INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Multimodal Approach to Recommend Movie Genres Based on Multi Datasets

Preksha Khant[1]*, **Bharat Tidke**[2]

1 M. Tech Student, Department of Computer Engineering and Technology, MIT – WPU, Pune, India
2 Professor, Department of Computer Engineering and Technology, MIT – WPU, Pune, India

OPEN ACCESS

Received: 22-05-2023
Accepted: 06-07-2023
Published: 08-08-2023

Citation: Khant P, Tidke B (2023) Multimodal Approach to Recommend Movie Genres Based on Multi Datasets. Indian Journal of Science and Technology 16(30): 2304-2310. https://doi.org/10.17485/IJST/v16i30.1238

* **Corresponding author**.

prekshakhant211@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: The main purpose of this study is to implement the multimodal as well as the multilabel approach of accurate movie genre classification using the Actor, Director, Writer, Poster Images, and Synopsis data. The various kind of data collected from IMDB as well as existing datasets are used to train the models. **Methods:** The 5 deep learning models are created on Poster Images, Text Data, Actors Data, Directors Data, and Writers Data. These models are then combined using the weighted average ensemble model of deep learning. The final output gives the prediction scores of the top 3 genres for a given movie. The main deep learning model used for poster image model training is CNN. The LSTM model is used for text/synopsis model training. The multilabel along with the multimodal approach yielded good results in terms of accurate predictions than the existing models. **Findings:** The total F1 Score is 0.65. Along with good results, it has some limitations which are further discussed in the paper. **Novelty:** The novelty of the work lies in 3 major aspects, firstly the consideration of the Actor, Director, and Writer data for the genre classification, as usually, an actor does a particular kind of movie, a writer writes a particular type of movie and director directs a particular type or genre movie. Although there are many exceptions but usually this is the case. The second novelty lies in combining the models using the weighted average method wherein every model is assigned a weight for how much it will contribute to the final genre classification. And lastly, the novelty lies in how the weights are defined i.e., we used the correlation method to determine the weights of every individual model for genre classification.

Keywords: Movie Genre Classification; CNN; Multimodal Movie Genre; Multilabel Movie Genre; LSTM; Deep Learning

1 Introduction

The fast expansion of online streaming platforms and their numerous uses, including digital storytelling, movie recommendations, and video retrievals, has significantly advanced the subject of classifying movie genres. Movie Genre Classification has always been the area of the study for past decades yet the results are not so convincing. Genre Classification is an important task as it is used in variety of domains like

recommendation. Movies usually have more than one genre as they depict different characteristics. The obvious distinction of genres is not possible as these genres overlap each other or they might have some characteristics of other genres. Examples of similarities between two or more genes are thrillers and dramas, which both feature lengthy conversations shot in slow motion. Various kinds of data or modalities are used for genre classification such as text, poster, video, and audio. Previous researchers have used ML as well as DL methods to classify the genres using unimodal individually or as the combinations of modalities making it a multimodal approach. In this study multimodal as well as multilabel approach is used.

The paper is organized into 5 sections. Section 1 presents an introduction to some of the previous works done in this particular field. Section 2, gives the overview of the dataset, and the detailed Proposed methodology is explained with the steps taken, Section 3, states and compares the results found with the existing works, and Section 4, Conclusion and Future works are directed.

Earlier research has looked into a range of techniques for classifying films according to their genre, the first of which is using the text/synopsis. Text features are used to train the machine learning models which automates the process of movie genre classification [(1)]. The text/synopsis of the movie is usually a short sentence that might not include the complete information of the movie which is why the model could not perform well in some genre cases. The other type of technique used is image or poster-based classification wherein the image features are passed to the deep learning models such as CNNs [(2)]. The main drawback of using posters for the classification of genres is they might not be the correct representative of the actual genres of the movie. For example, the movie poster might be the representation of the family and drama genre whereas the actual movie might be of crime and thriller.

The high-level features included object recognition, whereas low-level features include colors and edges. For the purpose of feature extraction as well as classification, the CNN model was used. The combination of text and images also is done to enhance the model performance in movie genre classification as in [(3)]. The video + audio features are also used for classification wherein the deep learning models such as in [(4)] where a pre-trained model named 13D is used for the generation of the video representations and the LSTM model to generate the fusion of features. The major disadvantage of using video + audio data features is it requires high processing resources as well as high processing time. The poster images, Synopsis, Subtitle, and Trailer were used as media for multimodal movie genre classification [(5)]. The deep Neural networks were used to individually create models for each media. The IMF (Intermediate Multimodal Fusion) is used to combine all the media and DMF (Dense Multimodal Fusion) is used to combine the unimodal. The late fusion strategies were also used to merge the models and finally to compare with the IMF and DMF. Human intuition was the purpose of movie genre classification from the poster images in [(6)]. The four main categories of features considered are object embedding, Actor recognition, Age and Gender estimation, and emotions and race estimation is used. For feature extraction, the YOLO-V5 model is used. The Glove word embeddings model is used for generating word vectors.

The major gaps of the previous works are in terms of the accuracy in the classification as well as not considering the actor, director, and writer data for the purpose of classification as usually, the actor does a particular kind of movie, similarly director directs a particular kind of movie and same with writers. Although there are some exceptions, usually this is the case and which is why they could be the major features in the classification purposes. Other gaps that were found are in terms of the fusion techniques used. The major previous works have used late fusion strategies for combining the unimodal for the purpose of classification [(5)].

Considering the major gaps, in order to improve the accuracy of the automated movie genre classification, we proposed a Multimodal and Multilabel ensemble classification method. The novelty of the approach lies in using Actor, Director, and Writer data usage along with poster and text data. The fusion strategy used to combine the models is the weighted-average method, where the weights are calculated for every individual model based on the correlation between actual and predicted values, found for that particular model. The primary goal of this research is listed below:

a. A new database is created which is collected from IMDB.

b. A Multimodal as well as Multilabel movie genre classification.

c. Using Actors, Directors, and Writers' data for movie genre classification.

d. Usage of the Weighted average ensemble method to combine the 5 deep learning models i.e., Text/Synopsis, Poster image, Actor, director, and Writer.

2 Proposed Methodology

For multimodal implementation, various available datasets were used in order to train the data. The first dataset used for training images and text was collected by us using the IMDB scraper tool which included all the famous past movies from all over the world from 1950 to 2020 having 8000 entries. The freely available dataset on Kaggle named movies.csv consists of 6820 movies from 1986 to 2016 for building actor, director, and writer models.

As mentioned above, our main aim was to implement the multimodal as well as the multilabel approach to correctly classify movie genres. In order to achieve our aims, specific methods were used and selected to give the best performances for the given movies.

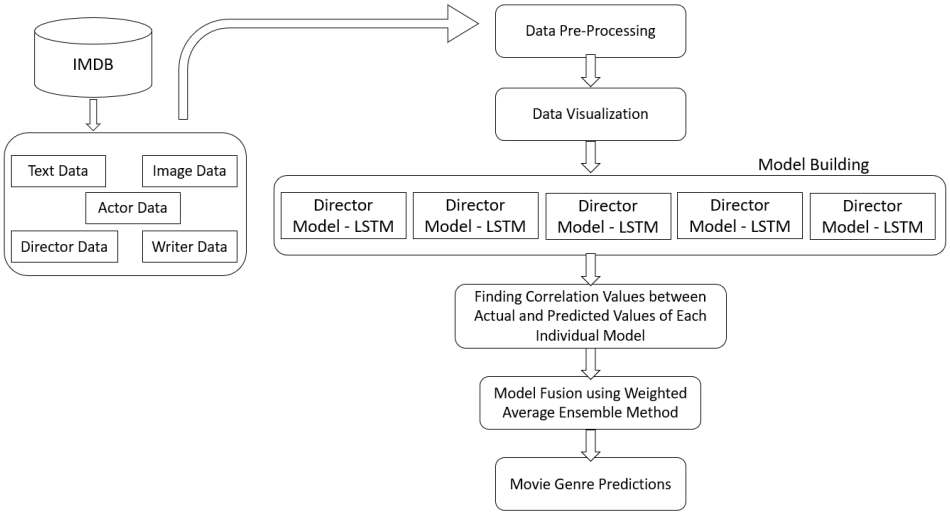

Fig 1. Proposed Algorithm for Genre Classification

Figure 1 is the basic idea behind this proposal which includes data collection in Step 1, next followed by the Step 2 having the data cleaning and preprocessing, next Step 3 is building the individual models, next the Classification is Step 4, next Step 5 is finding the correlation of the actual and predicted outputs, and Step 6 is the fusion of the individual models, lastly, Step 7 is concluded with the predictions of the individual movies.

Step 1, In this Step the Data collection is done from various sources for 5 modalities. The main source of movie data collection is IMDB.

Step 2 consisted of data cleaning and preprocessing, where all the null values from the data were removed initially, and then followed by this, the data preprocessing was done.

2.1 Data Preprocessing for Text data

To create the models from the textual data, firstly the data preprocessing was done. Firstly, the fields having null values are removed. Next, the data sentences were lowercase. However, the sentences included punctuation which was removed at this stage. Next, the stop words are removed and the stop word listing was done manually. Stemming was performed on the sentences in order to get the most basic forms of the words which were free from any prefixes or suffixes. The sentences were then treated with the removal of emojis or emoticons if any. The URLs and the HTML tags, if any, were being removed. Lastly, this preprocessed data is saved for further usage.

In the case of the Actor's data, the preprocessing included the conversion to lower cases and the removal of fields having null values. Similar to this, the director and writer's data were lowercase, and null fields were removed., the director and writer's data were being lowercased and null fields were removed.

2.2 Data Preprocessing for Image Data

Firstly, the images are brought the same size which is 400*400*3. The next part of the research included the pre-processing of images. Here, all the images were converted to RGB.

An RGB-colored image has three channels Red, Green, and Blue. The format in which the images are stored is [H, W, C] format where the image height is H, the image width is W, and the number of channels is C. All three channels range between 0 to 255.

After the conversion of images to RGB, they are converted to an array value and these are appended in the array storing all the image arrays.

2.3 Data Visualization

Once the pre-preprocessing is completed, the next part comes from visualizing the given data. The main aim of this step is to come up with a better idea about the data. For textual data, the genre word clouds were formed to understand the most frequent words in sentences.

It was noted that the most common words in the Sports genre were Cricket, play, team, win, player, world, game, victory, and journey. For the Thriller genre, frequent words include life, crime, murder, genre, and kill. Similarly, the most frequent words for all the genres were found and studied. Next, to check the distribution of different genres in the dataset, the plot is built where the Drama had the maximum number of data followed by comedy and thriller.

The sequence lengths of the text sentences were plotted. It was found that most of the sentence's length lies between 170-200 words.

2.4 Building Individual Models

In this Step, the individual models are built for all 5 modalities i.e., Text/Synopsis/Plot, Poster image, Actor, Director, and Writer.

2.4.1 Image model

The data was divided 70:20:10 between training, validation and testing respectively in order to develop the picture model. The Convolutional Neural Network (CNN) is the model that is utilized for picture training. VGG-16 which has 16 layers was used in the research. Conv2D, the convolution layer that helps create the convolution kernel and, when paired with the input layers of the Keras model, provides an output tensor, is the initial layer of the model. The Model comprised of 13 convolutional layers, 2 max-pooling layers and 3 Dense Layers. By conducting the convolution of the pictures and the kernel, the kernel is created as a filter or convolutional matrix that is then used for edge recognition, sharpening, blurring, and embossing. In between, the max pooling and dropout layers are introduced to forward the most useful neurons to the next layers. The model then introduces thick layers that are intricately coupled to the layers that came before them, meaning that every neuron in the current layer is connected to every neuron in the previous layer.

The model's optimizer is called "Adam." The loss function is Binary Cross Entropy because usually in multi-label classification problems this is used. Given that we are dealing with multilabel classification, the activation function employed in the last layer is the sigmoid activation function. The model is also trained using the provided training data and verified using the provided test data.

2.4.2 Text Model

For text model building firstly the tokenizer is used to transform the sentences into tokens. The text_to_sequences function is then invoked, which converts each text in the text into an integer sequence. Only words recognized by the tokenizer and the top num_words-1 common words are taken into consideration. The pad_sequences function is used in order to ensure the same length of all the sequences and it is set to maximum i.e., the max length of a sequence from the dataset will be chosen and all the sequences will be padded to the same length.

These padded sequences are then passed to the LSTM model. The LSTM model is built with the first layer being the embedding layer. Here all the sequences are embedded in the vectors. Since this technique takes less space for the representation of words i.e., dimension reduction and every word having a real-valued vector of fixed length are the main reasons for choosing this technique as compared to one-hot encoding. The next layer is the LSTM layer which gets input from the embedding layer which learns and remembers the long-term dependencies between time steps in time series and sequential data.

Lastly, the dense layers are introduced to connect each neuron of the current layer to every neuron of the preceding layer. The activation function used in the last layer is the sigmoid activation function as it helps in dealing with multilabel classification.

Similar to image models, the Adam and binary cross entropy are used as optimization and loss functions in the text model.

2.4.3 Actor, Director and Writer Models

In the actor's model, similar to the text model, initially the sentences are tokenized. Here the sentences are nothing but the full names of the Actor, Directors and the Writers. The length of these sentences is mostly 3 words. Then the text sentences are converted to sequences of max_length. Further, the padded sequences generated using the max_length are passed to LSTM model.

The embedding layer is the top layer of the LSTM model, 2^{nd} is the LSTM layer that learns and remembers the long-term dependencies. Followed by the dense and dropout layers are present that connects the neurons and drops out the less weighted neurons. The sigmoid activation function is and the optimizer used is Adam. The model is then trained and validated on the

given data.

2.5 Finding Correlation between Actual and Predicted Values

Since the correlation factor gives the correlation between the two given features, it is used at this stage to find the correlation between the actual and the predicted genre values. This will give us how well the predictions are done by each individual model and this will be further used to find the weight of that model. For each individual, the correlation values are found using the NumPy library function known as corrcoef. This function gives the correlation coefficient between the actual and predicted genres.

2.6 Fusion of Models

In this Step, the fusion of the models is done. The weighted average ensemble method is used to combine the results from all 5 models. In the weighted average ensemble method, each data point of a given model is multiplied by the given weight, which is then summed up and finally divided by the number of models. In contrast to a simple average, a weighted average takes into account the relative importance or contribution of the elements being averaged. As a consequence, it gives the average elements that occur relatively more frequently greater weight. This justifies the method's selection for fusion.

2.6.1 Building weighted Average Ensemble model

At this stage, firstly, all the models are loaded and assigned the names. Next, the output layers are created with the fixed lengths of genres i.e., 25. Following this, the new model is created with the output layer having 25 labels.

Functions are defined to preprocess the upcoming text, image, actor, director, and writer data.

Lastly, the function is defined wherein the predictions are made by each of the individual models and the results are merged using the weighted average ensemble method. Further, the top 3 genres are displayed to the user from the given 25 genres along with their probabilities.

2.7 Prediction of Movie Genres

The last step is to predict the movie genres based on the given inputs of the movie. The user gives the input as the actor's name, director's name, writer's name, the poster of the movie, and the synopsis of the movie as the input and the model gives the output as the top genres for that particular movie as shown in Figure 2.


```
1/1 [==============================] - 0s 233ms/step
1/1 [==============================] - 0s 485ms/step
1/1 [==============================] - 0s 458ms/step
1/1 [==============================] - 0s 488ms/step
1/1 [==============================] - 1s 1s/step
Top 3 predicted genres:
History (0.19)
Drama (0.13)
Fantasy (0.13)
```

Fig 2. Movie Genre Prediction

3 Results and Discussion

Machine learning[2,7] as well as deep learning models[8] were used to compare the existing model's performance with the proposed model. The proposed method is compared with the existing works having unimodal approaches as well as multimodal approaches below.

In cases where the datasets are severely skewed, a model that performs badly and only predicts the majority class will appear correct based on other measures, such as accuracy. The model won't have excellent accuracy or recall on the positive class; hence the full level of underperformance will be shown in the F1 score. Thus, the evaluation metric chosen for evaluating the model performance is F1-Score. F1 Score states the harmonic mean of the recall and precision. The F1 score states how well précised as well as how robust it is for the given values. A higher F1 score states that the model has high precision as well as high recall i.e., the model classifies each input correctly whereas low F1 scores state the imbalance between precision and recall and the model doesn't perform well for the given input test data. Results are shown in Table 1. The overall result analysis shows that the proposed model performed very well on the given input data with a total F1 score of 0.65. The unimodal text or synopsis data given as input for genre classification gave the F1 score of 0.58 using the machine learning algorithm – SVM[8]. The poster model performed very poorly in this case, due to some reasons such as the poster not being clearly able to represent the movie genres, or the poster not clearly containing many combinations of genres[2] with the F1 score of 0.22. The multimodal approach performed well considering text, subtitle, poster, and video data[7], with an F1 score of 0.64. The proposed method gave an overall F1 score of 0.65 which includes text, poster, actors, directors, and writers' data as input.

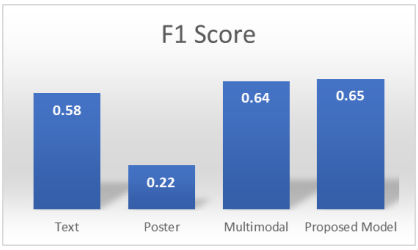

Fig 3. The comparison between the existing works with the proposed model

Table 1. Results of the Proposed model

Model Name	F1 Score
Text Model (Synopsis)	0.81
Actor Model	0.35
Director Model	0.36
Writer Model	0.33
Poster Model	0.89
Total	0.65

The text model gave the F1 score of 0.81. It was found that some of the genres such as Family, Biography, Fantasy, and Mystery were accurately classified using text as they consisted of some specific words belonging to particular categories. For example, the Family genre has words in it such as Mother, Father, Son, etc. Animation, Crime, War, were best recognized using the Poster model whereas it was difficult to differentiate between some genres that have similar characteristics such as Action and Thriller.

The Actor, Director, and Writer Models performed very well on the trained data. It is analyzed that the low values of the F1 Score in the proposed model were due to the less training of the models of Actor, Director, and Writer models. The model training was done on the small dataset and also it was trained on fewer features. The model accurately classified the genres for the trained actor, director, and writer models which means if trained on a large dataset with rigorous training it would yield more accurate and good results. Overall, it definitely improved the prediction of genres. Thus, more proper training is required on these models.

4 Conclusion

In this research, Genre Classification was the main objective as it is vital to get the overall idea about the movie and its characters. The main objectives of this study were to accurately classify a movie's genre based on text, poster, actor, director, and writer's data. The novelty of the work lies in using actor, director, and writer's data for genre prediction. Also, the ensemble of the models is done using the weighted average method where the weights are defined using the correlation method. Using this novel approach, the results yielded are better in terms of Accuracy and F1-Score. The overall F1-Score for the proposed model achieved is 0.65 whereas in previous works the maximum F1-Score achieved was of 0.64. Proposed model has capabilities globally as it uses Actor's, Director's and Writer's data as usually Actor's, Director's and Writer's do the movies in some specific genres they are good at.

Although the study has been more effective and efficient than the existing methods, there are some limitations to it. The data used for the study is unbalanced, which is why the training for some of the genres is very less. Deep learning requires huge computational resources as well as large datasets to give the best results, here we used a small dataset and instead, the large dataset could be used for training deep learning models. The data used for training the model was historical data. For future studies, high-level features such as items, scenes, and behaviors could be studied to get more visual results from the image. Also, the video and audio modalities could be incorporated into the proposed algorithm when having relevant resources. Also, the real-time data could be used in order to get the best of results.

References

1) Agarwal A, Das RR, Das AR. Machine Learning Techniques for Automated Movie Genre Classification Tool. *2021 4th International Conference on Recent Developments in Control, Automation & Power Engineering (RDCAPE)*. 2021. Available from: https://doi.org/10.1109/RDCAPE52977.2021.9633422.
2) Hossain N, Ahamad MM, Aktar S, Moni MA. Movie Genre Classification with Deep Neural Network using Poster Images. *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*. 2021. Available from: https://doi.org/10.1109/icict4sd50815.2021.9396778.
3) Braz L, Teixeira V, Pedrini H, Dias Z. Image-Text Integration Using a Multimodal Fusion Network Module for Movie Genre Classification. *11th International Conference of Pattern Recognition Systems (ICPRS 2021)*. 2021. Available from: https://doi.org/10.1049/icp.2021.1456.
4) Bi T, Jarnikov D, Lukkien J. Video Representation Fusion Network For Multi-Label Movie Genre Classification. *2020 25th International Conference on Pattern Recognition (ICPR)*. 2021. Available from: https://doi.org/10.1109/icpr48806.2021.9412480.
5) Mangolin RB, Pereira RM, Britto AS, Silla CN, Feltrim VD, Bertolini D. A multimodal approach for multi-label movie genre classification. 2020. Available from: https://doi.org/10.1007/s11042-020-10086-2.
6) Nadem F, Mahdian R, Zareian H. Genre Classification of Movies from a Single Poster Image Using Feature Fusion. *2021 7th International Conference on Signal Processing and Intelligent Systems (ICSPIS)*. 2021. Available from: https://doi.org/10.1109/icspis54653.2021.9729380.
7) Paulino MAD, Costa YMG, Feltrim VD. Evaluating multimodal strategies for multi-label movie genre classification. *2022 29th International Conference on Systems, Signals and Image Processing (IWSSIP)*. 2022. Available from: https://doi.org/10.1109/iwssip55020.2022.9854451.
8) Akbar J, Utami E, Yaqin A. Multi-Label Classification of Film Genres Based on Synopsis Using Support Vector Machine, Logistic Regression and Naïve Bayes Algorithms. *2022 6th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*. 2022. Available from: https://doi.org/10.1109/icitisee57756.2022.10057828.