Exploration of Speech and Music Information for Movie Genre Classification

MRINMOY BHATTACHARJEE, Indian Institute of Technology Guwahati, India
S. R. MAHADEVA PRASANNA, Indian Institute of Technology Dharwad, India and Indian Institute of Information Technology Dharwad, India
PRITHWIJIT GUHA, Indian Institute of Technology Guwahati, India

Movie genre prediction from trailers is mostly attempted in a multi-modal manner. However, the characteristics of movie trailer audio indicate that this modality alone might be highly effective in genre prediction. Movie trailer audio predominantly consists of speech and music signals in isolation or overlapping conditions. This work hypothesizes that the genre labels of movie trailers might relate to the composition of their audio component. In this regard, speech-music confidence sequences for the trailer audio are used as a feature. In addition, two other features previously proposed for discriminating speech-music are also adopted in the current task. This work proposes a time and channel Attention Convolutional Neural Network (ACNN) classifier for the genre classification task. The convolutional layers in ACNN learn the spatial relationships in the input features. The time and channel attention layers learn to focus on crucial time steps and CNN kernel outputs, respectively. The Moviescope dataset is used to perform the experiments, and two audio-based baseline methods are employed to benchmark this work. The proposed feature set with the ACNN classifier improves the genre classification performance over the baselines. Moreover, decent generalization performance is obtained for genre prediction of movies with different cultural influences (EmoGDB).

CCS Concepts: • **Information systems** → **Speech / audio search**; **Recommender systems**; Summarization; Clustering and classification; • **Computing methodologies** → *Feature selection*; *Neural networks.*

Additional Key Words and Phrases: Movie trailer genre classification, Speech-music classification, Spectral peak tracking, Attention

ACM Reference Format:
Mrinmoy Bhattacharjee, S. R. Mahadeva Prasanna, and Prithwijit Guha. 2024. Exploration of Speech and Music Information for Movie Genre Classification. *ACM Trans. Multimedia Comput. Commun. Appl.* 0, 0, Article 0 (2024), 20 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Automatic movie recommendation is an important application that lets viewers easily find their content of interest. Different approaches have been explored for providing movie recommendations to users. Some methods exploit knowledge of public sentiment from microblogging site data to generate recommendations [1]. Few other approaches use the preference of viewers for specific actors or directors to provide suggestions [2]. Nevertheless, movie recommendation based on preferred genres is arguably the most popular approach [2]. Genre labeling is a complicated task

Authors' Contact Information: Mrinmoy Bhattacharjee, mrinmoy.bhattacharjee@alumni.iitg.ac.in, Indian Institute of Technology Guwahati, Guwahati, Assam, India; S. R. Mahadeva Prasanna, prasanna@iitdh.ac.in, Indian Institute of Technology Dharwad, Dharwad, Karnataka, India and Indian Institute of Information Technology Dharwad, Dharwad, Karnataka, India; Prithwijit Guha, pguha@iitg.ac.in, Indian Institute of Technology Guwahati, Guwahati, Assam, India.

ACM 1551-6865/2024/0-ART0
https://doi.org/XXXXXXX.XXXXXXX

since movies can belong to multiple genres. Manual labeling may lead to inaccurate or incomplete labels due to the subjective bias of the annotators. Therefore, efficient automatic determination of movie genres is essential for providing valuable recommendations.

Genre classification of short movie trailers (≈ 3 minutes) has attracted many researchers. Movie trailers are designed in a particular manner so that different emotional responses may be evoked in the viewers [3]. Trailers usually contain rich and varied content that represents the theme of the actual movie. An automatic tool to efficiently determine the probable genres in a movie from its trailer can be beneficial. Viewers might make use of this information in deciding what movie to watch. On the other hand, content creators and distributors might utilize this information for targeted publicity.

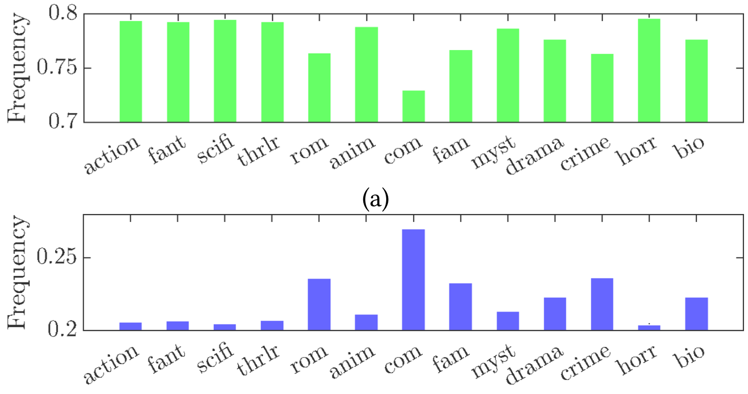
1.1 Related work

Researchers have explored various approaches to perform the task of movie genre classification in the past. One of the pioneering works was performed by Rasheed et al. [4] to classify movie previews into multiple hierarchical genres (ACTION and NON-ACTION movies) using audio-visual features. They used the peakiness in the audio energy plot as one modality. In [5], they followed up by performing movie preview or trailer classification into COMEDY, ACTION, DRAMA or HORROR genres using only low-level visual features and mean-shift clustering. Wang et al. [6] proposed the use of psychology and cinematographic information for affective understanding of movies to bridge the semantic gap between low-level audio-visual features and high-level emotions. Jain et ak. [7] used early-fusion of audio features like pitch, frequency domain energy and Mel-Frequency Cepstral Coefficients (MFCC), and visual features with a feed-forward neural network to classify movies into ACTION, HORROR, COMEDY, MUSIC and DRAMA genres. Austin et al. [8] proposed to categorize movies into ROMANCE, DRAMA, HORROR, and ACTION genres using their musical scores. The authors used timbral (MFCC, LPC, ZCR, and other standard spectral features) and rhythm (tempo, beat) features with SVM classifiers to perform pair-wise and four-class classification of genres. Giannakopoulos et al. [9, 10] performed Violent Scene Detection (VSD) by fusing audio-visual features. They used 12 standard audio features including MFCC for the task. They performed an early fusion of audio-based and video-based probability vectors with a k-Nearest Neighbour classifier to determine violent scenes. Irie et al. [11] performed affective scene classification using audio-visual features that included pitch, short-term energy, and MFCC. Souza et al. [12] performed VSD using local spatiotemporal features with a bag of visual words and a linear SVM classifier. Bag of words features refer to a representation where the order of elements is ignored, and the focus is on the frequency of individual elements within a given dataset. Zhou et al. [13] performed movie trailer genre classification by using category information of all shots in a trailer as a bag of visual words features to map the trailers into ACTION, COMEDY, DRAMA or HORROR genres. Chen et al. [14] performed VSD by first detecting an action scene, and then categorizing it as a horror scene using face, blood, and motion information. Wang et al. [15, 16] performed horror scene detection using Multiple Instance Learning with visual and aural features that included MFCC, power, spectral centroid, and Zero-Crossing Rate (ZCR) for each scene computed in the form of a bag of shots. Huang et al. [17] performed movie genre classification using an ensemble of one-vs-one Radial Basis Function kernel SVM classifiers with a combination of audio features like spectrum compactness, root-mean-square energy, ZCR, linear prediction coefficients, MFCC, and rhythm, along with visual features. Acar et al. [18] showed that mid-level audio features in the form of a bag of audio words of MFCC features performed better than low-level audio and visual features in VSD.

More recently, Simoes et al. [19] performed movie genre classification using MFCC and visual features with a Convolutional Neural Network (CNN) classifier. Authors observed that audio features improve the performance of all genres, especially the COMEDY genre. Tadimari et al. [20] performed movie genre classification using a linear-SVM classifier with audio-visual features. Wehrmann et al. [21] performed movie genre classification using an ensemble of classifiers trained on different audio-visual features including MFCC. In this work and their subsequent works [22, 23], authors observed that the detection of HORROR genre is significantly improved with the addition of audio information. Tarvainen et al. [24] observed that detecting the amount of speech and music in movie audio is very useful for scene detection. Hence, they used music emotion as an additional feature with image features for the acoustic scene classification of movies into categories like interior or exterior, and day or night. Alvarez et al. [25] performed aesthetic style clustering of movies and genre classification and observed that the inclusion of audio features with visual improved genre classification performance. Cascante-Bonilla et al. [26] performed genre classification of movie trailers using multiple modalities of text, video, audio, posters, and meta-data. For the audio modality, log-Mel scaled power spectrograms computed from 30s audio chunks are stacked and passed to a Convolutional Recurrent Neural Network (CRNN) for classification. The various modalities were combined using score-fusion for the final prediction. They also noted that the addition of audio modality significantly improved the overall performance. Chu et al. [27, 28] proposed a genre classification system using movie posters. Shambharkar et al. [29] performed genre classification using $3D$-CNN over stacks of video frames. Mangolin et al. [30] combined information from audio, video frames, posters, subtitles, and synopsis in a late-fusion framework for multi-label genre classification of movie trailers using Binary Relevance and Multi-Label k-NN classifiers. Yadav et al. [3] attempted to classify the genres of movie trailers from Indian cinema. Authors extracted facial frames from the movie trailers and mapped them to various emotions for use as a feature for genre classification. They proposed an Inception-Long Short-Term Memory-based classification system. Fish et al. [31, 32] defined the genre classification task as a weak-labeling method and proposed a multi-label context-gated approach. Audio embeddings used in their method were obtained from a VGG-style network trained for audio classification [33], that are temporally aggregated using a network called NetVLAD [34]. Authors observed that audio modality performed best in detecting COMEDY and SPORTS genres. Sharma et al. [35] used only the audio modality and followed a bag of audio words-based approach to classify movie trailers into ACTION, ROMANCE, HORROR, SCI-FI, and COMEDY genres. Authors observed that ACTION genre is best detected with their proposed method. Vishwakarma et al. [36] performed genre classification of movie trailers by extracting high-level cognitive and affective information obtained from multiple modalities of visual images, dialogues, and movie meta-data.

Based on the above discussion, it is obvious that there is a bias in the existing literature towards using the visual modality in the Movie Trailer Genre Classification (MTGC) task. However, despite the popularity of visual modality in the MTGC task, the audio component has also been useful [23]. The auditory stream is a rich medium for provoking various emotions [6]. The affective characteristic is said to be better captured by audio than video [37]. Some specific sounds and music are frequently used by movie editors to elicit specific emotional responses and to promote dramatic effects [15]. Audio information has also been found to aid in better detection of violent scenes [38]. Music used in movies of high-intensity genres like ACTION and HORROR has very distinct characteristics from those with softer emotional expressions, like DRAMA and ROMANCE [8].

Speech, music, and sound effects are the audio types frequently found in almost all movie scenes [39]. Researchers have observed that speech and music are more beneficial in predicting movie genres. Tarvainen et al. [24] mentioned that the prominence of speech and music alone might be enough to classify scenes in movie audio. Dialogues and environmental sounds are assumed to lack genre-specific information [31]. Wang et al. [6] also observed that environmental sounds are less helpful than speech and music in identifying emotions. In addition, sound effects or environmental sounds are sparsely distributed. At the same time, speech and music are the most



(a)

(b)

Fig. 1. Illustrating average genre-wise distribution of music (a) and speech (b) probability sequences computed for the movie trailers in the *Moviescope* dataset. These sequences are obtained by employing a speech vs. music classifier to predict the class-wise probabilities of consecutive non-overlapping 1s intervals in the trailer audio of movies.

significant components of movie trailers. Therefore, this work explores the usefulness of information about the most prominent audio types in movie trailers (speech and music) for genre classification. Genre-wise music and speech signals distribution in the *Moviescope* dataset is illustrated in Fig. 1. It can be observed from the figure that the frequency of speech and music varies across genres. For example, Comedy and Romance genres are associated with more speech and less music. The opposite is observed for Horror and Action. Such observations indicate that just the speech and music information might be adequate for movie genre classification.

1.2 Motivation

This work explores the audio modality of movie trailers for performing genre classification and analyzes its possibilities. Motivation for this work is derived from the fact that different audio types are naturally suited to provoke varied emotions [6]. Thus, the auditory stream of movie trailers has the potential to be a rich source of information for performing the task of movie genre classification. There are a few works that have performed audio classification as an intermediate step of MTGC using only basic audio features like Mel Frequency Cepstral Coefficients (MFCC) [9]. However, such features can be expected to perform poorly in the presence of noise and background music commonly found in movie audio. To the best of the authors' knowledge, previous literature in MTGC lacks any serious attention devoted to studying the complexities of movie audio modality. Therefore, this work attempts to provide a detailed and dedicated study of the movie audio modality so that it can aid in developing better MTGC frameworks in the future. Authors believe that basic audio features may not be sufficient to perform a complicated task like MTGC. Wang et al. [37] noted that it is not possible to find a direct relationship between human-level interpretations like movie genre and basic audio features like MFCC. Therefore, learning genre-specific information directly from the basic features may be difficult or unreliable. Nonetheless, these basic features may be leveraged to derive intermediate representations that may be better equipped to map the relationship between movie trailer audio and their corresponding genre labels. Such intermediate representations can be trained to inherently capture the audio signal-type information. Hence, this work proposes such an intermediate audio-based representation to be used in MTGC.

Additionally, since speech and music are more prominently found in movies [24], the movie trailer audio can be diarized into consecutive speech and music segments using the prediction scores of a speech-music discriminator. Such diarization sequences can also provide vital information about the underlying genre classes of the movie trailer. Thus, this work proposes for the first time to use confidence score sequences of speech and music segments in the movie trailer audio as a feature for training the MTGC classifier. Even though few previous works in MTGC have performed speech-music classification using basic features and classifiers, those methods are not easily generalizable to the diversity present in movie audio. This work proposes to use a sophisticated speech-music detection system that can tackle the complex audio scapes of movie trailers. It is hypothesized that more confident signal-type predictions will aid in developing better features and classifiers. The speech-music score sequences are generated using a recent classification method [40] that performed exceedingly well. The current work also employs various statistics and learned representations obtained from the spectral peak sequences of audio signals as additional features. Moreover, the attention mechanism has been recently shown to be a better way of modeling the long-term temporal evolution of signals [41]. However, attention-based aggregation of audio features has not been explored previously in the MTGC literature. Hence, this work also explores for the first time an approach for attention-based audio-feature aggregation for MTGC. It is to be noted that this work does not propose that only audio-based MTGC can be better than multi-modal approaches. In a complicated task like MTGC, multi-modal approaches will always be necessary to obtain satisfactory genre classification performance. The aim of this work is only to improve the audio component of a generic multi-modal MTGC system.

The main contributions of this work are summarized below.

- **First**, this work proposes speech-music confidence score sequences of movie trailer audio as a feature for MTGC for the first time (subsection 2.1).
- **Second**, unlike previous approaches that have simply used standard audio features like Mel frequency cepstral coefficients, this work proposes to use learned representations. Such features are expected to capture the information about audio types present in the underlying signal, which might benefit the MTGC task.
- **Third**, this work proposes attention-based sequence modeling and aggregation of the audio signal (speech and music) features for the first time in the MTGC task (section 3).

Rest of the paper includes a description of the proposed features in subsection 2.1, subsection 2.2 and 2.3, a description of the proposed classifier in subsection 3, discussion on experiments in section 4, and conclusion in section 5.

2 PROPOSED APPROACH

This work proposes to extract human-level information about speech and music from movie trailer audio for performing MTGC. Information about the audio signal type in movie trailers might aid in mapping features from the underlying audio signals with information of human interpretation, like movie genre. This work proposes to use speech-music confidence score sequences of movie trailer audio as a feature. In addition, a learned representation derived using spectral peak tracking [40] is used as another feature. Statistical measures computed from spectral peak tracks of trailer audio are also used as a feature. The feature extraction procedure is described in the following subsections.

2.1 Speech-Music Predictions

As discussed previously, speech and music are the most prominent components of the movie trailer audio. Hence, Speech and Music Confidence (SM-Conf) scores can be extracted for sound units in the movie trailer audio sequences. These scores in the form of a time series would represent many

Table 1. Comparative illustration of speech vs. music classification performance on the *MUSAN* dataset using x-vector feature based X-SMC system and CBoW feature based C-SMC system. Results are reported as "mean (μ) $F1$-score (over 3-fold cross-validation) ± standard deviation (σ)". The $F1$ scores are expressed in percentage.

		Music			**Speech**			
Features	**Acc**	**Prec**	**Rec**	**F1**	**Prec**	**Rec**	**F1**	**Avg. F1**
X-SMC	98.88 ±0.18	98.71 ±0.56	98.57 ±0.16	98.64 ±0.25	98.99 ±0.14	99.10 ±0.35	99.04 ±0.13	**98.84** **±0.19**
C-SMC	96.60 ±0.21	96.16 ±0.92	95.58 ±0.72	95.86 ±0.37	96.90 ±0.50	97.32 ±0.57	97.11 ±0.13	96.49 ±0.25

vital details about the distribution of speech and music, their switching rate, and overall proportion in the trailer. This work hypothesizes that such SM-Conf sequences might have genre-specific characteristics. Therefore, such information might help identify the genre labels of movie trailers. Thus, the SM-Conf sequences of movie trailer audio are used as a feature in the current task.

The SM-Conf scores can be obtained using trained classifiers that predict how much a sound unit resembles speech or music. Mirbeygi et al. [42] have reviewed various Speech vs. Music Classification (SMC) and separation methods previously proposed in the literature. This work explores two recently proposed feature sets for the SMC task, viz., x-vectors [43] and Component Bag-of-Words (CBoW) [40] features. Each of the SMC systems is described in the subsequent paragraphs.

2.1.1 X-vector based SMC. The first feature used in this work for SMC is the x-vector [43]. The x-vector was initially proposed for the speaker recognition task. Subsequently, researchers found that the x-vectors were useful in tasks other than speaker recognition as well [44, 45, 46, 47]. The x-vectors are extracted as embeddings from a DNN model trained for speaker recognition. In this work, the trained model provided by the SpeechBrain toolkit [48] is used to extract 512-dimensional x-vectors. For further details regarding x-vector computation, the reader is encouraged to refer to the original paper [43]. The *MUSAN* dataset [49] is used to train the x-vector SMC system (X-SMC). The x-vectors for consecutive 1s segments of speech and music signal from the *MUSAN* dataset are used to train a DNN for the SMC task. The DNN architecture is the same as the one used in [40]. The SMC training is performed in a three-fold cross-validation format. The speech and music signals are split into three non-overlapping folds. At each iteration, one fold is used as the testing set, while the other two are combined for training the model. The three-fold mean and standard deviation of the performance of the X-SMC system is illustrated in Table 1.

2.1.2 CBoW-ASPT-LSPT based SMC. The second feature used for SMC in this work is the CBoW feature proposed in [40]. The CBoW features are computed from the spectral peak amplitude or the location information. It was shown that the combination of amplitude and location information provided the best SMC performance. Accordingly, 200-dimensional CBoW features are used to train an SMC system (C-SMC). The CBoW features are computed for consecutive 1s segments of speech and music signals from the *MUSAN* dataset. A brief description of computing the CBoW features is provided in subsection 2.2. For further details, the reader is encouraged to refer to the original proposal [40]. The three-fold mean and standard deviation of the performance of the C-SMC system is illustrated in Table 1.

2.1.3 *SMC training and performance.*

This work employs the Deep Neural Network (DNN) classifier similar to the proposal in [40] for the SMC task. The DNN used for the SMC task has four hidden layers. The number of neurons in each layer is calculated as two times, two-thirds, half, and one-third of the feature dimension, respectively. For example, if the feature dimension is 60, then the hidden-layer sizes starting from the input side would be 120, 40, 30, and 20, respectively. The output of each hidden layer is passed through a *ReLU* activation, followed by *Batch Normalization.* The layer outputs are subjected to a *Dropout* factor of 0.4 during training to act as a regularization and avoid overfitting. The output layer has two neurons with *SoftMax* activation. The *SoftMax* layer predicts the likelihood of the input sample being eiher speech or music. The network is optimized using an *Adam* optimizer with an initial learning rate of 10^{-4}. The X-SMC system is trained with an input feature dimension of 512, whereas the C-SMC system uses an input feature of 200-dimensions. Performances of the two systems on the *MUSAN* dataset illustrated in Table 1 indicates that the X-SMC system performs better than C-SMC. It must be noted that the DNN architecture design implies that the C-SMC system has approximately half the number of parameters as that of the X-SMC system. Therefore, the C-SMC performs comparably to the X-SMC system with fewer parameters. It may be further argued that the C-SMC might be less prone to overfitting the *MUSAN* dataset than the X-SMC system because of the smaller model size.

This work uses the trained X-SMC and C-SMC systems to predict the SM-Conf score sequence for the movie trailer audio. It may be noted that the whole trailer audio is passed through an SMC system to obtain the SM-Conf sequences. Any non-speech and non-music sound present in the trailer gets a confidence score for either speech or music. The sounds that differ significantly from speech and music are believed to receive a low confidence score. Nevertheless, all confidence scores are retained for performing the movie genre classification task. Since the SMC systems are trained on a different dataset (*MUSAN*), a median filter of kernel width 5 is used to suppress the prediction noise in the SM-Conf score sequence. The sequence of smoothed SM-Conf scores is directly passed through attention layers (see subsection 3) to model the relationship of speech and music signals with the genre of a movie trailer. The learned feature representations used in this work are discussed next.

2.2 Learned representations

Existing movie genre classification works have primarily used audio features computed directly from the signal. Subsequently, such features will be referred to as *raw features* unless mentioned otherwise. *Raw features* measure typical characteristics of the underlying signal, like energy or zero-crossing rate. Discriminative models may be used to learn the category-specific characteristics of a particular classification task. The classification performance varies depending on the discriminability of a particular *raw feature.* It is generally observed that *raw features* tend to be perturbed when noise is added to the signals. Hence, learning an intermediate representation from the *raw features* that are not directly affected by the signal noise might be helpful.

Learned representations might capture critical properties of the underlying signal that may not be evident from the *raw features.* For example, the previously mentioned SM-Conf feature is also a form of learned representation that indicates the likelihood of a 1s audio interval being speech or music. However, the SM-Conf feature is a one-dimensional representation that may only capture some of the variabilities involved in a complicated task like movie genre prediction. A higher dimensional learned representation is required that can better capture information about concepts like speech and music. In addition, the multi-dimensional nature of the feature would help it retain sufficient movie genre information. The CBoW features proposed in [40] fit the above description and are adopted in this work. The CBoW features were shown to capture the striation pattern information of the signals under consideration. Curvy and linear striations characterize the speech and music signals, respectively [40]. The distinct patterns in these two signal types are learned for extracting the CBoW features. This intermediate representation is believed to aid in movie genre prediction since different aspects of speech and music signals carry genre-specific cues. The following paragraphs describe the CBoW feature in brief.

Let, $\mathbf{x}[n]$ ($n = 1, \dots N_s$) be a movie trailer audio of N_s samples. Also, let $X[k][t]$ ($k = 1 \dots n_b$, $t = 1 \dots T_f$) be its DFT magnitude spectrogram with n_b frequency-bins and T_f short-term frames of size T_w ms with a shift of T_s ms. For each frame spectra in X, n_p prominent spectral peaks are identified. The amplitude and frequency information of the selected spectral peaks are retained in two $n_p \times T_f$ sized matrices $\mathbf{A}$ and $\mathbf{L}$, respectively. The sequence of p^{th} ($p = 1 \dots n_p$) spectral peak amplitude or location across the audio signals is termed peak traces. The peak traces are believed to capture the distinct striation patterns in the time-frequency representation of speech and music signals. The distributions of these peak traces are modeled using univariate Gaussian Mixture Models (GMM), trained separately for speech and music signals.

An m_g-mixture GMM is trained for each p^{th} peak-trace ($p \in [1, n_p]$) across the training set of either speech or music. Thus, a total of n_p GMMs are trained separately for peak-amplitude of music (say $\mathcal{G}_{\text{A},mu}^{p=1\dots n_p}$), peak-location of music (say $\mathcal{G}_{\text{L},mu}^{p=1\dots n_p}$), peak-amplitude of speech (say $\mathcal{G}_{\text{A},sp}^{p=1\dots n_p}$) and peak-location of speech (say $\mathcal{G}_{\text{L},sp}^{p=1\dots n_p}$). Subsequently, for the amplitude or location of every n_p prominent peak in an audio frame, m_g posterior probabilities are obtained separately from the speech and music GMMs. Thus, a $2 \cdot n_p \cdot m_g$-dimensional feature vector $\mathcal{V}_A$ is obtained for each short-term frame by concatenating the posterior probabilities from music and speech peak amplitude GMMs. Similarly, a $2 \cdot n_p \cdot m_g$-dimensional feature vector $\mathcal{V}_L$ is obtained by concatenating the posterior probabilities from music and speech peak-location GMMs. Finally, $\mathcal{V}_A$ and $\mathcal{V}_L$ feature vectors are averaged over 1s segments. This step smooths the fluctuations introduced by the possible presence of non-speech and non-music signals in movie trailers. A more detailed description of the feature computation process is provided in [40].

The original proposal in [40] used the MUSAN dataset [49] for feature computation. The movie trailer datasets do not provide annotations for speech and music signals present in the audio. Therefore, this work uses GMMs trained using the MUSAN dataset to compute the $\mathcal{V}_A$ and $\mathcal{V}_L$ features for the movie trailers. Unless mentioned otherwise, these features will be collectively referred to as *GMM-Posterior Features* (GPF).

2.3 Statistical representations of audio segments

In addition to modeling the peak information distribution using GMMs, this work employs various statistical measures of the peak traces. Such measures capture gross information about the variations in the peak trace evolutions within an audio segment. In a previous work of the authors [40], only the mean and standard deviation computed over the spectral peak traces were used for the SMC task. This work extends the approach of [40] and computes twelve different statistical measures over tracks of spectral peak amplitude and location information. These measures are computed over 1s audio segments. The statistical measures computed in this work are maximum, minimum, median, mode, mean, standard deviation, geometric mean, geometric standard deviation, harmonic mean, entropy, skewness, and kurtosis. The 12 measures computed from n_p peak amplitude traces are concatenated to obtain a $12 \cdot n_p$-dimensional feature vector $\mathcal{U}_A$. Similarly, a $12 \cdot n_p$-dimensional feature vector $\mathcal{U}_L$ is obtained from the n_p peak location traces. All the statistical measures computed for the amplitude and location information of the n_p spectral peak tracks are concatenated to form a $24 \cdot n_p$-dimensional feature vector for each 1s audio segment. These features will be collectively referred to as *Statistical-measure Features* (SF). The SF features might be able to capture gross-level characteristics of 1s audio segments and aid the learned representations (GPF) in the movie genre classification task. The following section describes the classifier architectures employed for performing the movie genre classification in this work.

3 CLASSIFIER DESIGN

Researchers have previously used different classifiers in the movie genre prediction task. However, deep neural networks with attention mechanisms have rarely been explored for the MTGC task. Yu et al. [50] have recently shown the usefulness of attention in the MTGC task. However, they employed attention to spatio-temporal features obtained from the visual modality. To the best of our knowledge, the attention mechanism has not yet been used to aggregate audio features for MTGC. This work explores two types of classifiers based on the attention mechanism. First, the application of Transformer architecture [41] and second, the proposal of a variant of the CNN-Attention hybrid model [51]. Various input feature matrices $\mathcal{F}_q$ $(q = 1, \dots n_f)$ of sizes $d_m^{(q)} \times n_t$ are fed to both classifiers (subsection 2.1, subsection 2.2, and subsection 2.3). Here, n_f is the number of input features, $d_m^{(q)}$ indicates feature dimension size of the q^{th} input matrix, and n_t represents the number of consecutive 1s intervals in an audio segment considered as a classification unit. The classifier architectures are described next.

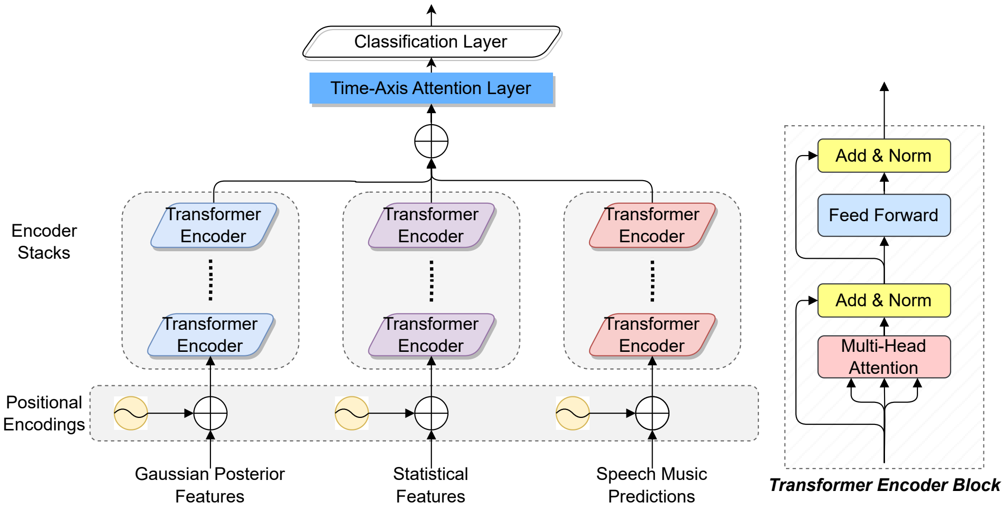


Fig. 2. Illustrating the TENN architecture used in this work.

3.1 Transformer-based architecture

The transformer architecture used in this work is illustrated in Fig. 2. The vanilla transformer architecture [41] was proposed for the machine translation task. Therefore, it had an encoder and a decoder component. This work performs the classification of input features into discrete categories. Hence, the decoder part of the transformer is not used here. The encoder component is kept the same as in the original proposal. The proposed architecture is referred to as Transformer-Encoder based Neural Network (TENN). The input to the encoder block is a tensor of size $d_m^{(q)} \times n_t$, where n_t represents the number of timesteps while $d_m^{(q)}$ indicates the feature dimension. Each layer inside the encoder block of TENN produces an output of the same shape as its input to enable residual connections. A fully-connected layer projects the output of the encoder block to a d_o-dimensional vector. The transformer architecture uses positional encoding to keep track of the order of input embeddings. This work also employs the same method as the original proposal to encode positional information into the features. As discussed previously, multiple features are used in this work. An intermediate-fusion strategy was applied to the different features in the transformer classifier. Each feature has a separate parallel branch of encoder stacks (as shown in Fig. 2). The original proposal of a transformer network established that only attention-based architectures can be equally or even more efficient than traditional neural networks like CNN [41]. The q^{th} attention output $\mathcal{A}^{(q)} = Attention\left(Q^{(q)}, K^{(q)}, V^{(q)}\right)$ of the encoder blocks used in this work is defined by eqn. 1 [41].

$$Attention\left(Q^{(q)}, K^{(q)}, V^{(q)}\right) = SoftMax\left(\frac{Q^{(q)}K^{(q)\top}}{\sqrt{d_q}}\right)V^{(q)} \tag{1}$$

For each of the q^{th} input, the query ($Q^{(q)}$), key ($K^{(q)}$) and value ($V^{(q)}$) matrices are set as $Q^{(q)} = K^{(q)} = V^{(q)} = \mathcal{F}_q$ and $d_q = d_m^{(q)}$. The transformer design also includes Multi-Head Attention (MHA). In single-head attention, the full rank query, key, and value matrices are used to compute the attention outputs. In a n_{heads} MHA, the query, key, and value matrices are projected to n_{heads} separate lower-dimensional sub-spaces. All the projections are parallelly attended. The outputs of these parallel operations ($\mathbf{H}_i$, $i = 1, \ldots n_{heads}$) are concatenated and then projected to obtain the output of size $n_t \times d_o$. The MHA is defined in eqn. 2, and the operation in each attention head is defined in eqn. 3 [41].

$$\mathrm{MHA}(Q^{(q)}, K^{(q)}, V^{(q)}) = concat\left(\left[\mathbf{H}_1, \ldots \mathbf{H}_{n_{\text{heads}}}\right]\right) \cdot W^o \tag{2}$$

$$\mathbf{H}_i = Attention(Q^{(q)\top} \cdot W_{Q,i}^{(q)}, K^{(q)\top} \cdot W_{K,i}^{(q)}, V^{(q)\top} \cdot W_{V,i}^{(q)}) \tag{3}$$

Here, $W_{Q,i}^{(q)}$, $W_{K,i}^{(q)}$ and, $W_{V,i}^{(q)}$ ($i = 1, \ldots n_{heads}$) are $d_m^{(q)} \times d_k$ weight matrices and W^o is a $d_m^{(q)} \times d_o$ linear transformation. This work uses $n_{heads} = 8$, $d_k = \dfrac{d_m^{(q)}}{n_{heads}}$, and $d_o = d_m^{(q)}$ as parameters in the encoder blocks. The output of the encoder stacks for each feature is concatenated and aggregated along the time axis using the attention mechanism (illustrated in Fig. 4). A detailed discussion on aggregation attention is provided in the following subsection (subsection 3.2). The aggregated output of the encoder stacks is then passed to a fully connected layer for genre classification. The output layer has *Sigmoid* activation. The network is trained with a *Binary Cross-entropy* loss and an *Adam* optimizer [52]. The initial learning rate is set to 0.001.

3.2 CNN-Attention hybrid architecture

This work proposes an Attention-based Deep Convolutional Neural Network (ACNN) classifier for the MTGC task. Attention-based aggregation of audio features for the MTGC task has not been previously explored. A block diagram of the proposed architecture is shown in Fig. 3. The ACNN consists of separate convolutional branches for each of the feature inputs $\mathcal{V}_A$, $\mathcal{V}_L$, $\mathcal{U}_A$, and $\mathcal{U}_L$. The *Convolutional and Pooling Layer Block* (CPLB) (see Fig. 3) is designed as a cascade of processing layers. Each CPLB layer consists of a convolution stage with *ReLU* activation followed by a max-pooling (by a factor of 2) along the feature dimension. Successive processing by a cascade of these layers gradually reduces the feature dimension to unity. Information is encoded along the channel dimensions while the feature dimension is pooled. The temporal dimension is not sub-sampled in the CPLB layers.

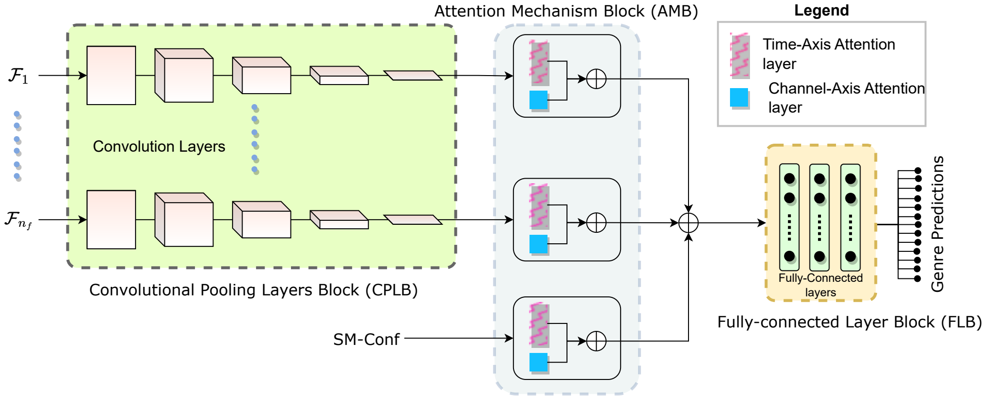

$\mathcal{F}_1$
$\mathcal{F}_{n_f}$
Convolution Layers
Convolutional Pooling Layers Block (CPLB)
Attention Mechanism Block (AMB)
SM-Conf
Legend
Time-Axis Attention layer
Channel-Axis Attention layer
Fully-Connected layers
Fully-connected Layer Block (FLB)
Genre Predictions

Fig. 3. Proposed ACNN classifier architecture for MTGC.

Consider an input $\mathbf{Z}^{(l)}$ of size $d_f \times d_t \times d_c$ to a certain l^{th} CPLB layer $\mathcal{L}_{CPLB}^{(l)}$. Here, d_f, d_t, and d_c refer to the input sizes along the feature, temporal, and channel dimensions. The convolution stage of $\mathcal{L}_{CPLB}^{(l)}$ processes $\mathbf{Z}^{(l)}$ with $n_k^{(l)}$ number of filter kernels of size $3 \times 3 \times d_c$. The negative responses of the convolution operations are suppressed by the *ReLU* activation function. The convolution and *ReLU* activation stages produce $\mathbf{Z}_c^{(l+1)}$ of size $d_f \times d_t \times d_c$. This is subjected to max-pooling along the feature dimension only to produce the output $\mathbf{Z}^{(l+1)}$ of size $\frac{d_f}{2} \times d_t \times d_c$. For the q^{th} CPLB, input to the first layer is $\mathbf{Z}^{(1)} = \mathcal{F}_q$ with input size $d_m^{(q)} \times n_t \times 1$. Each layer in the CPLB contains n_c convolution filter kernels. The number of convolution filters in each CPLB branch of the ACNN classifier is set to $n_c = 80$. A cascade of layers is applied to obtain an output tensor of the CPLB of size $1 \times n_t \times n_c$. This is further treated as a matrix output $\mathbf{C}_q$ of size $n_t \times n_c$.

The proposed *Attention Mechanism Block* (AMB) operates on a matrix input. In the case of speech-music prediction features (SM-Conf), this matrix is of size $n_t \times 2$. For all other features, they are first processed by respective CPLB units, and the corresponding outputs of size $n_t \times n_c$ are processed by AMB. The AMB is used to compute the attention-weighted sum of rows and columns of the input matrix. This provides attended feature vectors along the time and channel dimensions. A description of the AMB is provided next.

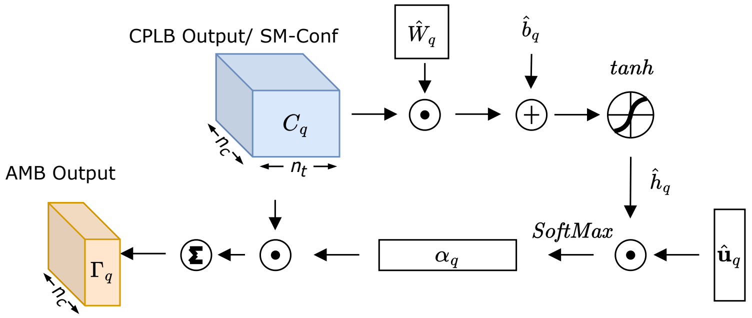


Fig. 4. Block diagram of the attention module for computing time-axis attention output Γ_q for the q^{th} input feature.

Yang et al. [53] proposed an attention mechanism for document classification. This work proposes an attention mechanism that is inspired from [53]. Current work employs attention to collate representations learned by the convolution layers along the time and channel axes (see Fig.4). The time-axis attention emphasizes time steps that are important in the underlying task. On the other hand, channel-axis attention aims to capture the most informative kernels in the last convolutional layer. $\mathbf{C}_q$ is fed to a multi-layer network to perform time-axis attention. A representation $\hat{h}_q$ of size $n_t \times n_c$ is obtained by using a $n_t \times n_t$ weight matrix $\hat{W}_q$ and a $n_c \times 1$ bias vector $\hat{\mathbf{b}}_q$ using eq 4.

$$\hat{h}_q = \tanh\left(\hat{W}_q \mathbf{C}_q \ + \ \mathbb{1} \cdot \hat{\mathbf{b}}_q^{\top}\right) \tag{4}$$

Here, $\hat{\mathbf{b}}_q = \left[b_1 \dots b_{n_c}\right]^{\top}$ and $\mathbb{1} = [1, \dots 1]^{\top}$ is a $n_t \times 1$ vector of all ones. Next, a trainable $n_t \times 1$ weight vector $\hat{\mathbf{u}}$ is used to obtain context-weights for every time step as follows:

$$\alpha_q = \mathit{SoftMax}\left(\hat{h}_q^{\top} \cdot \hat{\mathbf{u}_q}\right) \tag{5}$$

Finally, a n_c-dimensional time-axis attention-weighted context-vector Γ_q is obtained as follows:

$$\Gamma_q = \sum_{j=1}^{n_c} \alpha_q[j] \cdot \mathbf{C}_q[:][j] \tag{6}$$

Proceeding in a similar fashion, a n_t-dimensional channel-axis attention-weighted context-vector Λ_q is obtained by using another attention operation using $\left\{\tilde{W}_q, \tilde{b}_q, \tilde{u}_q\right\}$. Here, $\tilde{u}_q$ is a $n_c \times 1$ weight vector. The channel-axis attention mechanism is performed according to eqn. 7, eqn. 8, and eqn. 9, respectively.

$$\tilde{h}_q \quad = \quad \tanh\left(\tilde{W}_q \mathbf{C}_q \ + \ \mathbb{1} \cdot \tilde{\mathbf{b}}_q^{\top}\right) \tag{7}$$

$$\beta_q \quad = \quad \mathit{SoftMax}\left(\tilde{h}_q^{\top} \cdot \tilde{\mathbf{u}_q}\right) \tag{8}$$

$$\Lambda_q \quad = \quad \sum_{r=1}^{n_t} \beta_q[r] \cdot \mathbf{C}_q[r][:] \tag{9}$$

The context vectors are concatenated to form a $(n_t + n_c)$-dimensional vector Θ_q as in eqn. 10. Finally, Θ_q is fed to Fully-connected Layers Block (FLB) for classification.

$$\Theta q = \begin{bmatrix} \Gamma_q \\ \Lambda_q \end{bmatrix} \tag{10}$$

The time and channel attention output from each feature branch is concatenated and fed through a series of three 300-neuron fully-connected layers to a 13-node output layer that predicts each genre's probability. The output layer has 13 nodes because the *Moviescope* dataset has 13 unique genre labels for all the component movies. Since it is a multi-label classification task, the output nodes have a *Sigmoid* activation and are trained with a *Binary Cross-entropy* loss function. The network is optimized using the *Adam* optimizer [52] with an initial learning rate of 0.001. The outputs of all convolutional layers are passed through *Linear* activation and *Batch Normalization*. Each of the fully-connected layers is followed by *Batch Normalization*, *ReLU* activation, and a *Dropout* factor of 0.1. The hyperparameters of ACNN have been finalized after performing a grid search over various possible values. The following section discusses the experiments performed and their results.

4 EXPERIMENT AND RESULTS

Recently, Cascante-Bonilla et al. [26] published a multi-modal movie trailer dataset called *Moviescope*. This dataset consists of ≈ 5000 trailer videos, plot summaries, posters, and other metadata. Movies in the dataset are labeled with one or more genre labels from a list of 13 possible genres. Since the *Moviescope* dataset is one of the largest datasets publicly available, the proposed MTGC method has been benchmarked on this dataset. Only the audio component from the trailer videos is extracted and used in this work. The audio signals are processed with a sampling rate of $f_s = 16000$Hz, short-term frame size of $T_w = 10$ms with a shift of $T_s = 5$ms (similar to [40]). Hamming window is applied to the short-term frames to suppress windowing effects. The spectrograms have been computed with $n_b = 10^{-3} \cdot T_w \cdot f_s$ frequency bins. Spectral peak tracking is performed with $n_p = 10$ prominent peaks, and peak-trace GMMs are trained with $m_g = 5$ mixtures. Following Cascante-Bonilla et al. [26], a 30s segment is used as input to the system. Thus, the $\mathcal{V}_A$ input to ACNN classifier (see Fig. 3) is a 2-dimensional feature patch of size 30×100, since every $\mathcal{V}_A$ vector represents a 1s segment. Similarly, $\mathcal{V}_L$ input has a size of 30×100 (see Fig. 3). The statistical features $\mathcal{U}_A$ and $\mathcal{U}_L$ are presented to the classifier with an input size of 30×120 each (see Fig. 3). During the training of classifiers, the learning rate is reduced by a factor of 10 if the validation loss does not improve for 5 consecutive epochs. The minimum learning rate allowed in training is 10^{-8}. An early-stopping criterion is also employed whereby the training is stopped if the validation loss does not improve for 10 consecutive epochs. Codes used in this work are shared publicly [1].

The proposed approach involves the use of multiple features for the classification task. In this regard, two different feature fusion strategies have been used in this work. First, features are combined in an intermediate classifier layer, where separate branches in the network learn from different features. Second, a late-fusion strategy defined as a weighted sum of genre prediction scores from separate models trained on different features is employed. The score weights are determined experimentally. Following current literature [26], performances in this work are reported as the area under precision-recall curves ($AU(\overline{PRC})$) for each genre separately. Additionally, three other metrics are reported to assess the performance of the methods in an average sense. First, the mean average precision (**mAP**) across categories is reported that calculates the mean of the binary metrics, giving equal weight to each class. Second, the micro average precision (μ**AP**) computed over all samples across all classess pooled together. Third, the sample average precision (**sAP**) is reported, which does not calculate a per-class measure but instead calculates the metric over the true and predicted classes for each sample in the evaluation data, returning their weighted average.

4.1 Baseline Methods

The performance of the proposed genre classification method has been compared with two recent audio-based approaches from the literature. The proposal of Sharma et al. [35] is used as the *first baseline* ($B1$). It involves a set of 68 standard tempo-spectral audio features and K-Means clustering-based audio segmentation information for the MTGC task. The *second baseline* ($B2$) uses the proposal presented in [26] (authors of *Moviescope* dataset). This work uses the Log-Mel Spectrograms (LMS) of 30s segments as input to a Convolutional Recurrent Neural Network classifier for the MTGC task. The work in $B2$ has used multiple modalities for the MTGC task. However, the performance of this proposal is compared against the audio modality of $B2$ only. In this work, the LMS feature is input as a 128×1407 patch (see [26] for details) to a convolutional branch similar to [26], with 50 kernels in each layer.

[1] https://github.com/mrinmoy-iitg/MTGC_Speech_Music_Segmentation

Table 2. Performances of SM-Conf obtained from X-vectors [43] and CBoW-ASPT-LSPT [40] based speech vs. music classifiers. Results are reported in terms of genre-wise $AU(\overline{PRC})$. Also, macro average precision (**mAP**), micro average precision (μ**AP**) and sample average precision (**sAP**) are reported in the last three columns.

Feature	**action**	**anim**	**bio**	**com**	**crime**	**drama**	**fam**	**fant**	**horr**	**myst**	**rom**	**scifi**	**thrlr**	**mAP**	μ**AP**	**sAP**
X-SM-Conf	24.46	4.49	5.86	39.61	20.80	51.57	12.09	12.82	9.98	13.18	22.41	14.02	28.44	20.25	38.66	59.14
C-SM-Conf	28.05	4.90	8.88	61.35	25.69	64.53	11.28	13.28	15.01	13.05	25.51	16.24	34.21	**25.09**	**42.66**	**59.44**

Table 3. Performance of the transformer based classifier. Here, $B2$ indicates the only audio based results reported by Cascante-Bonilla et al. [26]. Results are reported in terms of genre-wise $AU(\overline{PRC})$. Also, macro average precision (**mAP**), micro average precision (μ**AP**) and sample average precision (**sAP**) are reported in the last three columns.

Feature	**action**	**anim**	**bio**	**com**	**crime**	**drama**	**fam**	**fant**	**horr**	**myst**	**rom**	**scifi**	**thrlr**	**mAP**	μ**AP**	**sAP**
$B2$	56.70	48	11.20	86.20	40	79	49.60	44.70	37.60	22.70	43	27	56.30	46.30	61.40	72.30
1 block	57.24	35.24	9.79	84.38	31.28	74.09	35.28	18.34	32.54	25.6	40.96	25.36	52.62	40.67	58.23	70.96
2 blocks	54.44	21.57	10.23	83.77	30.54	72.52	29.20	14.84	30.54	23.56	41.67	24.39	52.10	38.09	56.87	70.21
3 blocks	49.26	18.09	9.79	82.68	25.42	71.92	26.16	13.94	22.48	22.63	39.15	23.68	50.68	35.51	55.46	69.35
4 blocks	45.29	16.03	11.67	80.80	27.53	72.37	23.41	13.73	25.79	23.81	38.79	21.72	49.28	35.01	54.54	68.62
5 blocks	25.83	3.83	8.17	56.95	19.11	61.23	10	13.48	8.73	9.49	19.82	15.84	32.05	22.11	41.48	59.12
6 blocks	25.76	3.71	8.96	58.41	19.16	62.86	9.65	12.43	12.40	12.23	24.93	15.62	32.08	23.18	41.69	59.17

4.2 Speech-music classification system

As previously discussed, this work explores two established feature sets for the SMC task. Both the SMC systems were employed to obtain the SM-Conf sequences for the trailer audio from the *Moviescope* dataset. Here, X-SM-Conf indicates the confidence scores obtained from the X-SMC system. Similarly, the confidence scores obtained from C-SMC are labeled as C-SM-Conf. The X-SM-Conf and C-SM-Conf are then used to train separate ACNN models to perform the MTGC task. This experiment is performed to identify the more suitable SMC system through the obtained confidence scores. Table 2 presents the MTGC performances obtained using X-SM-Conf and C-SM-Conf as features of the ACNN classifiers. It was observed above that the X-SMC system performs better than C-SMC (see Table 1). Therefore, it may be expected that X-SM-Conf computed using X-SMC might perform better than the C-SM-Conf obtained from C-SMC in the MTGC task. However, it may be observed from Table 2 that the C-SM-Conf provides significantly better performance than X-SM-Conf. The poor performance of X-SM-Conf based MTGC may be due to poor generalizability of the X-SMC system on the *Moviescope* dataset. Since the x-vector features are embeddings extracted from a trained speaker verification model, they might be affected by the diverse set of sound types in movie trailers. On the contrary, the CBoW features are posterior probabilities obtained from GMMs trained on the peak traces of speech and music data. Hence, CBoW features may be less affected by the variety of sounds in movie trailers. Based on these observations, the C-SM-Conf score sequences are used in the rest of the experiments in the paper.

4.3 Performance of Transformer

Table 3 presents the performance of the transformer-based classifier illustrated in Fig. 2 (see subsection 3.1). The performance of this classifier is investigated with varying numbers of encoder

Table 4. Performances of baseline and proposed methods in the MTGC task. Here, $P1$:=GPF+SF+C-SM-Conf (Intermediate-Fusion), $P2$:=$P1$+**LMS** (Intermediate-Fusion), $P3$:=$P1$+**LMS** (Late-Fusion). Results are reported in terms of genre-wise $AU(\overline{PRC})$. Also, macro average precision (**mAP**), micro average precision (**μAP**) and sample average precision (**sAP**) are reported in the last three columns.

Feature	**action**	**anim**	**bio**	**com**	**crime**	**drama**	**fam**	**fant**	**horr**	**myst**	**rom**	**scifi**	**thrlr**	**mAP**	**μAP**	**sAP**
$B1$	43.77	20.83	8.53	76.79	27.91	72.07	26.71	14.86	24.18	20.39	38.55	24.85	46.80	34.74	53.17	68.47
$B2$	56.70	48.00	11.20	86.20	40.00	79.00	49.60	44.70	37.60	22.70	43.00	27.00	56.30	46.30	61.40	72.30
$P1$	58.85	51.14	11.74	85.90	37.52	79.87	46.55	27.35	40.99	32.92	41.53	27.37	60.65	**46.62**	**62.03**	**73.15**
$P2$	63.53	54.62	11.70	87.43	38.89	81.53	53.98	35.13	48.75	29.30	42.97	27.14	62.39	**49.34**	**64.90**	**75.53**
$P3$	37.05	67.40	9.86	86.65	40.79	78.83	61.74	37.05	50.28	24.55	42.17	22.87	60.14	**51.48**	**65.57**	**75.78**

blocks. Table 3 presents the results for one to six blocks in the encoder stack. Only the transformer architecture with one encoder block provides comparable performance with the $B2$ baseline. The performance also gradually drops with an increased number of encoder blocks. Such results indicate that the model might overfit the training dataset because of too many trainable parameters and too few training data. Indeed, the *Moviescope* dataset is not large enough to train deep networks like transformers. Dosovitskiy et al. [54] showed that vanilla transformer models required significant training data to outperform CNN models. Therefore, it might not be a good idea to train a transformer-based classifier on the *Moviescope* dataset from scratch. A possible workaround might be to use transfer learning to fine-tune a transformer model pre-trained on a task related to MTGC on the *Moviescope* dataset. However, using the proposed features with the pre-trained network in such a scenario will not be possible. Therefore, this work adopts another approach to perform the classification. The CNN models require comparatively lesser training data than transformer-based ones [54]. Thus, this work proposes to use the ACNN classifier for the MTGC task. The performance of the ACNN classifier in MTGC is discussed in the following subsection.

4.4 Performance of ACNN classifier

The performance of the proposed features with the ACNN classifier is presented in Table 4. It can be observed that the intermediate-fusion of GMF, SF, and SM-Conf features ($P1$ in Table 4) with the ACNN classifier performs better than both the baseline methods $B1$ and $B2$. The proposed system performs poorly for Comedy, Crime, Family, Fantasy, and Romance while improving for others. However, when combined with a *raw feature* like LMS, the system's performance improves significantly. In an intermediate fusion with the LMS feature ($P2$ in Table 4), the system provides lower performance only for the Crime, Fantasy, and Romance genres. However, the macro, micro, and sample average $AU(\overline{PRC})$ values improve significantly. In a late-fusion setting ($P3$ in Table 4), the average performances improve even further. The $P3$ system significantly improves the detection of Animation, Family, Horror, and Thriller genres. Note that GMMs trained on the *MUSAN* dataset are used in this work. The performance of the proposed features might have improved if the actual speech-music annotations for the movie trailer audio were available.

The effect of different segment durations on the classification performance of $P1$ is shown in Fig. 5. With the increase in segment size, there is a general trend of improved performance. There is a significant improvement when the segment size is increased to 30s. After that, a saturation of the performance is observed. Thus, 30s can be an optimal segment duration for training a classifier on the *Moviescope* dataset. Additionally, ablation results of the proposed features in a leave-one-out form are provided in Table 5. It can be observed that each of the proposed features brings some

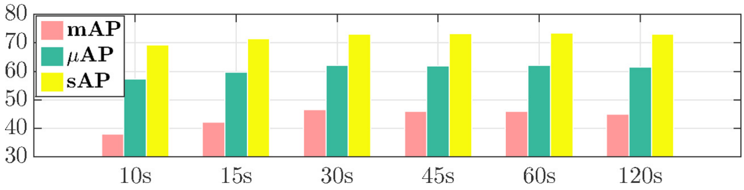


Fig. 5. Performance of $P1$ at different segment sizes.

Table 5. Ablation performances of the proposed features. Here, "Best" label in the table implies GPF+SF+SM-Conf (Intermediate-Fusion) with Time and Channel Attention. Results are reported in terms of genre-wise $AU(\overline{PRC})$. Also, macro average precision (**mAP**), micro average precision (**μAP**) and sample average precision (**sAP**) are reported in the last three columns.

Feature	**action**	**anim**	**bio**	**com**	**crime**	**drama**	**fam**	**fant**	**horr**	**myst**	**rom**	**scifi**	**thrlr**	**mAP**	**μAP**	**sAP**
Best	58.85	51.14	11.74	85.90	37.52	79.87	46.55	27.35	40.99	32.92	41.53	27.37	60.65	46.62	62.03	73.15
No GPF	54.83	40.2	10.05	83.42	34.26	73.93	37.26	21.86	32.66	23.52	40.88	25.42	53.52	41.26	58.06	71.03
No SF	60.91	41.18	10.49	85.64	39.04	77.87	40.76	21.56	41.57	31.69	43.49	25.32	57.04	44.65	59.09	70.61
No C-SM-Conf	60.45	44.73	10.43	86.18	35.79	78.96	40.8	22.79	38.64	30.08	41.07	28.46	58.91	44.74	61.55	72.93
No Time Attention	50.08	30.25	11.19	85.89	28.48	77.42	32.73	15.43	29.59	22.77	41.13	24.22	52.42	38.98	58.78	70.43
No Channel Attention	59.36	50.34	10.71	85.91	35.6	79.53	42.79	23.45	37.95	30.24	42.47	27.4	58.86	45.32	61.61	72.97
No Attention	60.27	48.48	10.95	85.28	34.17	79.95	41.75	25.91	41.59	28.96	41.85	25.73	58.51	45.24	61.61	72.84

additional information that improves the combined performance. Nonetheless, the GPF features and the time axis attention appear to be the essential components in the current proposal.

4.5 Ablation Study

Table 5 presents the results of the ablation study to investigate the importance of each of the proposed features and attention mechanisms. Each feature or attention operation is removed one at a time, and the subsequent performance of the system is noted. The particular feature or attention operation whose removal corresponds to the most significant drop in performance is considered as most important in the current task. In the present case, the removal of GPF features and the time attention correspond to the most significant drops in performance. Such results indicate that the learned GPF features are helpful in the current MTGC task. Using distribution modeling algorithms like GMMs to extract the GPF features might have made them more resilient to the complex sounds in movies. Moreover, the attention based temporal aggregation of the audio features appears to be aiding the MTGC task. Therefore, the results validate the hypothesis that learned features which tend to capture human-level information, like audio type, might be helpful in the MTGC task.

4.6 Generalization Performance

The generalization performance of the proposed MTGC models on a different dataset is also evaluated to establish the validity of the proposed method in MTGC. The *EmoGDB* [55] dataset consists of 100 Indian movie trailers with six non-overlapping genre labels for each movie. The *Moviescope* dataset predominantly consists of Hollywood movies. Hence, *EmoGDB* can be considered

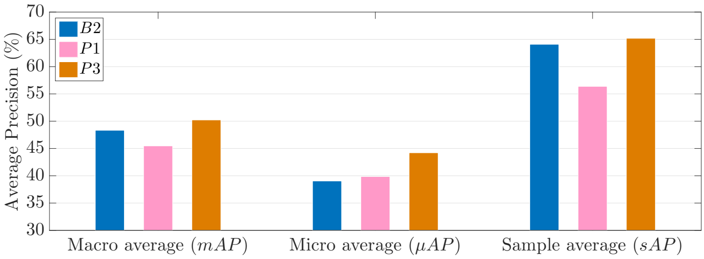


Fig. 6. Illustrating the generalization performance in the form of bar-charts. Results are reported in terms of macro average precision (**mAP**), micro average precision (**μAP**) and sample average precision (**sAP**).

an ideal choice for evaluating the generalization performance of models trained on the *Moviescope* dataset. For this experiment, the model predictions for only the six possible labels from the *EmoGDB* dataset are considered. The genre-wise predictions of both baseline and proposed methods for *EmoGDB* dataset are scaled to $[0, 1]$ over all samples before computing the performance metrics. This step accounts for the differences in train and test data. The cross-dataset results of the $B2$ baseline [26] and the proposed methods are illustrated in Fig. 6. It can be observed that the proposed features alone (see $P1$ in Fig. 6) do not perform better than the baseline in this case. However, the late-fusion combination of proposed features with LMS (see $P3$ in Fig. 6) provides significant improvement. Such results validate the efficacy of the proposed approach in MTGC.

5 CONCLUSION

The present work proposes the use of speech-music probability sequences for the task of movie trailer genre classification. The audio-type cues are encoded in a learned feature representation computed using spectral peak tracking of audio spectrograms. The learned feature, peak trace statistical measures, and sequence of speech-music confidence scores of trailer audio are proposed as features in this work. An Attention-based CNN classifier is used to perform the classification. Results obtained with the proposed approach justify the utilization of speech-music segmentation in movie genre classification. Moreover, the generalization performance of the proposed approach is also found to be satisfactory.

This work only considered the speech and music audio types for MTGC. Movies also contain other less frequent audio categories like environmental sounds and sound effects. Future work may be directed to extract genre-specific information from such audio types as well. Moreover, multi-modal approaches that include the present proposal as a component may also be explored for improved performance.

REFERENCES

[1] Sudhanshu Kumar, Kanjar De, and Partha Pratim Roy. Movie Recommendation System Using Sentiment Analysis From Microblogging Data. *IEEE Trans. on Comput. Social Syst.*, 7(4):915–923, 2020.

[2] Syjung Hwang and Eunil Park. Movie Recommendation Systems Using Actor-Based Matrix Computations in South Korea. *IEEE Trans. on Comput. Social Syst.*, 9(5):1387–1393, 2022.

[3] Ashima Yadav and Dinesh Kumar Vishwakarma. A unified framework of deep networks for genre classification using movie trailer. *Elsevier Appl. Soft Comput.*, 96:106624, 2020.

[4] Zeeshan Rasheed and Mubarak Shah. Movie genre classification by exploiting audio-visual features of previews. In *Proc. Object Recognit. Supported by User Interaction for Service Robots*,

volume 2, pages 1086–1089. IEEE, 2002.

[5] Zeeshan Rasheed, Yaser Sheikh, and Mubarak Shah. On the use of computable features for film classification. *IEEE Trans. on Circuits and Syst. for Video Technol.*, 15(1):52–64, 2005.

[6] Hee Lin Wang and Loong-Fah Cheong. Affective understanding in film. *IEEE Trans. on Circuits and Syst. for Video Technol.*, 16(6):689–704, 2006.

[7] Sanjay K Jain and RS Jadon. Movies genres classifier using neural network. In *Proc. 24th Int. Symp. on Computer and Inf. Sciences*, pages 575–580. IEEE, 2009.

[8] Aida Austin, Elliot Moore, Udit Gupta, and Parag Chordia. Characterization of movie genre based on music score. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, pages 421–424. IEEE, 2010.

[9] Theodoros Giannakopoulos, Alexandros Makris, Dimitrios Kosmopoulos, Stavros Perantonis, and Sergios Theodoridis. Audio-visual fusion for detecting violent scenes in videos. In *Proc. Hellenic Conf. on Artificial Intell.*, pages 91–100. Springer, 2010.

[10] Theodoros Giannakopoulos, Aggelos Pikrakis, and Sergios Theodoridis. A multi-class audio classification method with respect to violent content in movies using bayesian networks. In *Proc. IEEE 9th Workshop on Multimedia Signal Process.*, pages 90–93. IEEE, 2007.

[11] Go Irie, Takashi Satou, Akira Kojima, Toshihiko Yamasaki, and Kiyoharu Aizawa. Affective audio-visual words and latent topic driving model for realizing movie affective scene classification. *IEEE Trans. on Multimedia*, 12(6):523–535, 2010.

[12] Fillipe D. M. de Souza, Guillermo C. Chávez, Eduardo A. do Valle Jr., and Arnaldo de A. Araujo. Violence Detection in Video Using Spatio-Temporal Features. In *Proc. 23rd SIBGRAPI Conf. on Graphics, Patterns and Images*, pages 224–230, 2010.

[13] Howard Zhou, Tucker Hermans, Asmita V Karandikar, and James M Rehg. Movie genre classification via scene categorization. In *Proc. 18th ACM Int. Conf. on Multimedia*, pages 747–750, 2010.

[14] Liang-Hua Chen, Hsi-Wen Hsu, Li-Yun Wang, and Chih-Wen Su. Violence detection in movies. In *Proc. 8th Int. Conf. Computer Graphics, Imaging and Visualization*, pages 119–124. IEEE, 2011.

[15] Jianchao Wang, Bing Li, Weiming Hu, and Ou Wu. Horror video scene recognition via multiple-instance learning. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, pages 1325–1328. IEEE, 2011.

[16] Jianchao Wang, Bing Li, Weiming Hu, and Ou Wu. Horror movie scene recognition based on emotional perception. In *Proc. IEEE Int. Conf. on Image Process.*, pages 1489–1492. IEEE, 2010.

[17] Yin-Fu Huang and Shih-Hao Wang. Movie genre classification using SVM with audio and video features. In *Proc. Int. Conf. on Active Media Technol.*, pages 1–10. Springer, 2012.

[18] Esra Acar, Frank Hopfgartner, and Sahin Albayrak. Violence detection in hollywood movies by the fusion of visual and mid-level audio cues. In *Proc. 21st ACM Int. Conf. on Multimedia*, pages 717–720, 2013.

[19] Gabriel S Simões, Jônatas Wehrmann, Rodrigo C Barros, and Duncan D Ruiz. Movie genre classification with convolutional neural networks. In *Proc. Int. Joint Conf. on Neural Networks (IJCNN)*, pages 259–266. IEEE, 2016.

[20] Adarsh Tadimari, Naveen Kumar, Tanaya Guha, and Shrikanth S Narayanan. Opening big in box office? Trailer content can help. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, pages 2777–2781. IEEE, 2016.

[21] Jônatas Wehrmann, Rodrigo C Barros, Gabriel S Simões, Thomas S Paula, and Duncan D Ruiz. (Deep) learning from frames. In *Proc. 5th Brazilian Conf. on Intell. Syst. (BRACIS)*, pages 1–6. IEEE, 2016.

[22] Jônatas Wehrmann and Rodrigo C Barros. Movie genre classification: A multi-label approach based on convolutions through time. *Elsevier Appl. Soft Comput.*, 61:973–982, 2017.

[23] Jônatas Wehrmann and Rodrigo C Barros. Convolutions through time for multi-label movie genre classification. In *Proc. Symp. on Appl. Comput.*, pages 114–119, 2017.

[24] Jussi Tarvainen, Jorma Laaksonen, and Tapio Takala. Film mood and its quantitative determinants in different types of scenes. *IEEE Trans. on Affect. Comput.*, 11(2):313–326, 2018.

[25] Federico Álvarez, Faustino Sánchez, Gustavo Hernández-Peñaloza, David Jiménez, José Manuel Menéndez, and Guillermo Cisneros. On the influence of low-level visual features in film classification. *PLOS ONE*, 14(2):1–29, Feb 2019.

[26] Paola Cascante-Bonilla, Kalpathy Sitaraman, Mengjia Luo, and Vicente Ordonez. Moviescope: Large-scale analysis of movies using multiple modalities. *arXiv preprint*, arXiv:1908.03180, 2019.

[27] Jeong A Wi, Soojin Jang, and Youngbin Kim. Poster-Based Multiple Movie Genre Classification Using Inter-Channel Features. *IEEE Access*, 8:66615–66624, 2020.

[28] Wei-Ta Chu and Hung-Jui Guo. Movie genre classification based on poster images with deep neural networks. In *Proc. Workshop on Multimodal Understanding of Social, Affect. and Subjective Attributes (MUSA2'17)*, pages 39–45, 2017.

[29] Prashant Giridhar Shambharkar, Pratyush Thakur, Shaikh Imadoddin, Shantanu Chauhan, and MN Doja. Genre Classification of Movie Trailers using 3D Convolutional Neural Networks. In *Proc. 4th Int. Conf. on Intell. Comput. and Control Syst. (ICICCS)*, pages 850–858. IEEE, 2020.

[30] Rafael B Mangolin, Rodolfo M Pereira, Alceu S Britto, Carlos N Silla, Valéria D Feltrim, Diego Bertolini, and Yandre MG Costa. A multimodal approach for multi-label movie genre classification. *Springer Multimedia Tools and Appl.*, pages 1–26, 2020.

[31] Edward Fish, Jon Weinbren, and Andrew Gilbert. Rethinking movie genre classification with fine-grained semantic clustering. *arXiv preprint*, arXiv:2012.02639, 2020.

[32] Edward Fish, Jon Weinbren, and Andrew Gilbert. Rethinking Genre Classification With Fine Grained Semantic Clustering. In *Proc. IEEE Int. Conf. on Image Process. (ICIP)*, pages 1274–1278, 2021.

[33] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint*, arXiv:1609.08675, 2016.

[34] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. *IEEE Trans. on Pattern Analysis & Machine Intell.*, 40 (06):1437–1451, Jun 2018. ISSN 1939-3539.

[35] Aditya Sharma, Mayank Jindal, Ayush Mittal, and Dinesh Kumar Vishwakarma. A Unified Audio Analysis Framework For Movie Genre Classification Using Movie Trailers. In *Proc. Int. Conf. on Emerging Smart Comput. and Informatics (ESCI)*, pages 510–515. IEEE, 2021.

[36] Dinesh Kumar Vishwakarma, Mayank Jindal, Ayush Mittal, and Aditya Sharma. Multilevel profiling of situation and dialogue-based deep networks for movie genre classification using movie trailers. *arXiv preprint*, arXiv:2109.06488, 2021.

[37] Shangfei Wang and Qiang Ji. Video Affective Content Analysis: A Survey of State-of-the-Art Methods. *IEEE Trans. on Affect. Comput.*, 6(4):410–430, 2015.

[38] Mihai Gabriel Constantin, Liviu Daniel Stefan, Bogdan Ionescu, Claire-Helene Demarty, Mats Sjoberg, Markus Schedl, and Guillaume Gravier. Affect in Multimedia: Benchmarking Violent Scenes Detection. *IEEE Trans. on Affect. Comput.*, pages 1–1, 2020.

[39] David Bordwell and Kristin Thompson. *Film Art: An Introduction.* McGraw Hill, 8, revised edition, 2008. ISBN 0073310271, 9780073310275.

[40] Mrinmoy Bhattacharjee, S. R. Mahadeva Prasanna, and Prithwijit Guha. Speech/Music Classification Using Features From Spectral Peaks. *IEEE/ACM Trans. on Audio, Speech, and Language Process.*, 28:1549–1559, 2020.

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Inf. Processing Systems*, 30, Dec 2017.

[42] Mohaddeseh Mirbeygi, Aminollah Mahabadi, and Akbar Ranjbar. Speech and music separation approaches-a survey. *Multimedia Tools and Appl.*, pages 1–43, 2022.

[43] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur. X-vectors: Robust dnn embeddings for speaker recognition. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, pages 5329–5333. IEEE, 2018.

[44] Xulong Zhang, Jianzong Wang, Ning Cheng, and Jing Xiao. Singer identification for metaverse with timbral and middle-level perceptual features. *arXiv preprint*, arXiv:2205.11817, 2022.

[45] Laureano Moro-Velazquez, Jesus Villalba, and Najim Dehak. Using x-vectors to automatically detect parkinson's disease from speech. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Process. (ICASSP)*, pages 1155–1159. IEEE, 2020.

[46] David Snyder, Daniel Garcia-Romero, Alan McCree, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur. Spoken Language Recognition using X-vectors. In *Odyssey*, pages 105–111, 2018.

[47] Hossein Zeinali, Lukas Burget, and Jan Cernocky. Convolutional neural networks and x-vector embedding for DCASE2018 acoustic scene classification challenge. *arXiv preprint*, arXiv:1810.04273, 2018.

[48] Mirco Ravanelli, Titouan Parcollet, Peter Plantinga, Aku Rouhe, Samuele Cornell, Loren Lugosch, Cem Subakan, Nauman Dawalatabad, Abdelwahab Heba, Jianyuan Zhong, Ju-Chieh Chou, Sung-Lin Yeh, Szu-Wei Fu, Chien-Feng Liao, Elena Rastorgueva, François Grondin, William Aris, Hwidong Na, Yan Gao, Renato De Mori, and Yoshua Bengio. SpeechBrain: A General-Purpose Speech Toolkit, 2021. arXiv:2106.04624.

[49] David Snyder, Guoguo Chen, and Daniel Povey. Musan: A music, speech, and noise corpus. *arXiv preprint*, arXiv:1510.08484, 2015.

[50] Yitong Yu, Ziyu Lu, Yang Li, and Delong Liu. ASTS: attention based spatio-temporal sequential framework for movie trailer genre classification. *Multimedia Tools and Appl.*, 80(7):9749–9764, 2021.

[51] Jing Li, Kan Jin, Dalin Zhou, Naoyuki Kubota, and Zhaojie Ju. Attention mechanism-based CNN for facial expression recognition. *Neurocomputing*, 411:340–350, 2020. ISSN 0925-2312.

[52] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint*, arXiv:1412.6980, 2014.

[53] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proc. Conf. of the North Amer. chapter of the Assoc. for Comput. Linguistics: Human Lang. Technol.*, pages 1480–1489, 2016.

[54] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *Proc. Int. Conf. on Learn. Representations (ICLR)*, Vienna, Austria, May 2021.

[55] Ashima Yadav and Dinesh Kumar Vishwakarma. A unified framework of deep networks for genre classification using movie trailer. *Elsevier Appl. Soft Comput.*, 96:106624, 2020.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009