- ORIGINAL ARTICLE -

Transformer-based Automatic Music Mood Classification Using Multi-modal Framework

Clasificación automática del estado de ánimo de la música basada en transformadores utilizando un marco multimodal

Sujeesha Ajithakumari Suresh Kumar[1] and Rajeev Rajan[2]

[1,2] *Speech, Audio and Language Lab, College of Engineering, Trivandrum , APJ Abdul Kalam Technological University, Trivandrum*
{tve20ecsp16, rajeev}@cet.ac.in

Abstract

According to studies, music affects our moods, and we are also inclined to choose a theme based on our current moods. Audio-based techniques can achieve promising results, but lyrics also give relevant information about the moods of a song which may not be present in the audio part. So a multi-modal with both textual features and acoustic features can provide enhanced accuracy. Sequential networks such as long short-term memory networks (LSTM) and gated recurrent unit networks (GRU) are widely used in the most state-of-the-art natural language processing (NLP) models. A transformer model uses self-attention to compute representations of its inputs and outputs, unlike recurrent unit networks (RNNs) that use sequences and transformers that can parallelize over input positions during training. In this work, we proposed a multi-modal music mood classification system based on transformers and compared the system's performance using a bi-directional GRU (Bi-GRU)-based system with and without attention. The performance is also analyzed for other state-of-the-art approaches. The proposed transformer-based model acquired higher accuracy than the Bi-GRU-based multi-modal system with single-layer attention by providing a maximum accuracy of 77.94%.

Keywords: BERT, bidirectional GRU, music, self-attention, transformer

Resumen

Según los estudios, la música afecta nuestro estado de ánimo y estamos también inclinados a elegir un tema basado en nuestros estados de ánimo actuales. basado en audio técnicas pueden lograr resultados prometedores, pero las letras también dan información sobre los estados de ánimo de una canción que puede no estar presente en la parte de audio Por lo tanto, un multimodal con características tanto textuales como acústicas puede proporcionar una mayor precisión. Redes secuenciales tales ya que las redes de memoria a corto plazo (LSTM) y las redes de unidades recurrentes (GRU) son ampliamente utilizadas en el procesamiento de lenguaje natural (NLP) más avanzado. modelos Un modelo de transformador utiliza la atención propia para calcular las representaciones de sus entradas y salidas, a diferencia de las redes de unidades recurrentes (RNN) que utilizan secuencias y transformadores que pueden paralelizarse sobre las posiciones de entrada durante el entrenamiento. En este trabajo, propusimos un sistema de clasificación de estados de ánimo musicales multimodal basado en transformadores y comparamos el rendimiento del sistema usando un sistema bidireccional basado en GRU (Bi-GRU) con y sin atención. El rendimiento también se analiza para otros enfoques de vanguardia. El modelo basado en transformadores propuesto adquirió mayor precisión que el sistema multimodal basado en Bi-GRU con atención monocapa al proporcionar una precisión máxima del 77,94%.

Palabras claves: BERT, GRU bidireccional, música, autoatención, transformador.

1 Introduction

Music is a significant and often emotional experience for many people. Humans are deeply influenced by music in numerous ways. It helps to increase our memory and task pertinacity, light up our mood, turn down anxiety and depression, stave off fatigue, improve our pain response, and work out more efficiently. For example, music has been used to reduce stress and discomfort related to surgical and dental procedures, alleviate anxiety and depression in coronary care units, and boost recovery from heart attacks. We all experience joy, anger, sadness, and other emotions because we are all human beings. Each of us experiences a variety of emotions, which have an impact on our behaviour. Whether a person is in a good, bad, or depressed mood, music can influence their emotions, feelings, thoughts, and physical states. When we listen to music, the rhythm and tone we hear change our mood in various ways. Our hearts begin to beat in time

with the rhythm when we listen to it. Our brain interprets a slow heartbeat with high diastolic pressure as sad or depression. Love or happiness can be indicated by a dreamy rhythm with occasional upbeats, whereas a fast beat shows anger. Regarding tones, music in the "major key" transmits a happy message to our brain, whereas music in the "minor key" sends a sad message. All of this strongly impacts our brains, causing us to feel what is being conveyed to us genuinely.

In recent years, the online music industry has seen massive growth. Media streaming applications such as Apple, Spotify, and YouTube Music have become very popular. Access to immense music resources has increased the need to manage efficiently, index, search and organize music data. Categorizing music with label information such as genre, artist, and emotion is more convenient, among which classification based on emotion has become an important criterion. Listeners' moods can be a helpful representation of music recommendation systems. The mood is a psychological state of feeling related to internal emotions and affect, which is how emotions are expressed outwardly. Studies show that music not only affects our moods but also that we seem to choose music based on our current moods. Music mood classification based on acoustic features mostly depends on songs' spectral and rhythmic features. Lyrics-based classification exploiting natural language processing techniques is also gaining popularity. A classification model based on hybrid feature sets, audio, and lyrics can provide a more promising audio mood classification system. Manual classification is not feasible with a vast online library of songs. Hence we use music information retrieval (MIR) techniques [1] to fetch musical information from music repositories and arrange them according to query relevance. MIR is used in different fields, including computational music theory, music creation applications, music recommendation, classification, and music browsing interfaces.

The significant contributions of our research work can be concluded as follows:

1. Multi-modal methods are efficient compared to uni-modal methods for predicting the mood of a song.

2. Integrating attention mechanisms improves the performance of the system.

3. Multi-modal transformer-based approaches can enhance the system's efficacy, thereby proposing a spectrogram-independent multi-modal music mood classification system.

1.1 Related Works

Various attempts to classify songs based on moods and emotions using acoustic and textual features can be seen in the literature. Gordon C. Bruner [2] first attempted to classify songs according to moods. The work in [3] introduces a mood detection approach for classical music from acoustic data based on Thayer's model. In [4], a hierarchical framework is presented to automate the task of mood detection from acoustic music data by following some music psychological theories. Using audio mining techniques, implicit knowledge and data relationships from the audio and audio similarity measure are extracted in [5]. Later, implementing the term frequency-inverse document frequency (TF-IDF) embedding method on lyrics, Van Zaanen and Kanters [6] developed a machine-learning model based on emotion. It is worth noting that the fusion of lyrics and acoustic features significantly improves the performance of classifiers [7, 8]. In [9], a novel method for categorizing music by mood based on the content was provided. Three different modalities—audio, lyric, and MIDI—were employed in this research. Following acquiring three feature sets, they develop three variations of the standard co-training algorithm. The findings showed that these techniques could significantly raise classification accuracy. RNNs such as LSTM[11] and GRU [12] are used to process sequential data by storing previous inputs. In the study [13], Abdillah et al. employed the Bidirectional Long-short Term Memory (Bi-LSTM) deep learning method with GloVe to classify the song's emotions using the lyrics of the song. A multi-modal mood classification is done based on bi-directional LSTM (Bi-LSTM), and TF-IDF by Rajan R et al. [14]. In sequential networks such as LSTM and GRU, as the length of the sentence increases, it gets harder to capture the information in this vector because the meaning of every input sentence is captured in one vector. Its performance deteriorates with long sentences since it tends to forget parts of it, and gradually the hidden vector becomes a bottleneck. To solve this bottleneck problem, attention mechanisms are introduced. Self-attention allows the inputs to interact with each other and decides to give more weight to the relevant feature. It reduces the computational complexity of the layers processed. It's possible that assigning a high attention weight compared to the rest of the sequence will result in better results. Bahdanau et al. [15] introduced an additive attention mechanism that addresses the bottleneck problem when a fixed-length encoding vector is used in RNN. Dot-product attention scheme [16] has been used in various NLP applications and recently in audio processing applications [17, 18, 19, 20].

Ashish Vaswani et al. introduced transformers in [10]. Using attention mechanisms and positional embeddings transformer avoids recursion by processing sentences as a whole. The advantage of multi-headed attention in transformers is that different input vectors relate semantically in multiple ways. Transformers are widely used in NLP, and in past years, different variants and pre-trained models such as bidirectional encoder representations from transformers (BERT) and distilBERT [21, 22, 23, 24] are also introduced. Since

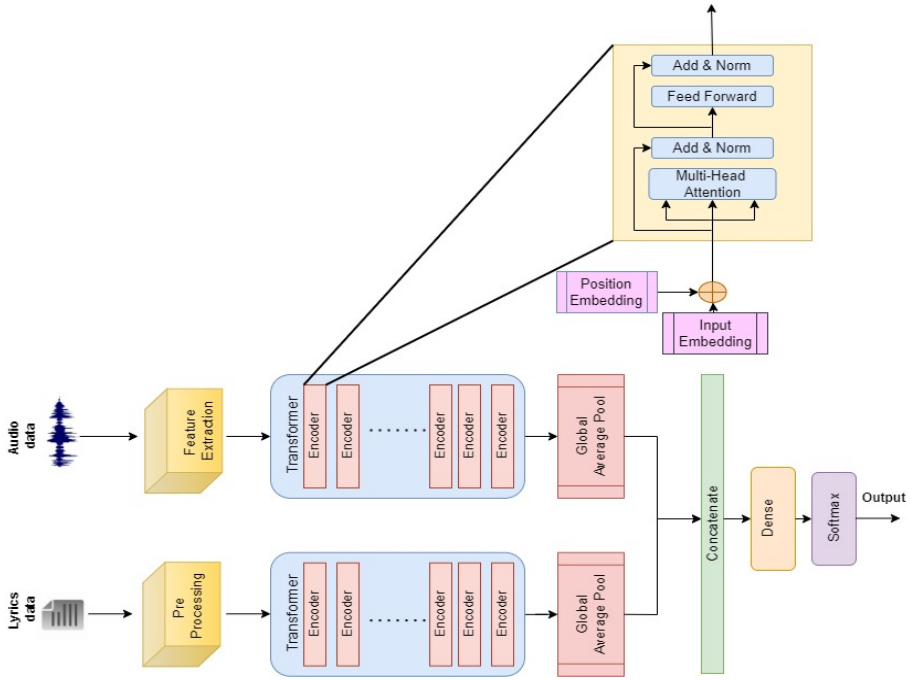

Figure 1: Multi-modal fusion architecture based on the transformer. Transformer architecture is incorporated from [10]

transformers are good at processing sequential data, we can also use them for audio classification problems. Transformers working with spectrogram[25, 26] were also introduced recently. The study in [27] , proposed a transformer-based approach model using XLNet. The authors also employed a robust methodology to enhance the accuracy of web crawlers for extracting lyrics. Pyrovolakis et al. [28] examined and compared single-channel and multi-modal approaches for music mood detection by applying deep learning architectures. They proposed a multi-modal methodology for classification based on convolutional neural networks (CNN) and BERT. They also proved that the correct extraction and combination of audio features could further improve the prediction goal.

1.2 Motivation

Generally, music is classified according to different genres, album names, artists, etc. Ordinary people, especially those who don't know much about the genre, often find it difficult to choose songs based on these classifications. Since music can change moods and relieve stress, most people choose to hear music based on their moods. This necessitates the development of a user-centric music classification system based on mood. Such a system can ease the selection of songs and reduce browsing time. Musical information can be derived from both audio as well as lyrics features. The use of both modalities can result in a system with improved performance. Besides, the potential of transformers to understand the relationship between sequential elements has not yet been explored in music mood classification. These factors motivated us to develop a multi-modal architecture based on a user-centric music mood classification system.

2 System Description

The proposed model is a multi-modal music mood classification system based on a transformer. The transformer architectures take advantage of the attention mechanism to enhance the performance of deep learning NLP translation models. The transformer made it possible to facilitate greater parallelization during training, which significantly speeds up training. A detailed block representation of the transformer-based multi-modal mood classification system is given in Figure 1. Initially, we experimented with Bi-GRU and CNN frameworks. Later we studied the effect of the attention mechanism on the task. Two modalities of a song - text, audio and their fusion are employed in the feature extraction phase. Four mood classes, namely, aggressive, happy, sad, and relaxed, are considered in this study.

The acoustic features with position encoding are initially fed to the transformer model. The transformer model contains a stack of encoders. Each encoder consists of multi-head attention, point-wise feed-forward networks, and layer normalization. Multiple attention mechanisms enable the model to capture more relationships between inputs than possible with a single attention mechanism. The stack of encoders processes this input sequence and produces an encoded representation of the input sequence.

The lyrics contain words that must be pre-processed before being fed to the model. Cleaning, stop word

Table 1: Extracted audio features

Sl. No	Features	Description
1	Zero Crossing Rate (ZCR)	It is the rate at which the sign of a signal is changed. It detects whether a speech frame is voice, unvoiced, or silent. Unvoiced segments give higher ZCRs than voice segments, and ideally, ZCRs are zero for silence segments.
2	Chroma	Chroma vector consists of 12-elements which show the energy content corresponding to the 12 pitch classes in the song. For chroma implementation, STFT analysis is used.
3	Spectral Centroid	It indicates where the " center of mass" for a sound is located.
4	Spectral Bandwidth	It is the difference between upper and lower frequencies of the spectrum.
5	Spectral Flatness	Spectral flatness characterizes the audio spectrum. It helps to measure how much sound resembles a pure tone. A high spectral flatness means the spectrum is similar to white noise. It is also called tonality coefficient.
6	Spectral Roll-off	RMS is the root-mean-square-energy. It helps to perceive loudness, which can be used for event detection.
7	RMS	Root Mean Square energy helps to perceive loudness, which can be used for event detection.
8	Tempo	It determines beats per minute. It helps to identify the speed at which musical piece is played.
9	Tonnetz	It gives tonal centroid features of 6 pitch classes
10	MFCC	Mel Frequency Spectrum provides better representation of audio because frequency bands are equally dis- tributed in mel scale. It gives the overall shape of the spectral envelope.
11	PLP	Predominant Local Pulse is used to find stable tempo for each frame.

removal, and stemming have been done as part of pre-processing step. We employed the BERT embedding technique to compute textual vectors. BERT uses wordpiece embedding input for tokens. In the multimodal framework, acoustic and textual feature vectors are fed separately through the global average pool layers. Then these two layers are concatenated, and features are provided through the dense layer. Finally, the softmax activation function is used at the output layer for predicting the moods according to songs. We also experimented with other deep learning architectures such as Bi-GRU, XLNet, and CNN to analyze the performance from a multi-modal perspective. The following subsections describe feature extraction and classification methods in detail.

2.1 Feature Extraction

Feature extraction helps to identify and extract key features in the input data set. It transforms raw data into numerical features by preserving the information in the original data set. We computed two sets of features from audio and lyrics.

2.1.1 Audio Feature Extraction

The acoustic features are extracted using the Librosa package [29]. The acoustic features extracted are listed in Table 1. Zero-crossing rate (ZCR), tempo, and predominant local pulse are temporal features, and the rest are spectral features. Mel-frequency cepstral coefficient (MFCC) features are the most commonly used feature for audio classification. Here, 25 MFCC coefficients, its first derivative(delta) and second derivative(delta-delta) features are extracted. The whole features extracted for a song are concatenated for a single representation before proceeding with training.

2.1.2 Lyrics Feature Extraction

The lyrics provide a narrative and additional intrigue that instrumental music alone cannot provide. Words in lyrics play an important role in evoking emotions. Figure 2 shows a word cloud representation of commonly occurring words in four mood corpora. These words are pretty helpful in determining a song's mood.

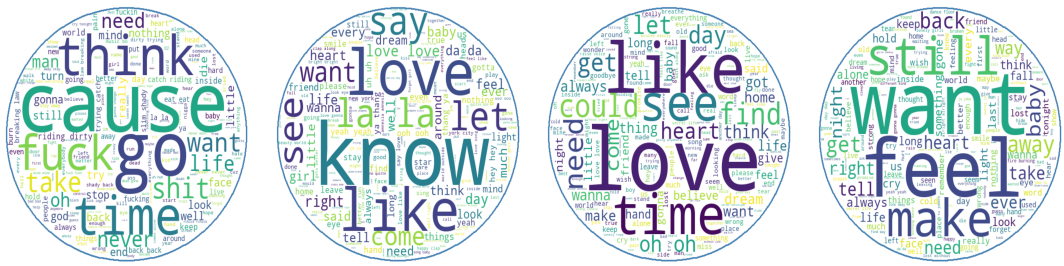

Figure 2: Most common words in (a) aggressive corpus (b) happy corpus (c) relaxed corpus (d) sad corpus

Four embedding techniques, namely, GloVe, Word2Vec, BERT, and XLNet, are used for this research work. Cleaning, stop word removal, and stemming have been done as part of pre-processing step, followed by vectorization.

- Word2Vec: Word2Vec accepts text inputs and gives corresponding vectors as outputs. The semantic closeness of the words to each other are also revealed in this representation. The length of the vector depends on the corpus size. The relationship is derived using the cosine distance between the words. The vector representation of each word in the corpus places words with similar contexts next to one another in the vector space. The one-hot encoding is used to show the word vectors in the Word2Vec models. These word vectors serve as inputs to the neural network, which sums the inputs with parameter tables in its hidden layers. The softmax function is then applied at the output layer to predict the right word positions in the one-hot vectors of the output. Word2Vec has two architectural modeling options, the CBoW (Continuous Bag of Words) and the Skip-gram, for the vector representation of the words. The CBoW model attempts to predict the target word in the output using the nearby words of a target word as input. The Skip-gram model takes a word as input and forecasts the nearby words as output. Skip-gram represents uncommon words or phrases well and performs well with small datasets. While CBOW can better represent more frequent words and trains more quickly than Skip-Gram.

- Global Vectors (GloVe): GloVe encodes the co-occurrence probability ratio as a vector difference between the words. The co-occurrence matrix tells how often a particular word pair occurs together. Each value in the co-occurrence matrix represents a pair of words occurring together. GloVe uses a weighted least squares objective that minimizes the difference between the dot product of the vectors of two words and the logarithm of their number of co-occurrences[30].

$$J = \sum_{i,j=1}^{V} f\left(X_{i,j}\right)\left(w_i^T w_j + b_i + bj - logX_{ij}\right)^2 \quad (1)$$

where w_i and b_i are the word vector and bias respectively of word i, w_j and b_j are the context word vector and bias respectively of word j, X_{ij} is the number of times word i occurs in the context of word j , and f is a weighting function that assigns lower weights to rare and frequent co-occurrences.

- BERT: The BERT tokenizer accepts text as input for tokenization. While maintaining the occurrence order of words, it creates a sequence of terms-words matching each input word in the corresponding term provided by its vocabulary. The tokenizer tries to break down input words that are not recognized in the vocabulary into vocabulary tokens to the maximum number of characters, and it may split a word into characters. When a word is split, the first token will remain in the order it appears, and the subsequent tokens will build on each other using the double symbol # at the start. Consequently, the model can recognize the tokens that result from splitting.

- XLNet: The process of making embeddings for XLNet is different from BERT; first, we will tokenize the texts with sentencepiece, then, we will add “<sep>”, ”<cls>” and pad mask to the embeddings. In XLNet, the word token output is calculated by taking into account the permutation of all word tokens in the sentence.

2.2 Classification Schemes

Various classification models such as support vector machine (SVM)-based classifier, Bi-GRU with self-attention model, CNN-based model and transformer-based models are utilized in the study and compared with the proposed model.

2.2.1 Support vector machine-based classifier

As a baseline machine learning-based classifier, we experimented with SVM. The goal of the SVM algorithm is to create a hyperplane that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future.

2.2.2 Bi-GRU with self-attention

Audio, lyrics, and multi-modal architectures are trained using networks without and with attention. A bidirectional GRU, or Bi-GRU is a bidirectional recurrent neural network that consists of two GRUs - one takes the input in the forward direction and the other in the backward direction. The GRU model consists of two gates [12]: the reset gate r and the update gate z. At time step, t, the GRU unit output, h_t, is calculated as follows

$$z_t = \sigma\left(W_{xz}x_t + U_{hz}h_{t-1}\right) \tag{2}$$

$$r_t = \sigma\left(W_{xr}x_t + U_{hr}h_{t-1}\right) \tag{3}$$

$$\tilde{h}_t = \tanh\left(Wx_t + U\left(r_t \odot h_{t-1}\right)\right) \tag{4}$$

$$h_t = \left(1 - z_t\right) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{5}$$

where the feedforward weights of the update gate z_t, the reset gate r_t, and the output candidate activation h_t at time step t are W_{xz}, W_{xr}, and W. The recurrent weights of the update gate z_t, the reset gate r_t, and the output candidate activation h_t, are U_{hz}, U_{hr}, U respectively. The symbol $\odot$ denotes the element-wise (Hadamard) multiplication. σ is the logistic sigmoid function and tanh is the hyperbolic tangent function.

Attention is "withdrawal from something to deal effectively with others"[31]. The idea is extended to deep neural networks by focusing on certain input features while ignoring others. Attention models are implemented by relating each output sequence to a certain part of the input sequence before producing the output. Attention can be global or local, depending on whether the attention is given to all the input positions or a subset. They differ in the derivation of the context vector for the input sequence. The alignment score function can be additive, dot product, and scaled dot product. Self-attention, also known as intra-attention, relates different positions of an input sequence to obtain a representation of the entire sequence. It learns self-alignment[10]. Our study employs self-attention based on the dot product alignment function for calculating the attention, β, between input and output.

2.2.3 Convolutional Neural Network

The convolutional neural network (CNN) is one of the most well-liked deep neural networks. A basic CNN comprises several layers, each of which converts one volume of activations into another using a differentiable function. Convolutional, pooling, and fully-connected layers are the layers used by CNN. Complete CNN architecture is built by stacking these layers.

- The convolutional layer calculates the output of neurons connected to local regions in the input, each computing a dot product between their weights and a small region connected to the input volume.
- ReLU applies an element-wise activation function. It does not change the volume.
- Pooling layers perform a downsampling operation along width and height (spatial dimensions).
- The arrangement of neurons in a fully connected layer is comparable to that of the traditional neural network. In a fully connected layer, each node is directly connected to every other node in the layer previous to and next to it.

2.3 Transformer

The transformer is a network architecture relying entirely on attention mechanisms, eschewing recurrence and convolutions completely [10]. The transformer uses multi-head self-attention for computing representations of the input sequence. Transformer model architecture follows an Encoder- decoder structure. But for classification purposes here, we used an encoder only. It consists of a positional encoding layer, an encoding block (repeats N_x times), a softmax layer, and a linear layer. The encoding block consists of a position-wise fully connected feedforward sub-layer and a multi-head self-attention sub-layer. The input is first passed through the positional encoding layer. The network benefits from comprehending the relative or absolute positional information in each sequence. Then it is fed through the encoding blocks N_x times. The linear and softmax layer receives the output of the final encoding block. Transformer architecture is described as follows:

Multi-head attention techniques implement self-attention layers running in parallel. Queries, keys, and values are linearly projected h times with different, learned linear projections to *dk*, *dq*, and *dv* dimensions, respectively. The attention function is applied parallel to the projected queries, keys, and values versions. The output will have dimension dv. Parallel outputs are then concatenated, resulting in the final attention vector. Queries and keys have dimension dk and values have dimension dv. The dot products of the query with all keys are computed, divided by $\sqrt{dk}$, and finally, a softmax function is utilized to obtain the weights on the values. This is extended to a set of queries, keys, and values packed into matrices and represented by Q, K, and V, respectively. Figure 3 shows the schematic representation of multi-head attention.

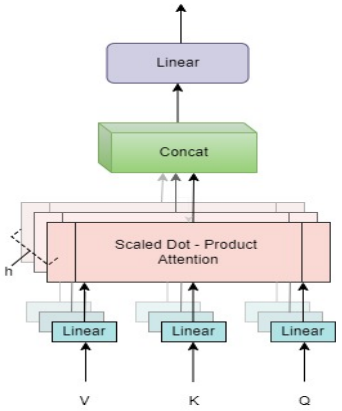

Figure 3: Multi-head attention [10]

$$Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}}) \qquad (6)$$

3 Performance evaluation

3.1 Dataset

The available dataset that we used for training and evaluation is a subset of MoodyLyrics Dataset [32] It consists of a set of 2000 song titles alongside their corresponding mood label, from four basic moods—happy (Q1), aggressive (Q2), sad (Q3) and relaxed (Q4). This dataset does not contain the needed data (audio files, lyrics, or any information on the genre). It only provides less information, such as song titles, related artist names, and mood labels. Using this data, we gathered 680 songs' audio and lyrics. The lyrics of the entire song are saved in .txt format. The words that make up a song are called the lyrics and provide strong and relevant information about the emotional state that a song can elicit in the listener. Audio files are truncated to 30 seconds and saved as .wav files. Songs are classified based on Russel's emotional circumplex model[33]. All human emotions, according to circumplex, are dispersed in a circular two-dimensional space with valence and arousal axes. The dataset is divided by the ratio of 70:10:20 as train, validation, and test pattern.

3.2 Experimental Setup

Initially, we implemented a support vector machine (SVM)-based classification for the proposed work. Eleven acoustic features are extracted using the librosa package and combined to form an input tensor (2191, 119). We imported the SVC class from Sklearn.svm library to create the SVM classifier. The extracted audio features are used to train the SVM classifier using audio features. We have used kernel='poly' and the degree='3', for better results. The polynomial kernel displays the similarity of the vectors in the training set of data in a feature space over the polynomials of the initial variables utilized in the kernel. Similarly, we have implemented an SVM classifier using lyrics features. The better result is obtained when we employ kernel='rbf', c='1', and gamma= '1e-3'. A Radial basis kernel (RBF) non-linearly maps samples into a higher dimensional space. Following that, we merged lyrical and audio features and developed an SVM classifier. We have applied kernel='rbf', c='10', and gamma='1e-3'.

Afterwards, we explored deep learning models. The extracted audio features are fed to the Bi-GRU layer of 512 units. A batch normalization layer is used for normalizing inputs. The normalized inputs are passed through several dense and dropout layers with a dropout value of 0.3. Finally, the output layer consists of the softmax activation function, which predicts the mood classes. The model is trained over 100 epochs with a batch size of 32.

We also experimented with a CNN using audio features. The features are fed to a 2D convolution layer of filter size 32 and kernel size 7×7, followed by the ReLU activation layer and max-pooling layer. Again it passes through a series of convolution layers (filter size=64, 128, 256, and kernel size=3) and max-pooling layers. The max-pooling layer has a stride size of 2. With a batch size of 32, the model is trained over 100 epochs.

For lyrical features, the experiments are done with two models - Word2Vec and GloVe. Experiments are carried out with various embedding dimensions and maximum word lengths (maximum number of words). Several combinations of embedding dimensions (100, 200, 300) and maximum word length (100, 200, 300) are considered for analyzing the best combination. A Bi-GRU network of 256 units is used to train the model with several dense and dropout layers with a dropout value of 0.3. Batch normalization is also integrated into the model. ReLU is used in dense layers, and softmax is used in the output layer as an activation function. The model is trained for 100 epochs. A batch size of 32 is used. In the GloVe framework, we experimented with various vector-length pre-trained representations. The co-occurrence matrix is built from these vectors, and its untrained weights are replaced at the level of integration. The embedding layer uses it as a weight matrix.

In the multi-modal system, two modalities are used - audio and text. The acoustic and textual features were computed using Word2Vec/GloVe embeddings combined to obtain a hybrid model. The acoustic features are fed into a Bi-GRU network with 512 units and dense layers with 256 and 128 neurons with a dropout of 0.3. Textual features are processed through a Bi-GRU layer, followed by dense layers with 256 and 128 neurons and a dropout of 0.3. The acoustic and lyrical features are then concatenated and passed through layers of 256, 128, and 64 neurons with a dropout of 0.3. A batch normalization layer is also used for normal-

izing the input data. In the training phase, an Adam optimizer with a learning rate of 0.0001 was utilized. The batch size used is 32, and the model has iterated over 100 epochs.

We also experimented with an attention framework for a multi-modal system. An attention layer in the Word2Vec-based text input channel is included to capture the relevant aspects. Acoustic features are extracted and fed to a Bi-GRU network of 512 neurons after batch normalization. Then it passes through a self-attention layer. The output from the attention layer is then fed to dense layers. After that, the audio and lyrics features are concatenated and processed via dense and dropout layers. Since the ReLU is the function most frequently employed for hidden layers, it is used in this instance. At the output layer, the softmax activation function is used. The model is trained for One hundred epochs with a batch size of 32. An early stopping with patience ten is also used to train the models better. In the next hybrid model based on attention, we used GloVe embeddings and repeated the process as in the Word2Vec-based multi-modal with attention. With a single attention layer, we acquired better results for the multi-modal system.

Inspired by these results, we implemented a transformer-based music mood classification system that uses multi-head attention. Different input vectors relate to each other semantically in multiple ways. Multi-head attention catches such types of relations because the heads work parallelly. We analyzed the uni-modal transformer classification model based on textual features. For this, we use a pre-trained transformer model BERT. The BERT model was pre-trained on BookCorpus, a dataset consisting of 11,038 unpublished books and English Wikipedia (excluding lists, tables, and headers). BERT is simply a transformer architecture Encoder stack. The Encoder stack of $BERT_{BASE}$, one of the pre-trained BERT models, includes 12 layers. It also has larger feed-forward networks (768) with 12 heads. It has 110 million parameters. The BERT model requires a particular representation of the input data to operate correctly, called BERT embeddings. In particular, the BERT's tokenizer generates a sequence of word tokens by comparing each input word to the dictionary of the BERT. Tokenizer appends the token [CLS] to the beginning of the list once tokenization is complete and the [SEP] token at the end of each sentence. BERT foresee three parallel vectors of fixed length 128 - input_ids ,input_masks and segment_ids. The identification IDs of each token in the input are used to create the vector input ids. These IDs are stored in the dictionary of the model. Vector segment ids aid in the separation of sentences that make up an input. Sequences greater than 128 characters will be trimmed. The sequence is filled with empty tokens if the number of tokens is less than 128. Each of BERT's 12 layers is an encoder, and each encoder is made up of three various embedding vector processing layers. The first layer implements a multi-head-attention mechanism with 12 heads of attention. The second layer comprises a normalization layer and a feed-forward network. It also includes a layer of each encoder which consists of a position encoding technique that integrates position information in embedding vectors. The encoder outputs are fed to a global max-pooling layer followed by a dense layer of 16 units with a ReLU activation function. Later the features are fed to the output layer, which consists of the softmax activation function. The BERT model is trained over 30 epochs. The learning rate adapted is 0.0001.

Yudhik Agrawal [27] proposed a transformer-based approach to music emotion recognition from lyrics using XLNet, which was adopted to study. XLNet is an auto-regressive language model. A greater understanding of contextual information is possible for the network due to the integrated recurrence of the transformer. We use the adam optimizer with an initial learning rate of 2e-5 and a dropout regularization with a 0.1 discard probability for the layers. We used cross-entropy loss here. A batch size of four was used. As they are trained on large corpora, pre-trained (XLNet-base-cased) models have access to rich information. Training the classifier is quite cheap because the pre-trained model layers already encode rich linguistic information.

Multi-modal system (CNN+BERT) proposed in [28] is also implemented for the performance comparison. Acoustic features extracted are trained using the CNN model, and for lyrics, BERT is used. Later both models are combined, and finally, integrated features are processed through a softmax output layer. Adam optimizer with a learning rate of 2e-5 is used. 32 is used as batch size.

Next, we analyzed a multi-modal classification system based on Bi-GRU and BERT. The acoustic features are fed into a Bi-GRU network with 256 units and dense layers with 256, 128, and 64 neurons with a dropout of 0.2. Text is transformed in representations BERT expects, as described in section 2.1.2. Later BERT embedding features passed through a stack of encoders. The output is pooled and fed to a dense layer of 16 units. A dropout of 0.2 is given to layers. Acoustic and textual dense layers are concatenated and passed through the dense layer of 16 units and then through the output layer. Adam is chosen as the optimizer for the training process, with a learning rate of 0.0001. The number of epochs for this process are selected to be equal to 20.

Finally, we introduce a better efficient model, which is purely based on transformers. Here, both acoustic and textual features are trained using transformer models. The architecture of the proposed model is shown in Table 2. As a first step, we have created a transformer-based model using acoustic features. Here we need the encoder part of the transformer. We select the number

Table 2: Architecture of proposed transformer-based multi-modal music mood classification system

Sl.No	Layer	Output Shape
1	Audio Input	(None, 2191, 119)
2	Audio encoder(12 layers)	(None, 2191, 119), (None, 2191, 119), (None, 2191, 119), (None, 2191, 119), (None, 2191, 119), (None, 2191, 119), (None, 2191, 119), (None, 2191, 119), (None, 2191, 119), (None, 2191, 119), (None, 2191, 119), (None, 2191, 119)
3	Dropout	(None, 2191, 119)
4	Global Average Pooling1D	(None, 119)
5	Dense	(None, 64)
6	Text Input	'input_mask': (None, 128), 'input_word_ids':(None, 128), 'input_type_ids': (None, 128)
7	BERT Encoder(12 layers)	'sequence_output': (None, 128, 768), 'encoder_outputs': [(None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768), (None, 128, 768)], 'pooled_output': (None, 768), 'default': (None, 768)
8	Dropout	(None, 768)
9	Dense	(None, 64)
10	Merge	(None, 128)
11	Dense	(None, 32)
12	Dropout	(None, 4)

of encoder layers (N_x) as 12, the number of heads in the multi-head attention models as 7, the dimension of the feed-forward network model(dff) as 2048, drop out value of 0.3, and the number of expected features in the encoder(d_{model}) as 119 (dimension of input features extracted). ReLU is used as the activation function. Next, we developed a transformer classification model based on textual features. We have used a pre-trained $BERT_{BASE}$ model. Textual features in BERT embeddings are fed to the stack of encoders. The output from encoders is fed to the pooling layer. The pooling layer and dense layer concatenated both the acoustic and lyrical models and passed through an output layer with a softmax activation function. The pooling layer is used to reduce the dimension of inputs from transformer encoders. We used Adam as the optimizer with a learning rate of 0.00001 over 20 epochs.

3.3 Results and Analysis

Precision, recall, F1-score, and overall accuracy are used to evaluate performance. The evaluation parameters show that the proposed multi-modal architecture with attention and transformer models outperforms multi-modal architecture without attention and single-modal architectures. The evaluation parameters gradually increase from single-modal architectures to multi-modal architectures with transformers.

The overall accuracy of the SVM classifier based on audio features is 52.94%. The average precision, recall, and F1 score is 0.55, 0.53, and 0.53, respectively. The lyric-based SVM classifier obtained an accuracy of 32.35% with average precision, recall, and F1 score of 0.33, 0.32, and 0.31, respectively. With average precision, recall, and F1 score of 0.62, 0.60, and 0.59, respectively, the audio+lyrics-based SVM classifier achieved an accuracy of 59.55%. Precision, Recall, and F1 score of three SVM classifiers are given in table 3

The performance metrics for audio-based classification using Bi-GRU and CNN networks are given in Table 4. Average precision, recall, and F1-score of 0.56, 0.56, and 0.55 are reported for Bi-GRU-based classification and 0.63, 0.62, and 0.62 are reported for CNN-based classification. The highest metrics values are reported for the aggressive class for the Bi-GRU model. For the CNN model, the highest precision and F1 score is attained for the class sad, and the highest recall is for the class happy. The Bi-GRU model and CNN model have provided an overall accuracy of 56.00 % and 61.76%, respectively.

For the second phase with textual features, the results are tabulated in Table 5, for the embedding dimension and maximum word length as 100 for

Table 3: Precision, recall and F1 score for audio-based, lyrics-based, and audio+lyrics-based classification using SVM

	Lyrics			Audio			Audio+Lyrics		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Aggressive	0.28	0.26	0.27	0.85	0.68	0.75	0.85	0.68	0.75
Happy	0.35	0.50	0.41	0.56	0.65	0.60	0.51	0.76	0.61
Relaxed	0.31	0.38	0.34	0.38	0.44	0.41	0.54	0.41	0.47
Sad	0.36	0.15	0.21	0.40	0.35	0.38	0.56	0.53	0.55
Average	0.33	0.32	0.31	0.55	0.53	0.53	0.62	0.60	0.59

Table 4: Precision, recall and F1 score for audio-based classification

	Audio(Bi-GRU)			Audio(CNN)		
	P	**R**	**F**	**P**	**R**	**F**
Aggressive	0.69	0.74	0.71	0.52	0.44	0.48
Happy	0.56	0.44	0.49	0.62	0.76	0.68
Relaxed	0.49	0.59	0.53	0.53	0.59	0.56
Sad	0.50	0.47	0.48	0.85	0.68	0.75
Average	0.56	0.56	0.55	0.63	0.62	0.62

both Word2Vec and GloVe. An accuracy of 44.15% is obtained for the Word2Vec classification. The model's performance is analyzed with various combinations of embedding dimension and maximum word length. Among all the combinations, the combination (100,100) has the highest accuracy of 50.73% for GloVe, compared to others. Figures 4 show the performance of the Word2Vec and GloVe systems for various combinations of embedding dimension and maximum word length. The performance of Word2Vec and GloVe embedding techniques are compared. In both methods, the highest accuracy is reported for aggressive and happy. It is important to note that the GloVe model performs better than the Word2Vec model.

Table 5: Precision, recall and F1 score for lyric-based classification

	Lyrics(Word2Vec)			Lyrics(GloVe)		
	P	**R**	**F**	**P**	**R**	**F**
Aggressive	0.63	0.50	0.56	0.70	0.76	0.73
Happy	0.49	0.59	0.53	0.56	0.44	0.49
Relaxed	0.34	0.41	0.37	0.47	0.41	0.44
Sad	0.33	0.26	0.30	0.33	0.41	0.37
Average	0.45	0.44	0.44	0.52	0.51	0.51

As a fusion model, audio and textual features are combined to build a hybrid model. The metrics are tabulated in Table 6. Average precision, recall, and F1 scores of 0.67, 0.64, and 0.63 and 0.67, 0.65, and 0.65 are reported for multi-modal classification with Word2Vec and GloVe, respectively. It is worth noting that the metrics got improved in the fused model. From these results, the performance will be improved if we use both acoustic and lyrical features. The overall accuracy of our hybrid model based on Word2Vec and GloVe are 63.97% and 65.44%, respectively. The efficacy of the multi-modal system can be inferred by comparing the three confusion matrices.

Table 6: Precision, recall and F1 score for multi-modal classification

	Multi-modal(Word2Vec)			Multi-modal(GloVe)		
	P	**R**	**F**	**P**	**R**	**F**
Aggressive	0.69	0.59	0.63	0.76	0.56	0.64
Happy	0.59	0.56	0.58	0.67	0.65	0.66
Relaxed	0.84	0.47	0.60	0.56	0.65	0.60
Sad	0.57	0.94	0.71	0.67	0.76	0.71
Average	0.67	0.64	0.63	0.67	0.65	0.65

As mentioned in Section 2, we carried out experiments to see the effect of attention on the proposed task. First, we attempted a self-attention-based music mood classification system. The word clouds shown in Figure 2 expresses the frequent and dominant words for each mood. The attention mechanism provides more attention weight to these words. Similarly, the model also pays attention to a song's dominant features for acoustic features. Paying attention to the relevant textual and acoustic features helps to predict the mood of a song more precisely and hence improves the performance of the multi-modal system. The performance matrices for multi-modal music mood classification using the self-attention mechanism are shown in Table 7.

We can observe that attention networks have increased the performance of hybrid models. Class aggressive, relaxed, and sad are predicted better compared to class happy in both Word2Vec and GloVe-based hybrid models. The metrics are improved to 0.75, 0.74, and 0.73 for the Word2Vec model and 0.78, 0.76, and 0.77 for the GloVe model, respectively for self-attention. The overall accuracy obtained for the attention-based Word2vec hybrid model is 73.52 %, and that for the attention-based GloVe

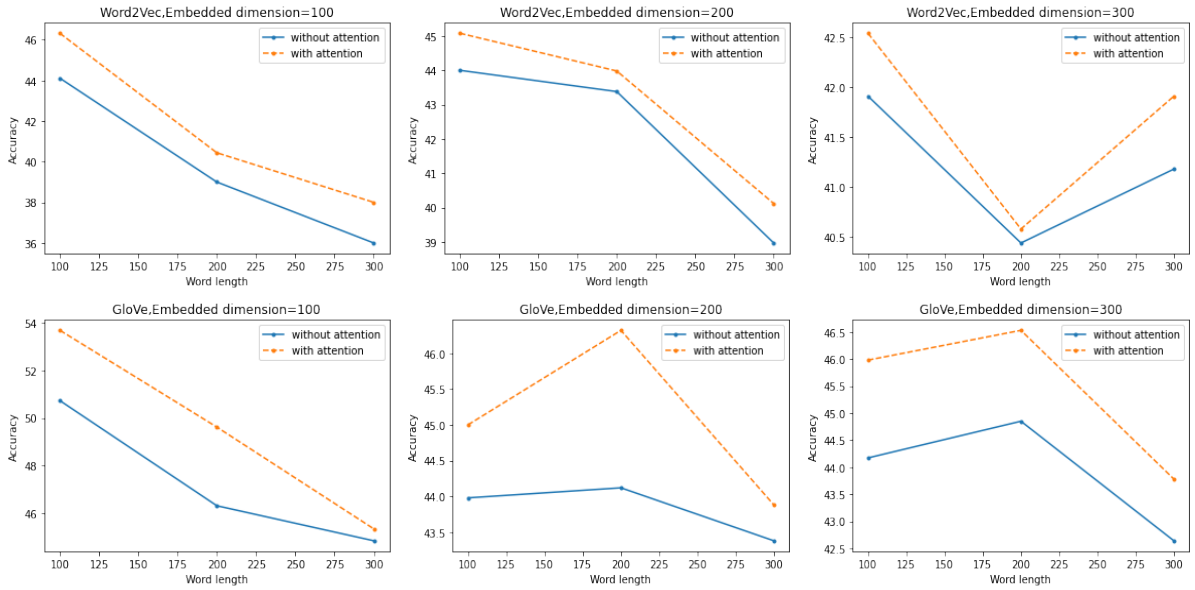

Figure 4: Performance of the Word2Vec system and GloVe system for various combinations of embedding dimension and Maximum word length

Table 7: Performance metrics for multi-modal with attention classification

	Multi-modal with attention (Word2Vec)			**Multi-modal with attention (GloVe)**		
	P	**R**	**F**	**P**	**R**	**F**
Aggressive	0.79	0.65	0.71	0.92	0.68	0.78
Happy	0.64	0.85	0.73	0.67	0.76	0.71
Relaxed	0.77	0.71	0.74	0.73	0.79	0.76
Sad	0.78	0.74	0.76	0.80	0.82	0.81
Average	0.75	0.74	0.73	0.78	0.76	0.77

Table 8: Performance metrics for uni-modal transformer based classification

	BERT			**XLNet**		
	P	**R**	**F**	**P**	**R**	**F**
Aggressive	0.90	0.76	0.83	0.88	0.65	0.75
Happy	0.56	0.68	0.61	0.57	0.76	0.65
Relaxed	0.43	0.29	0.35	0.48	0.38	0.43
Sad	0.47	0.59	0.52	0.45	0.50	0.47
Average	0.59	0.58	0.58	0.59	0.57	0.57

hybrid model is 76.47%. Hence, attention networks play an essential role in improving the performance of classification tasks. Next, we analyze the performance of transformer-based models. As mentioned before, Transformers employed multi-head attention; therefore they catch semantic relationships between input vectors in multiple ways because the heads work parallelly. Performance metrics of the BERT-based model and XLNet-based model are shown in Table 8. The overall accuracy of the BERT-based and XLNet-based models is 58.08% and 57.25%, respectively. The average precision, recall, and F1 score of the BERT-based model is 0.59, 0.58, and 0.58, and that of the XLNet-based model is 0.59, 0.57, and 0.57. The high performance is obtained for class aggressive in the BERT model. While class Aggressive has the highest precision and F1 score for the XLNet-based model, class Happy has the highest recall value.

We also analyze the performance of transformer-based multi-modal systems. Table 9 shows the performance of the CNN+BERT-based multi-modal system and Bi-GRU+BERT-based multi-modal system. The average precision, recall, and F1 score of the CNN+BERT model are 0.73, 0.71, and 0.71. While the Bi-GRU+BERT-based model acquired an average precision value of 0.76, an average recall value of 0.74, and an average F1 score of 0.74. The overall accuracy of the CNN+BERT model is 71.31%, and that of the Bi-GRU+BERT model is 73.5%. For both models, the highest precision and F1 score are for class aggressive and class sad, respectively. The highest recall value is obtained for relaxed in the CNN+BERT model and sad in the Bi-GRU+BERT model.

Table 10 shows the performance of our proposed model. The overall accuracy is increased to 77.94%. The average precision, recall, and F1 score obtained are 0.78. The highest precision, recall, and F1 score of 0.94, 0.88, and 0.91 are reported for the class aggressive. Class happy also obtained better results compared to other systems. The confusion matrices of four models, audio-based, textual-based, multi-modal-based, and proposed transformer-based models, are given in Figure 5.

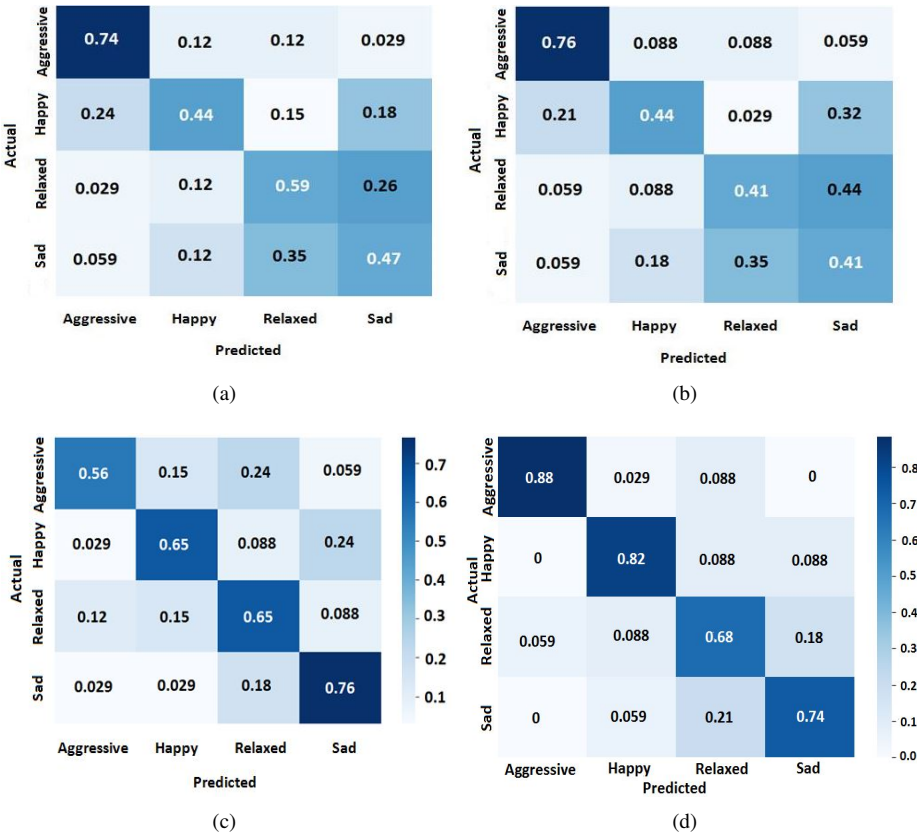

Figure 5: Confusion matrix for classification system,(a) audio-based, (b) lyrics(GloVe) based, (c) multi-modal, (d) proposed transformer-based

Table 9: Performance metrics for multi-modal transformer-based classification

	CNN+BERT			Bi-GRU+BERT		
	P	**R**	**F**	**P**	**R**	**F**
Aggressive	0.88	0.65	0.75	0.92	0.65	0.76
Happy	0.64	0.62	0.63	0.61	0.74	0.67
Relaxed	0.63	0.85	0.72	0.65	0.76	0.70
Sad	0.78	0.74	0.76	0.87	0.79	0.83
Average	0.73	0.71	0.71	0.76	0.74	0.74

BERT stacks several levels of attention, each of which makes use of the results of the layer before it. As it progresses through the model's deepest layers, BERT is able to create extremely detailed representations by repeatedly composing word embeddings.

Each attention head develops a different attention pattern since the attention heads do not share parameters. There are 12 layers and 12 heads in the $BERT_{BASE}$, for $12 \times 12 = 144$ different attention processes. Using the BertViz library [34], we visually observed the attention weights of BERT model. Figure 6 shows the visualization of attention for all heads. Each cell in the diagram displays the attention pattern for a specific head (depicted by a column) in a specific

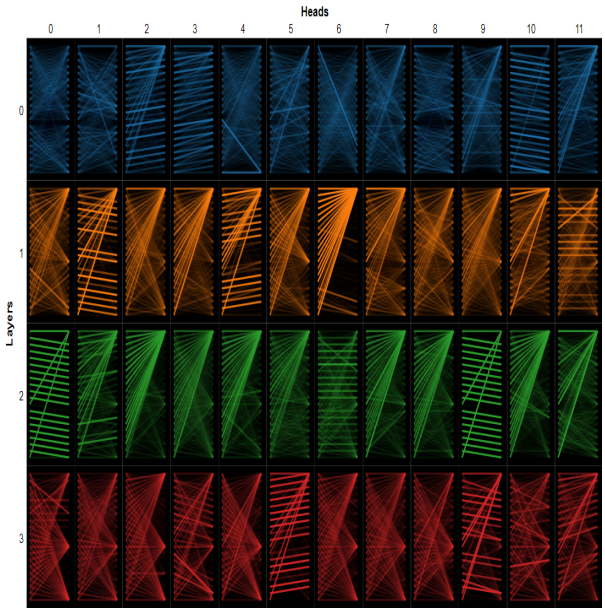

Figure 6: Attention visualization of the first four layers for two lines of a song "that all changed into lies that drop like acid rain" and "you washed away the best of me"

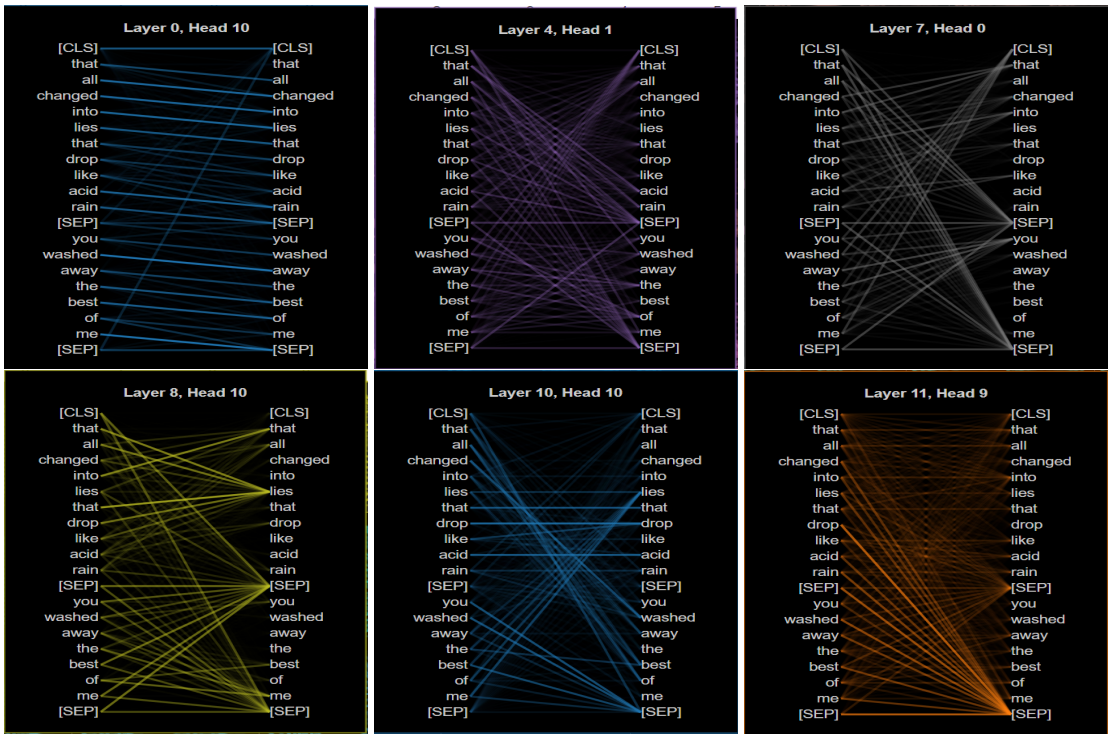

Figure 7: Attention variation of two sentences "that all changed into lies that drop like acid rain" and "you washed away the best of me" among various layers and heads. The lines show the attention from each token (left) to every other token (right). Darker lines indicate higher attention weights.

Table 10: Performance metrics for proposed multi-modal transformer based classification

Class	**Precision**	**Recall**	**F1 score**
Aggressive	0.94	0.88	0.91
Happy	0.82	0.82	0.82
Relaxed	0.64	0.68	0.66
Sad	0.74	0.74	0.74
Average	0.78	0.78	0.78

layer (indicated by row). The attention patterns are specific to the input text. As seen in the illustration, BERT results in various attention patterns.

Figure 7 accurately displays two sentences' attention variations across various layers and heads. In this figure, attention visualizes as lines connecting the word being updated (left) with the word being attended to (right). Weights near one show very dark lines, whereas weights close to zero show faint lines or are not visible at all. Colour intensity indicates attention weight. The [SEP] symbols are unique separator tokens denoting a sentence boundary, and the [CLS] symbol is added to the front of the input and used for classification tasks.

Figure 8 shows the graphical representation of the systems' precision, recall, and F1 score. The x-axis shows the models M1, M2,..etc, which means model 1, model 2,...etc. Y-axis represents performance matrices precision, recall, and F1 score. The figures show that our proposed model performs better among the 13 deep learning models discussed. The overall accuracies provided by music mood classification systems are given in table 11, and it is visually represented in Figure 9. The Table and Figure show the gradual increase in the performance of the transformer-based multi-modal music mood classification system.

In all our tests, a significantly higher number of songs from Q1 and Q2 were correctly classified when compared to Q3 and Q4. This seems to indicate that emotions with higher arousal are easier to differentiate with the selected features. Out of the two, Q2 obtained the highest F1-Score. This goes in the same direction as the results obtained in [35], and might be explained by the fact that several excerpts from Q2 belong to the heavy-metal genre, which has very distinctive, noise-like, acoustic features.

To reaffirm the importance of the proposed approach, the classifiers are compared using the widely used statistical test, namely, McNemar's statistical hypothesis test [36]. This test was adopted as per the findings of Dietterich in [37]. The skill measure adopted for comparing the models is classification accuracy. The contingency table is constructed based on the success(1)/failure(0) measure of the two models being compared. It is of the form,

$$\begin{bmatrix} n11 & n01 \\ n10 & n00 \end{bmatrix}$$

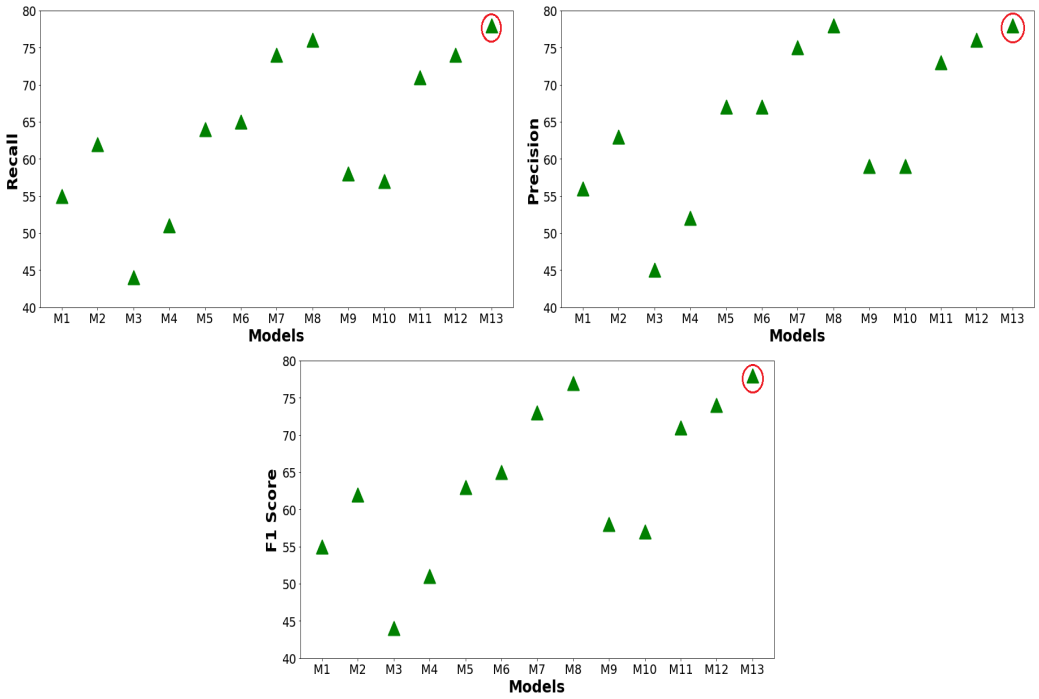

Figure 8: Recall, precision, F1 score of 13 (M1-M13) music mood classification systems. Red circled points indicate the performance of our model.

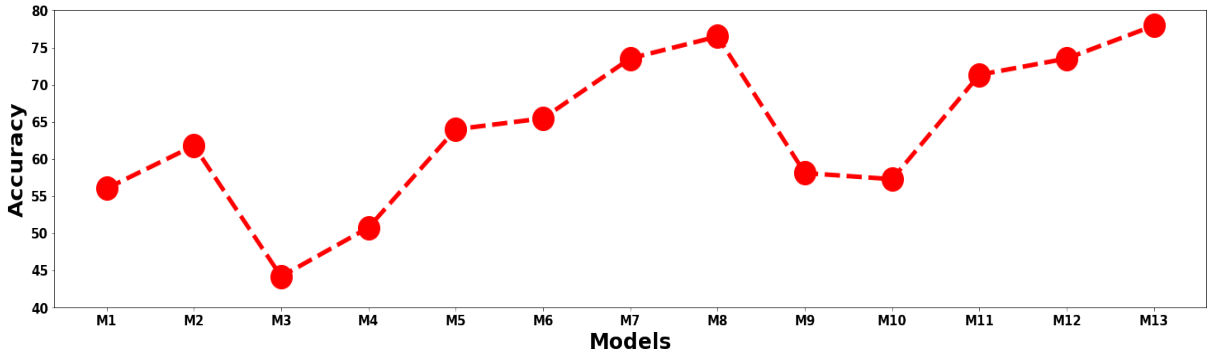

Figure 9: Comparison of overall accuracy of 13 (M1-M13) music mood classification systems

where $n11$ indicates the count of the moods of the songs that were correctly classified by both the models and $n10$ indicates the count of the moods correctly classified by model 1 but misclassified by model 2. Similarly, the other two counts, $n01$ and $n00$ are defined. Thus the total number of samples in the test set would be the sum of these, as $n = n00 + n01 + n10 + n11$. When doing the statistical hypothesis test, the null hypothesis($H0$) is defined as the condition $n01 = n10$, that is the two models have the same error rate or the same proportion of misclassifications. McNemar's test checks for the marginal homogeneity in the contingency table by testing if there is a significant difference between the counts $n01$ and $n10$. This is done using the test statistic t, defined in [36] to include the continuity correction term -1 in the numerator as,

$$t = \frac{(|n01 - n10| - 1)^2}{(n01 + n10)} \tag{7}$$

This test statistic has a Chi-Squared distribution with 1 degree of freedom, and if $H0$ is accepted, then the probability that $t > \chi^2_{1,0.95} = 3.841459$ is less than $\alpha = 0.05$. This test is implemented in Python using the mcnemar() function of the Statsmodels module

The p-value calculated from t statistics is compared with an alpha value to make the final decision as

- $p > \alpha$: fail to reject $H0$, both models have a similar proportion of errors on the test dataset.
- $p \leq \alpha$: reject $H0$, there is a significant difference in the proportion of errors, indicating one is better than the other.

Table 11: Overall accuracy of fourteen systems of the experiment.

Model	**Scheme**	**Overall Accuracy (in%)**
M0	Acoustic+Textual features (SVM)	59.55
M1	Acoustic features (Bi-GRU)	56.00
M2	Acoustic features (CNN)	61.76
M3	Textual features (Word2Vec+Bi-GRU)	44.15
M4	Textual features (GloVe+Bi-GRU)	50.73
M5	Multi-modal fusion (Word2Vec+Bi-GRU)	63.97
M6	Multi-modal fusion (GloVe+Bi-GRU)	65.44
M7	Multi-modal fusion with single attention (Word2Vec+Bi-GRU)	73.52
M8	Multi-modal fusion with single attention (GloVe+Bi-GRU)	76.47
M9	BERT	58.08
M10	XLNet [27]	57.25
M11	CNN+BERT [28]	71.32
M12	Bi-GRU+BERT	73.50
M13	**Proposed multi-modal transformer-based**	**77.94**

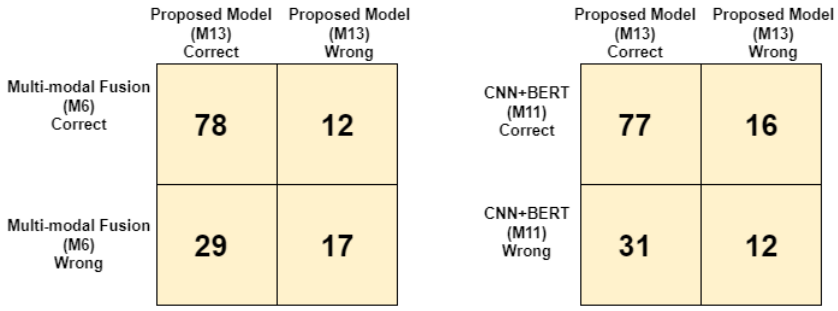

Figure 10: Contingency tables given by the proposed model (M13) against the multi-modal fusion(M6) model(left) and the CNN+BERT(M11) model(right)

The contingency tables obtained from the McNemar test done on the proposed model against the Multi-modal fusion (M6) and the CNN+BERT model (M11) are shown in the left and right figures in Fig. 10, respectively. We can find the difference in the proportions of the errors by looking at the values corresponding to $n01$ and $n10$. A large difference is visible, which indicates the effectiveness of using the proposed model against the baseline systems. On calculating the test statistics, t=6.244 and t=4.170 were obtained, respectively, which resulted in p-values of 0.012 and 0.041. Hence $H0$ is rejected in both cases on taking $\alpha = 0.05$, which proves that the margins of accuracy score gained by the proposed system are statistically significant.

4 Conclusion

In this research, we proposed a multi-modal music mood classification system based on transformers that outperformed all other state-of-the-art methods. Acoustic features are extracted from songs, and the model is trained using Bi-GRU and CNN to obtain a testing accuracy of 56.00% and 61.76%. Lyrics features extracted using Word2Vec and GloVe and trained using the Bi-GRU model to provide an accuracy of 44.15% and 50.73%, respectively. Multi-modal architecture considering the combinations of audio with lyrics-Word2vec and audio with lyrics-Glove are concatenated and trained using the Bi-GRU model. An accuracy of 63.97% and 65.44% is obtained for this model. Then, we integrated an attention mechanism into this multi-modal architecture, and the accuracy increased to 73.52% and 76.47%, respectively. Later, we implemented various transformer models and proposed a multi-modal transformer-based music mood classification system. Our proposed model outperformed all models with an overall accuracy of 77.94%. By analyzing the results, we can conclude that multi-modal yield better results than uni-modals. A multi-modal architecture that incorporates the attention mechanism improves the system. Furthermore, multi-head attention-based transformer multi-modal architecture achieved the highest accuracy. Multi-modal architecture implementing different transformer variants, considering different data sets, etc, can be done as future works.

Competing interests

The authors have declared that no competing interests exist.

Funding

No funding is availed for the proposed work.

Authors' contribution

SS wrote the program, conducted the experiments and wrote initial manuscript. RR conceived the idea, analyzed the results, and corrected the manuscript; All authors read and approved the final manuscript.

References

[1] M. Schedl, E. Gómez, J. Urbano, *et al.*, "Music information retrieval: Recent developments and applications," *Foundations and Trends® in Information Retrieval*, vol. 8, no. 2-3, pp. 127–261, 2014.

[2] G. C. Bruner, "Music, mood, and marketing," *Journal of marketing*, vol. 54, no. 4, pp. 94–104, 1990.

[3] D. Liu, L. Lu, and H.-J. Zhang, "Automatic mood detection from acoustic music data," in *4th Int. Conf. Music Information Retrieval (ISMIR'03)*, pp. 13–17, Johns Hopkins University, 2003.

[4] L. Lu, D. Liu, and H.-J. Zhang, "Automatic mood detection and tracking of music audio signals," *IEEE Transactions on audio, speech, and language processing*, vol. 14, no. 1, pp. 5–18, 2005.

[5] M. Hemalatha, N. Sasirekha, S. Easwari, and N. Nagasaranya, "An empirical model for clustering and classification of instrumental music using machine learning technique," in *2010 IEEE International Conference on Computational Intelligence and Computing Research*, pp. 1–7, IEEE, 2010.

[6] M. Van Zaanen and P. Kanters, "Automatic mood classification using tf* idf based on lyrics.," in *ISMIR*, vol. 9, pp. 75–80, 2010.

[7] R. Mayer, R. Neumayer, and A. Rauber, "Combination of audio and lyrics features for genre classification in digital audio collections," in *Proceedings of the 16th ACM international conference on Multimedia*, pp. 159–168, 2008.

[8] C. Laurier, J. Grivolla, and P. Herrera, "Multimodal music mood classification using audio and lyrics," in *2008 seventh international conference on machine learning and applications*, pp. 688–693, IEEE, 2008.

[9] Y. Zhao, D. Yang, and X. Chen, "Multi-modal music mood classification using co-training," in *2010 International Conference on Computational Intelligence and Software Engineering*, pp. 1–4, IEEE, 2010.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[11] M. Sundermeyer, R. Schlüter, and H. Ney, "Lstm neural networks for language modeling," in *Thirteenth annual conference of the international speech communication association*, 2012.

[12] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, 2014.

[13] J. Abdillah, I. Asror, Y. F. A. Wibowo, *et al.*, "Emotion classification of song lyrics using bidirectional lstm method with glove word representation weighting," *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, vol. 4, no. 4, pp. 723–729, 2020.

[14] R. Rajan, J. Antony, R. A. Joseph, J. M. Thomas, *et al.*, "Audio-mood classification using acoustic-textual feature fusion," in *2021 Fourth International Conference on Microelectronics, Signals & Systems (ICMSS)*, pp. 1–6, IEEE, 2021.

[15] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 4945–4949, IEEE, 2016.

[16] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *Conference on Empirical Methods in Natural Language Processing*, pp. 1412–1421, 2015.

[17] A. Dipani, G. Iyer, and V. Baths, "Recognizing music mood and theme using convolutional neural networks and attention.," in *MediaEval*, 2020.

[18] H. Lu, H. Zhang, and A. Nayak, "A deep neural network for audio classification with a classifier attention mechanism," *arXiv preprint arXiv:2006.09815*, 2020.

[19] Q. H. Nguyen, T. T. Do, T. B. Chu, L. V. Trinh, D. H. Nguyen, C. V. Phan, T. A. Phan, D. V. Doan, H. N. Pham, B. P. Nguyen, *et al.*, "Music genre classification using residual attention network," in *2019 International Conference on System Science and Engineering (ICSSE)*, pp. 115–119, IEEE, 2019.

[20] S. Yu, Y. Yu, X. Chen, and W. Li, "Hanme: hierarchical attention network for singing melody extraction," *IEEE Signal Processing Letters*, vol. 28, pp. 1006–1010, 2021.

[21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *North American Association for Computational Linguistics (NAACL)*, 2018.

[22] W.-C. Chang, H.-F. Yu, K. Zhong, Y. Yang, and I. Dhillon, "X-bert: extreme multi-label text classification with using bidirectional encoder representations from transformers," in *NeurIPS Science Meets Engineering of Deep Learning Workshop*, 2019.

[23] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[24] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," *Advances in neural information processing systems*, vol. 32, 2019.

[25] Y. Gong, Y.-A. Chung, and J. Glass, "Ast: Audio spectrogram transformer," *Interspeech 2021*, pp. 571–575, 2021.

[26] H. Zhao, C. Zhang, B. Zhu, Z. Ma, and K. Zhang, "S3t: Self-supervised pre-training with swin transformer for

music classification," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 606–610, IEEE, 2022.

[27] Y. Agrawal, R. G. R. Shanker, and V. Alluri, "Transformer-based approach towards music emotion recognition from lyrics," in *European Conference on Information Retrieval*, pp. 167–175, Springer, 2021.

[28] K. Pyrovolakis, P. Tzouveli, and G. Stamou, "Multimodal song mood detection with deep learning," *Sensors*, vol. 22, no. 3, p. 1065, 2022.

[29] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, vol. 8, pp. 18–25, Citeseer, 2015.

[30] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[31] W. James, F. Burkhardt, F. Bowers, and I. K. Skrupskelis, *The principles of psychology*, vol. 1. Macmillan London, 1890.

[32] E. Çano and M. Morisio, "Moodylyrics: A sentiment annotated lyrics dataset," in *Proceedings of the 2017 International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence*, pp. 118–124, 2017.

[33] J. A. Russell, "A circumplex model of affect.," *Journal of personality and social psychology*, vol. 39, no. 6, pp. 1161–, 1980.

[34] J. Vig, "Bertviz: A tool for visualizing multihead self-attention in the bert model," in *ICLR workshop: Debugging machine learning models*, 2019.

[35] G. R. Shafron and M. P. Karno, "Heavy metal music and emotional dysphoria among listeners.," *Psychology of Popular Media Culture*, vol. 2, no. 2, p. 74, 2013.

[36] B. S. Everitt, *The analysis of contingency tables*. CRC Press, 1992.

[37] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural computation*, vol. 10, no. 7, pp. 1895–1923, 1998.

Citation: S. A. Suresh Kumar and R. Rajan. *Transformer-based Automatic Music Mood Classi-fication Using Multi-modal Framework*. Journal of Computer Science & Technology, vol. 23, no. 1, pp. 18–34, 2023.
DOI: 10.24215/16666038.23.e02
Received: August 18, 2022 **Accepted:** February 2, 2023.