Research on Music Genre Classification Algorithm Based on Deep Learning

Liu Ying
1024041002
Nanjing University of Posts and Telecommunications
Nanjing, China

Abstract—**Music genre classification is a crucial task in the field of music information retrieval and serves as an essential step in music recommendation and automatic labeling systems. It plays a significant role in helping users find their favorite music. Understanding music genre classification is vital for gaining insights into human preferences for music, and it has been widely applied in building music recommendation systems. In this paper, we propose using a long short-term memory (LSTM) model for music genre classification, as an alternative to convolutional neural networks. By training a deep model, we aim to classify music into ten distinct genres. Additionally, a hierarchical classification approach is employed to further enhance the accuracy. Initially, the LSTM classifier is trained to distinguish between two primary categories: fortissimo and weak tone classes. Then, the music is further categorized into multiple subclasses.**

Index Terms—**Music information retrieval, Music genre classification, LSTM.**

I. Introduction

At the beginning of the 21st century, the world is witnessing the rapid growth of online music data, and the integration of the Internet into daily life has greatly facilitated the expansion of online music content. Efficient and accurate automatic music information processing—especially in terms of access and retrieval—has become an increasingly important topic, attracting growing attention. Music can be classified based on its genre, which typically follows a hierarchical structure. Music genres in online databases are key organizational elements for managing large volumes of music data. These genres are assigned by both human experts and users, a process that can be time-consuming and costly. Defining precise musical genres is challenging, and current genre classification is predominantly done manually, with many musical pieces straddling the boundaries between genres. This complexity arises because music is a constantly evolving art form, with composers and performers often drawing influence from various genres. However, it is recognized that audio signals (whether digital or analog) within the same genre tend to share certain characteristics, such as similar instrumentation, rhythmic patterns, and pitch distributions, suggesting that automatic genre classification is indeed feasible.

Given the time and labor required to organize music collections, manual genre classification has become increasingly costly, necessitating more efficient tools to browse, organize, and dynamically update these collections. Consequently, the automatic classification of music genres has become essential[4]. The availability of automatic music genre classification has a positive impact on the development of music recommendation systems and automated playlists, which are part of the broader field of music information retrieval. Music information retrieval plays a crucial role in digital audio processing, search, and retrieval.

Automatic music genre classification involves the computational analysis of musical signals to assign them to specific genres based on musical feature representations. It is a fundamental component of music information retrieval systems. In this paper, the genre classification process is divided into two stages: feature extraction and multi-class classification. During the feature extraction stage, relevant information is derived from the music signals. Feature extraction must be comprehensive (capturing the essence of the music), compact (minimizing data storage requirements), and efficient (avoiding excessive computational complexity). To meet the first criterion, the extracted features should encompass both low-level and high-level aspects of the music. In the second stage, a mechanism (whether an algorithm or mathematical model) is employed to map the extracted features to their corresponding musical genre labels.

II. Related Works

This section reviews relevant work in the field of music information retrieval, with a focus on automatic music genre classification systems. With the increasing storage capacity and widespread use of digital audio music, collecting large volumes of music data from the Internet has become easier. Music information, such as genre, emotion, and style, is used to manage metadata for extensive music libraries[1]. Aucouturier and Pachet[2] argue that musical genres can provide useful descriptions of musical content, as certain properties (e.g., timbre, beat, rhythm, and instruments) influence music style. However, people tend to perceive music holistically rather than identifying individual sources, attributes, or isolated elements[3]. This creates a gap between how humans perceive music and how computers analyze it. Additionally, constructing consistent and scalable corpora for classification tasks remains a challenge, as it is difficult to provide precise definitions for each genre[4]. Therefore, automatic genre classification continues to be a complex problem.

The application of machine learning-based models for automatic genre labeling has opened up new possibilities, yielding promising results. Feature extraction is a common and important technique for predicting the true label of any audio file. Many models, such as deep neural networks (DNNs)[5], have been combined with other models like decision trees[6] and maximum posterior probability models[7] for music genre classification. DNNs have proven to be effective at processing and training large datasets. Similarly, other neural networks, such as convolutional neural networks (CNNs), use spectrogram features to predict the class of a given audio file, achieving prediction results that closely match the true genre. Although music theory defines multiple genres and sub-genres, drawing clear boundaries between them remains difficult. Therefore, further approaches are needed to categorize large volumes of audio files accurately.

Most classification methods so far have been based on supervised learning with some unsupervised samples[8]. Supervised learning requires time-consuming manual labeling, while unsupervised learning often suffers from poor performance and accuracy. Popular supervised algorithms include support vector machines (SVM), nearest neighbors (NN), Gaussian mixture models (GMM), and linear discriminant analysis (LDA). Recently, semi-supervised classification methods have gained popularity in the machine learning community, as they provide good accuracy with minimal labeling work. However, the field is still relatively new, with limited work on automatic genre classification using semi-supervised learning. One notable approach is by Song et al., who proposed a content-based classification of music genres by combining various musical characteristics for feature fusion[8].

Recurrent neural networks (RNNs), a type of deep learning model, have been widely used for sequential data and are capable of learning temporal relationships[10]. However, due to issues like gradient disappearance and gradient explosion, RNNs struggle to learn long-term dependencies. The introduction of long short-term memory (LSTM) networks and gated recurrent units (GRUs) as variants of RNNs has addressed these problems. This paper employs LSTM for music genre classification tasks, as LSTM is better suited for capturing long-term dependencies, making it ideal for music signal processing. By adjusting the time-based gradient backpropagation, LSTM can mitigate the problems of gradient disappearance and explosion.

III. Problem Statement

Feature extraction is a critical component of any classification system, and various features are often employed in music genre classification to create descriptive representations tailored to specific pattern recognition tasks. For audio signal classification, key dimensions of music, such as timbre, harmony, spatial position, melody, and rhythm, are typically considered. These characteristics are extracted from the audio signals and serve as the foundation for classification. Once the relevant features are extracted, standard classifiers can be applied to categorize the music into the appropriate genres.

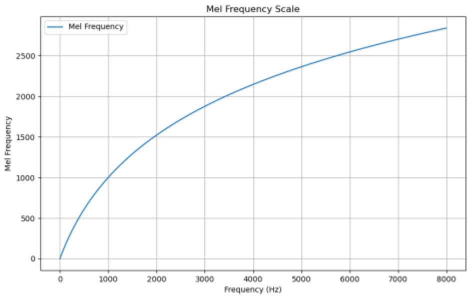


Fig. 1. Mapping between MEL frequency and actual frequency.

The spectrum envelope derived from the fast Fourier transform (FFT) of a signal can be utilized to categorize the amplitude traits of different music genres. By examining amplitude characteristics, this study aims to pinpoint signal variations, noise levels, loudness, and various other spectral attributes that define the discrete-time signals employed in automatic music genre classification. Typically, frequency spectrum features are extracted using the short-time Fourier transform (STFT). However, human auditory perception is nonlinear, as depicted in Figure 1, which illustrates the mapping between Mel frequency and actual frequency. In essence, human auditory perception aligns more closely with Mel frequency perception. Consequently, this paper employs Mel frequency scaling to derive the Mel spectrum when analyzing spectral characteristics. Timbre texture features are crucial for differentiating mixed audio signals that might share identical or similar rhythmic and pitch content. The methodology for utilizing these features originates from speech recognition, where the process begins by segmenting the sound signal into statistically stationary frames. This is commonly achieved through windowed operations at regular time intervals, using a window function such as a Hamming window to mitigate the edge effect. Subsequently, the time structure features of each frame are calculated, and statistical measures (like mean and variance) are derived for these features.

Spectral variability is a measure of the dispersion within data variability, indicating how closely signals cluster together or how widely they are dispersed. This dispersion can be quantified by calculating the standard deviation of the signal's amplitude spectrum. Mel frequency cepstrum coefficients (MFCCs) are the individual components that collectively constitute the Mel frequency cepstrum representation.

Many musical pieces exhibit consistent rhythmic patterns, which contribute to the perception of rhythm. To grasp the essence of music for genre classification, it is essential to comprehend and maintain rhythm as a defining characteristic. This section of the paper introduces a rhythm detection framework tailored for categorizing music genres. Specifically, the beat histogram is highlighted as the key feature vector for this purpose.

Energy is a core feature in speech and audio analysis, reflecting the intensity of a signal by summing the squares of

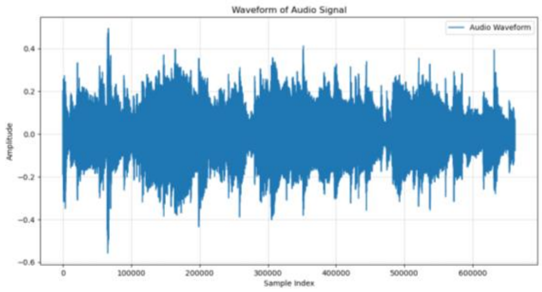


Fig. 2. Beat diagram of Jazz music.

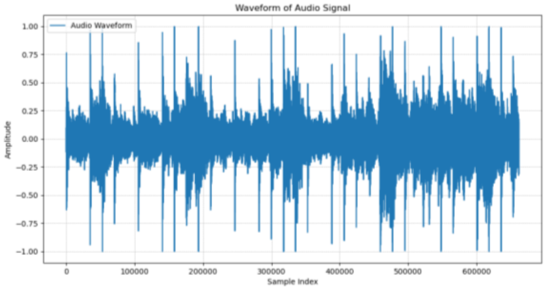


Fig. 3. Beat diagram of rock music.

its discrete-time components. As depicted in Figure 2-5, the distribution of energy across different beats varies with time segmentation. By assessing the energy amplitude within beats, we can extract beat characteristics. This involves examining the arithmetic mean of the energy in the first n windows of the signal (in this paper, n is set to 100 for the experiments) and determining the proportion of windows with energy levels below this mean. The paper also discusses calculating the percentage of silence in the signal as a measure of low-energy segments. A beat histogram represents the range of signal strength across rhythmic intervals, which is derived by measuring the energy in n consecutive windows and performing a fast Fourier transform. This type of feature extraction can result in large matrices, making it computationally intensive but also a rich source of information for rhythmic analysis.

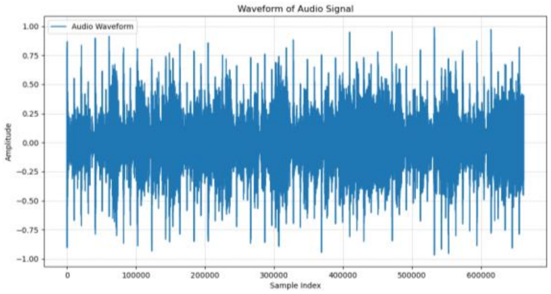


Fig. 4. Beat diagram of disco music.

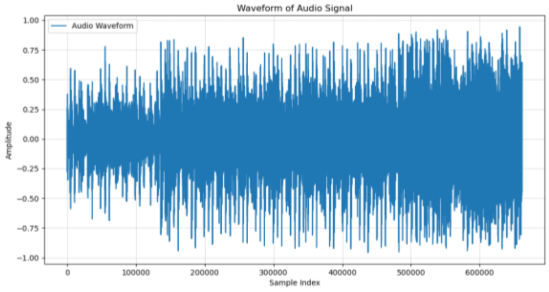


Fig. 5. Beat diagram of pop music.

Rhythm is characterized by repeated patterns of sound and the time of silence measured from the beat. It is this feature that can comprehensively describe the speed characteristics of music. Here is the related description of rhythm characteristics proposed by Pampalk in his paper[11].

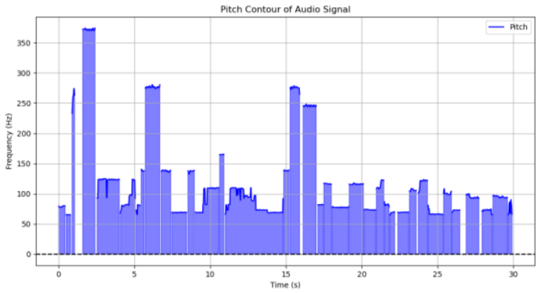


Fig. 6. Beat pitch visualization of rock music.

Pitch content characteristics capture the melodic and harmonic essence of musical signals, which can be revealed through diverse pitch detection methodologies. By determining the principal peaks within the autocorrelation function, one can construct a pitch histogram. This histogram is formulated by aggregating the envelope sums of each frequency band, which are derived from the signal's spectral decomposition. Subsequently, pitch features are distilled from this histogram. Key pitch content features often encompass: the magnitude and frequency of the histogram's most dominant peak, the gap between the two most conspicuous peaks, and the cumulative total of the histogram's values, among others.

Spectral feature extraction plays a pivotal role in genre detection, innovatively merging the distinct attributes of musical pieces with their underlying chord structures and progressions. In music theory, melodies and harmonies are often viewed as the horizontal and vertical dimensions, respectively. A melody unfolds through a sequence of note events, while harmony arises from the concurrent sounding of pitches and the deployment of diverse chords. Chroma[12], a term referring to the 12 semitones in an octave, is instrumental in characterizing both melody and harmony, offering a comprehensive representation of pitch data. It encapsulates a joint time-pitch domain, delivering intensity readings associated with each semitone across an octave. Collectively, all octaves are depicted as a grid that outlines the harmonic essence of a musical composition.

The prevalent approach to Chroma feature extraction, as proposed by Ellis, involves aligning the Chroma vector with the beat for a standardized beat-length representation. Given the variability in song durations, this method selects a fixed number of Chroma vectors—specifically 30—for each track, either at random or in sequence. These vectors are then concatenated in a column-wise fashion to form a robust 360-dimensional feature vector. This vector is particularly advantageous as it is minimally affected by the instruments used and the rhythmic structure of the piece. Each Chroma vector captures information about the prevailing note and its affiliation with a specific chord, providing a nuanced insight into the music's harmonic and melodic content.

No individual feature vector can solely achieve superior classification performance, as each music genre possesses unique musical traits. Consequently, it is essential to combine multiple feature vectors. This paper employs two primary approaches for such combinations. The first approach involves applying weighted distance measures to each feature vector, while the second utilizes a majority voting scheme during the classifier's output phase. This section delves into the feature set used in the proposed music genre classification system. The feature set comprises amplitude-based features that capture timbral aspects, such as loudness, noise, and compactness. It also includes a method for rhythm analysis that leverages the tempo characteristics of the signals. Furthermore, an algorithm is presented that describes the tonal qualities of a musical signal based on its pitch attributes. Lastly, the paper examines Chroma features in the context of chord progression, highlighting their significance in representing the harmonic environment of a musical piece.

IV. Algorithms

The Long Short-term Memory Model (LSTM) is an artificial recurrent neural network (RNN) architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTMS have feedback connections that allow them to process not only individual data points (such as images), but also entire data sequences (such as voice or video). For example, LSTM is suitable for tasks such as undivided, connected handwriting recognition or speech recognition.

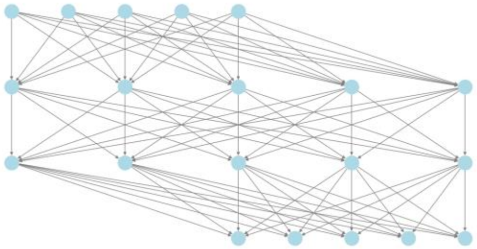


Fig. 7. Deep LSTM network architecture for music genre classification.

A typical Long Short-Term Memory (LSTM) unit is comprised of a memory cell, an input gate, an output gate, and a forget gate. The memory cell is capable of retaining information over arbitrary time spans. The three gates work in concert to control the ingress and egress of information to and from the memory cell. LSTM networks excel in tasks related to classification, processing, and forecasting in the context of time series data, particularly when there are unpredictable gaps between pivotal events within the series. The LSTM architecture was specifically designed to counteract the issues of gradient explosion and vanishing that are commonly faced during the training of conventional Recurrent Neural Networks (RNNs). One of the key advantages of LSTMs over RNNs, Hidden Markov Models (HMMs), and other sequence learning algorithms is their reduced sensitivity to the temporal distances between data points in a sequence.

The deep LSTM network architecture utilized in this study is illustrated in Figure 7. The architecture consists of four layers. The initial layer serves as the input layer, equipped with 13 neuron nodes designed to process the input features. Following this, the two intermediate hidden layers are configured with 128 and 32 nodes respectively, facilitating the extraction and retention of complex temporal patterns within the data. The final layer is the output layer, containing 10 nodes that correspond to the predicted probabilities across ten distinct music genre categories. This structured approach allows the network to effectively handle the nuances of music genre classification by learning from sequential data.

The LSTM network in this paper is trained on a dataset that is divided into ten genre labels, with a total of 600 tracks: 420 for training, 120 for validation, and 60 for testing, each lasting 30 seconds. The training process involves sending 35 samples through the network at a time. The training is conducted over a period of 20 years, during which both accuracy and loss continue to improve. At the 20-year mark, the test accuracy reaches its peak, and the loss is at its lowest. However, the classification accuracy in this paper is moderate, ranging from 0.5 to 0.6, indicating room for improvement. It is suggested that with a larger labeled training dataset, the accuracy could potentially increase to between 0.6 and 0.7. The current limitations are attributed to the small size of the labeled training data, which leads to low accuracy and overfitting. While some genres, such as metal, are distinct and easily identifiable, others that are similar pose a greater challenge for categorization.

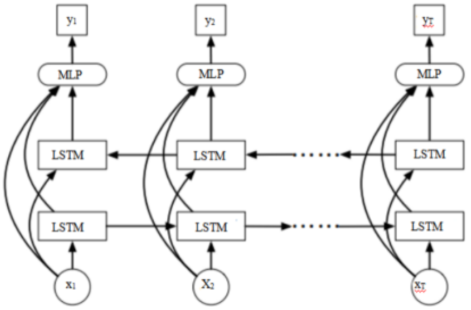


Fig. 8. Multi-layer LSTM classification network topology.

The LSTM classifier designed in this chapter for music genre recognition features a specific architecture, as detailed in Figure 8. It includes two recurrent neural network layers, with the first layer utilizing gated recurrent units for learning temporal relationships within the features. The network is structured to handle time series aggregation features effectively. After regularization, the first layer accepts input of size (1x96x1400), and the features are pooled using a maximum pooling of size (2x2), reducing the time dimension to 720 and the frequency to 48, resulting in an output size of (63x48x720) for the second hidden layer. A second layer with size (122x3x3) followed by a maximum pool of size (3x3) produces an output size of (8x16x240). The third layer, with an output size of (128x4x60), feeds into the fourth and final layer (128x3x3), which, after pooling, provides an output size of (128x1x15). This output is further processed to be compatible with RNN networks.

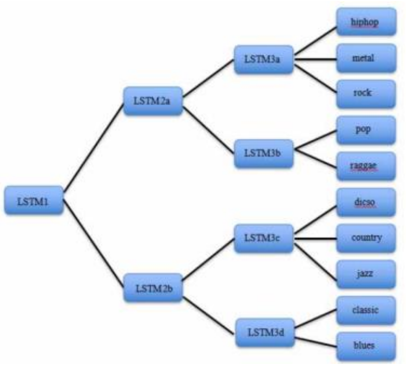


Fig. 9. The hierarchy of the LSTMs in the multi-step classifier.

When analyzing audio files across different genres, similarities in waveforms are observed. Genres like hip hop, metal, pop, rock, and reggae exhibit similar low and high-frequency energy in their spectrograms, while jazz, disco, country, classical, and blues spectrograms share more similarities. A divide-and-conquer strategy is considered for further classification. The LSTM classifiers used are listed for reference.

LSTM1: It categorizes music into two categories: hard (hiphop, metal, pop, rock and reggae swing) and light (jazz, disco, country, classical and blues).

LSTM2a: It divides music into sub-forte 1(hip hop, metal and rock) and sub-forte 2 (pop and reggae swing).

LSTM2b: It divides music into two categories: secondary Mild 1(disco and country) and secondary Mild 2(jazz, classical and blues).

LSTM3a: It divides music into hip-hop, metal and rock. Similarly, training data were only sampled from hip-hop, metal and rock.

LSTM3b: It is used to distinguish between pop music and reggae swing.

LSTM3c: It is used to distinguish between disco music and country music.

LSTM3d: It is used to distinguish between jazz, classical and blues.

V. Evaluation

The GTZAN database is used for genre classification experiments and contains 1000 recordings covering 10 musical genres. Genres represented in the database include: Classical, country, disco, Hiphope, jazz, rock, blues, reggae, pop, and metal. Figure 10 shows the sample proportion of each category in the GTZAN dataset, where each type has 100 recordings. All recordings are mono, with a sampling rate of 22,050 Hz and a duration of approximately 30 seconds. Each recording is divided into 30 segments, so the duration of each segment is 1 second.

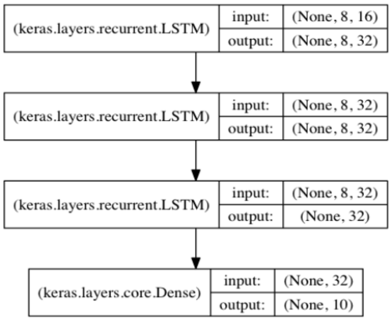


Fig. 10. LSTM-based music genre classifier network structure.

The GTZAN dataset, compiled by Tzanetakis, is a well-established collection used for benchmarking in music information retrieval tasks. In the preprocessing phase, all stereo MP3 audio files are transformed into mono waveform files with a uniform sampling rate of 16 kHz. This dataset encompasses ten distinct music genres: blues, classical, country, disco, hip hop, jazz, metal, pop, reggae, and rock. Each genre is represented by 100 audio tracks, each 30 seconds in duration, in the .au format. In this study, a random selection process is employed to allocate samples from the GTZAN dataset for both training and testing purposes, ensuring that the test and training datasets are mutually exclusive. The dataset is organized into ten music genre folders, with nine folders dedicated to training and the remaining one used for testing. To enhance the robustness of the results, a ten-fold cross-validation approach is applied, and the average accuracy across these folds is reported as the final test accuracy. The LSTM network is trained with specific hyperparameters: a dropout rate of 0.5 to prevent overfitting and a learning rate of 1e-5 to facilitate stable convergence. Training is terminated and the model is saved once the precision curve reaches a stable convergence area, indicating that further training would not yield significant improvements in performance. This methodical approach to training and validation aims to achieve a reliable and generalizable model for music genre classification.

To assess the effectiveness of the proposed music genre classification method and to benchmark it against conventional approaches, this paper undertakes a classification task using the GTZAN dataset. The dataset comprises 1000 music clips, each 30 seconds long, sampled at 22050 Hz. Each clip is categorized into one of ten genres, with an equal representation of 100 clips per genre.

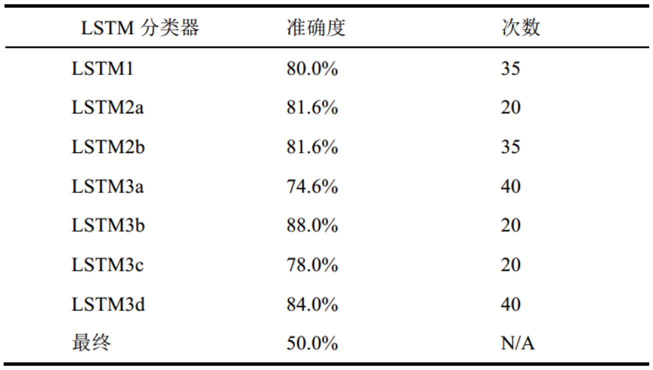
The experimental music genre classifier is constructed with an LSTM-based topology, as depicted in Figure 10. This architecture includes three LSTM layers. The input layer's dimension matches the dimensionality of the input feature vector. The middle LSTM layer consists of 32 neurons, and the final layer is a fully connected layer with 10 nodes, employing softmax activation for classification across the ten music genres. Given the limited size of the dataset in this experiment, it's important to note that LSTM, a deep learning model that typically requires extensive data to achieve robust learning, may face challenges. Therefore, special considerations and potentially data augmentation techniques should be applied prior to training to enhance the model's performance. For training, validation, and testing, the dataset is split in an 8:1:1 ratio across different music genres to ensure a representative sample for each genre. To rigorously evaluate the model's performance and classification accuracy, the dataset is further divided into ten parts for cross-validation. This approach helps ensure the model evaluation's accuracy and fairness while minimizing the impact of outliers on the classification results.

During model training, the Adam optimization algorithm is employed, which adapts the learning rate. The learning rate is initialized at 0.002 and decreases linearly to 0, facilitating a more nuanced training process. A batch size of 64 is chosen, meaning that for each training iteration, 64 samples are selected from the training set, with each sample being trained once before the dataset is randomly shuffled again.It's acknowledged that fitting issues can arise during the training of deep learning models, such as overfitting or underfitting. As the model complexity increases, the training error is expected to decrease with continued training. However, it's crucial to monitor the validation error to ensure that the model generalizes well to unseen data and to avoid overfitting to the training set.

Early termination is a strategy used in training machine learning models, including neural networks, to prevent overfitting. It involves monitoring the model's performance on a validation set after each training epoch, which is a complete pass through the entire training dataset. The training is halted prematurely if the validation accuracy ceases to improve or even declines, indicating that the model is starting to overfit to the training data rather than generalizing well to new, unseen data.

In the context of this experiment, early termination is likely employed to ensure that the LSTM model does not overfit to the training data, especially given the relatively small dataset size. The goal is to stop training when the model's ability to generalize to the validation set no longer improves.

The experiment also explores the classifier's ability to distinguish between six different schools of music, which could refer to genres or styles. The network's architecture is detailed in Figure 11, which is not provided here but presumably outlines the number of nodes and their activation functions in each layer. The input layer has 13 nodes, which correspond to the dimensionality of the input feature vector, likely Mel-frequency cepstral coefficients (MFCCs). The first hidden layer consists of 128 nodes using the sigmoid activation function, the second hidden layer has 32 sigmoid neurons, and the output layer has 6 nodes, which might be intended for a classification task with six categories.



LSTM 分类器	准确度	次数
LSTM1	80.0%	35
LSTM2a	81.6%	20
LSTM2b	81.6%	35
LSTM3a	74.6%	40
LSTM3b	88.0%	20
LSTM3c	78.0%	20
LSTM3d	84.0%	40
最终	50.0%	N/A

Fig. 11. Network's architecture.

It's important to note that the number of output nodes should match the number of classes in the classification task. If there are indeed six schools or genres to classify, the output layer should have six nodes, each representing a class, and typically with a softmax activation function to output probabilities for each class. If the task involves ten music genres, as mentioned in previous responses, the output layer should be adjusted to have ten nodes instead.

VI. Conclusion

As the Internet and multimedia devices have become ubiquitous, the proliferation of digital music on various platforms has been immense. This surge in music availability has made manual organization and categorization a daunting task. Consequently, there is a growing need for an efficient system that can manage vast music databases and enable users to swiftly and precisely locate music that aligns with their preferences.

In the realm of music classification, conventional methods often rely on prior knowledge and involve intricate feature extraction processes, which can result in features that are not broadly applicable. Deep learning-based classification models offer a solution by automating the music classification process through training.Music signals, inherently time series data, can be addressed by classification models that function as encoding-decoding frameworks. During encoding, the input data is condensed into a fixed vector representation, a process that inevitably leads to some information loss. Music information retrieval systems can decipher the context within audio signals and categorize them based on attributes such as genre, instrument, and mood, thus enhancing the user experience by providing a more refined selection of music options.The more precisely the music's inherent labels are identified, the better the system can cater to users' preferences, allowing for smarter music selection. Deep feature extraction and machine learning techniques can offer pre-labeled retrieval mechanisms that comprehend nearly all the content within a musical piece, from local to temporal features. The synthesis of features using deep networks represents an exploratory system for labeled classification.

This discussion centers on music genre classification, encompassing the entire process from audio feature extraction and classifier training to the development of a comprehensive automatic music genre recognition system. Instead of using convolutional neural networks, this paper employs a long short-term memory (LSTM) model for music genre classification. By training a deep learning model, music can be accurately sorted into ten distinct genres. Furthermore, the paper introduces a hierarchical classification approach to enhance accuracy. Initially, music is categorized into strong (fortissimo) and soft classes using an LSTM classifier, followed by further subdivision into multiple subclasses. This innovative multi-stage classification method is applied to categorize music into different genres at each stage, and experiments indicate that this layered approach can significantly boost classification accuracy.

References

[1] N. Scaringella, G. Zoia, and D. Mlynek, "Automatic genre classification of music content: a survey," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 133-141, 2006.

[2] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293-302, 2002.

[3] T. Lidy and A. Rauber, "Evaluation of feature extractors and psycho-acoustic transformations for music genre classification," in *Proc. 6th Int. Conf. Music Inf. Retr. (ISMIR)*, 2005, pp. 34-41.

[4] T. Li and M. Ogihara, "Music genre classification with taxonomy," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, vol. 5, 2005, pp. V/197-V/200.

[5] B. K. Baniya, D. Ghimire, and J. Lee, "Automatic music genre classification using timbral texture and rhythmic content features," in *Proc. 17th Int. Conf. Adv. Commun. Technol. (ICACT)*, 2015, pp. 434-443.

[6] S. Dieleman and B. Schrauwen, "End-to-end learning for music audio," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2014, pp. 6964-6968.

[7] G. Bengolea et al., "Feature Analysis for Audio Classification," in *Progress in Pattern Recognit., Image Anal., Comput. Vis.*, 2014, pp. 239-246.

[8] N. Agera, S. Chapaneri, and D. Jayaswal, "Exploring Textural Features for Automatic Music Genre Classification," in *Proc. Int. Conf. Comput., Commun., Control Autom. (ICCUBEA)*, 2015, pp. 822-826.

[9] K. Markov and T. Matsui, "Music genre classification using Gaussian Process models," in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process. (MLSP)*, 2013, pp. 1-6.

[10] J.-P. Xu et al., "Speech emotion recognition based on feature selection and extreme learning machine decision tree," *Neurocomputing*, vol. 273, pp. 271-280, 2018.

[11] J. Andén and S. Mallat, "Deep Scattering Spectrum," *IEEE Transactions on Signal Processing*, vol. 62, no. 16, pp. 4114-4128, 2014.

[12] G. Song, Z. Wang, F. Han, S. Ding, and M. A. Iqbal, "Music AutoTagging Using Deep Recurrent Neural Networks," *Neurocomputing*, vol. 291, pp. 187-197, 2018.