each label, to be considered as being predicted or not. It is usually equal to 0.5 in most algorithms, but it is not a very

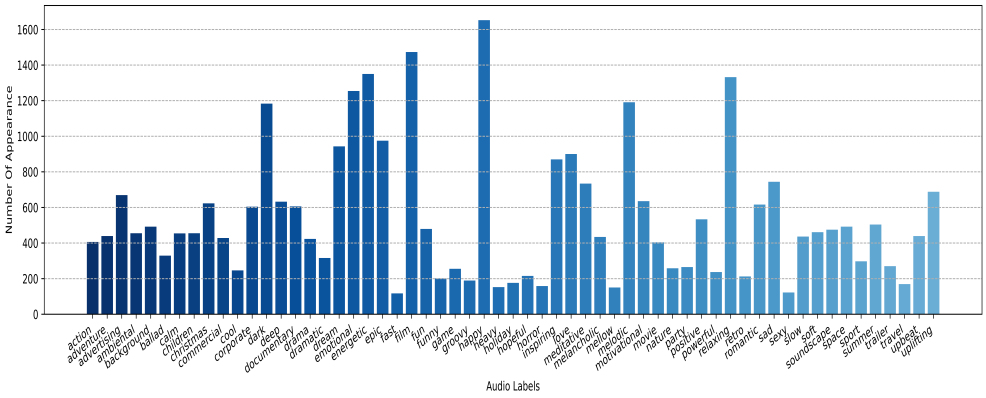

Figure 4: Distribution of labels through the dataset when the number of labels is 59.

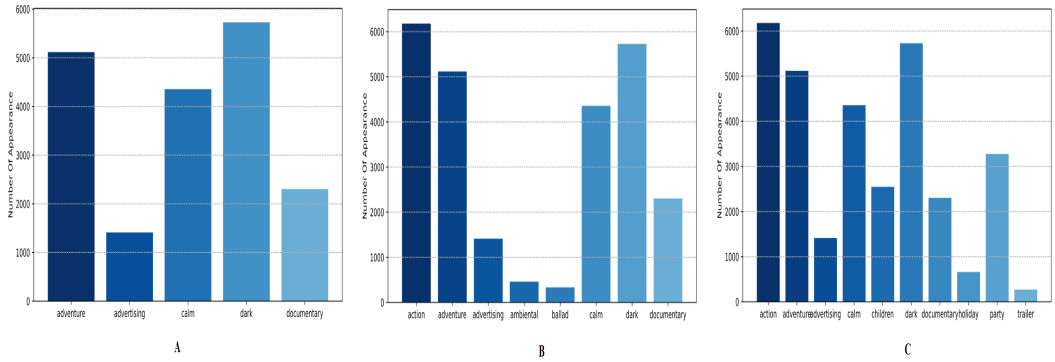

Figure 5: Distribution of labels through the dataset when the number of labels are 5(A), 8(B) and 10(C).

Table 1: A comparison between results of classifiers when using different feature vectors.

Feature extracting method	Classifier	N-Clusters	F1-macro	F1-micro	Roc-Auc	Avg-precision
Baseline(Hand-Craft)	Label Power set	59	9.1	11.0	60.1	4.3
	RakelD		10.9	12.0	62.0	4.6
	ML-KNN		7.6	6.3	55.7	3.6
VGGish	Transformer	59	17.1	**19.2**	62.2	6.6
	CNN		**29.1**	33.1	60.7	**16.7**
	Label Power set		10.9	12.3	63.2	4.5
	RakelD		15.9	17.0	**68.9**	6.0
	ML-KNN		11.4	13.9	57.1	9.7
Hand-Craft + VGGish	Label Power set	59	11.1	11.9	60.2	4.6
	RakelD		12.9	13.8	64.5	5.2
	ML-KNN		9.1	11.7	56.6	4.1

effective solution for this problem. What we propose is to predict a threshold for each label specifically based on their maximized F1 score, if our prediction is more than the threshold we consider the label to be predicted accurately.

5 CONCLUSION

We presented an automatic mood recognition algorithm based on VGG-ish network as a feature extractor. It was shown that features obtained through the VGG-ish network can be more effective for automatic music tagging and classification tasks.

,
,

Table 2: Results of different Classifier on reduce labeling on VGGish .

Feature extracting method	Classifier	n-Cluster	F1-macro	F1-micro	Roc-Auc	Average-precision
VGGish	Transformer	59	17.1	19.2	62.2	6.6
		10	41.4	45.7	70.7	25.6
		8	37.5	48.1	**72.8**	23.8
		5	52.7	56.6	67.8	27.8
	CNN	59	29.1	33.1	60.7	16.7
		10	37.2	41.1	63.6	29.7
		8	35.2	43.0	64.5	31.0
		5	54.1	56.0	**70.6**	44.2
	Label power set	59	10.9	12.3	63.2	4.5
		10	57.2	43.0	**67.3**	27.2
		8	38.2	41.1	64.6	26.5
		5	47.3	49.0	63.0	37.2
	RAKELD	59	15.9	17.0	**68.9**	6.0
		10	35.8	41.1	66.2	25.7
		8	36.6	38.9	62.8	25.1
		5	48.6	49.6	62.7	36.5
	ML-KNN	59	11.4	13.9	57.1	9.7
		10	17.1	18.3	59.4	18.9
		8	30.5	28.0	**63.9**	25.5
		5	42.0	47.7	60.9	35.2

In our first experiments, the proposed architectures with different input representations were compared using the MTG-Jamendo dataset. we confirmed that classifiers that use features generated through a VGG-ish network have better performance than the other proposed architectures. Then we compared different classifiers which use the same feature vectors(VGG-ish representations) and observed that CNN outperforms the other methods. Transformers are state-of-the-art architectures that have taken the world of NLP by storm in the last few years. In our experiments, we used a transformer for the classification task but because of the lack of resources we were not able to adapt it for our purpose, so we could not use its full potential. We also tried to adapt VGG-ish architecture in order to get better representations, but we failed due to the same problem of not having enough resources. we were able to just train it for one epoch through our dataset which had no improvement after all.

REFERENCES

[1] Er M B. Aydilek I B. 2019, 12(2): 1622-1634.. Music emotion recognition by using chroma spectrogram and deep visual features[J]. International Journal of Computational Intelligence Systems.

[2] M.M. Azmy. 2017. Feature extraction of heart sounds using velocity and acceleration of MFCCs based on support vector machines, in: 2017 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT).

[3] Dawen Liang Daniel PW Ellis Matt McVicar Eric Battenberg Brian McFee, Colin Raffel and Oriol Nieto. 2015. librosa: Audio and music signal analysis in python. In Proceedings of the 14th Python in Science Conference. Austin, TX, 18–24.

[4] J. F. Gemmeke A. Jansen R. C. Moore M. Plakal D. Platt R. A. Saurous B. Seybold M. Slaney R. J. Weiss Chaudhuri, D. P. W. Ellis and S. Hershey. K. Wilson. 2017. CNN architectures for large-scale audio classification. In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 131–135.

[5] Brown C. Chauhan J.; Grammenos A.; Han J.; Hasthanasombat A.; Spathis D.; Xia T.; Cicuta P.; Mascolo C. 2020.. Exploring Automatic Diagnosis of COVID-19 from Crowdsourced Respiratory Sound Data. arXiv 2020, arXiv:2006.05919.

[6] Kong-Pang Pun Wei Han. Cheong-Fat Chan, Chiu-Sing Choy. 2006. An efficient MFCC extraction method in speech recognition, in: IEEE International Symposium on Circuits and Systems.

[7] P. Tovstogan A. Porter D. Bogdanov, M. Won and X. Serra. 2019. The mtg-jamendo dataset for automatic music tagging.. In *Machine Learning for Music Discovery Workshop, International Conference on Machine Learning (ICML).*

[8] J. Davis and M. Goadrich. 2006, pp. 233–240. The relationship between precision-recall and roc curves.. In *in Proc. of the 23rd international conference on Machine learning.*

[9] Chen Q Liu X. et al, Wu X. 2017.. CNN based music emotion classification[J]. arXiv preprint arXiv:1704.05665.

[10] He D Wang Y. et al, Li F. 2020, 34(07): 12265-12272.. Multi-label classification with label graph superimposing[C]//Proceedings of the AAAI Conference on Artificial Intelligence.

[11] Kong Y Han D. et al, Han J. 2022, 16(6): 166335.. A survey of music emotion recognition[J]. Frontiers of Computer Science.

[12] Li P Li J. et al, Hu X. 2022, 121: 108259.. Learning common and label-specific features for multi-label classification with correlation information[J]. Pattern Recognition.

[13] Zhang L Liu S. et al, Yang X. 2021.. A simple transformer way to multi-label classification[J]. arXiv preprint arXiv:2107.10834.

[14] Wu B et al. Horner A, Zhong E. 2014: 117-126.. Music emotion recognition by multi-label multi-layer multi-instance multi-view learning[C]. Proceedings of the 22nd ACM international conference on Multimedia.

[15] Wu B Yang X Y. Li J, Dong Y Z. 2018, 24(4):365–389.. emotion recognition methods. Multimedia System.

[16] Chen C. Li Q. 2020, 2020: 1-11.. A multimodal music emotion classification method based on multifeature combined network classifier[J]. Mathematical Problems in Engineering.

[17] Jakob Uszkoreit Llion Jones Aidan N. Gomez Łukasz Kaiser Noam Shazeer, Niki Parmar and Ashish Vaswani. Illia Polosukhin. 2017. Attention is all you need. In Advances in Neural Information Processing Systems. 5998–6008.

[18] James Russell. 1980. A circumplex model of affect. In *Journal of Personality and Social Psychology. 39 (6): 1161–1178.*

[19] Trohidis K et al. Tsoumakas G, Kalliris G. 2011, 2011: 1-9.. Multi-label classification of music by emotion[J]. EURASIP Journal on Audio, Speech, and Music Processing.

[20] Hizlisoy S. Tufekci Z, Yildirim S. 2021, 24(3): 760-767.. Music emotion recognition using convolutional long short term memory deep neural networks[J]. Engineering Science and Technology, an International Journal.

[21] Yu Z B. Zhang M L. 2021, 44(9): 5199-5210.. Multi-label classification with label-specific feature generation: A wrapped approach[J]. IEEE Transactions on Pattern

Analysis and Machine Intelligence.