Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 222 (2023) 591–600

Procedia Computer Science

www.elsevier.com/locate/procedia

International Neural Network Society Workshop on Deep Learning Innovations and Applications (INNS DLIA 2023)

# Task-Specific and Graph Convolutional Network based Multi-modal Movie Recommendation System in Indian Setting

Prabir Mondal[a,*], Pulkit Kapoor[b], Siddharth Singh[c], Sriparna Saha[b], Jyoti Prakash Singh[a], Naoyuki Onoe[d]

[a]*National Institute of Technology, Patna-800005, India*
[b]*Indian Institute of Technology, Patna-801103, India*
[c]*Indian Institute of Technology, Jodhpur-342037, India*
[d]*Sony Research India Pvt. Ltd., Bengaluru, Karnataka – 560103, India*

## Abstract

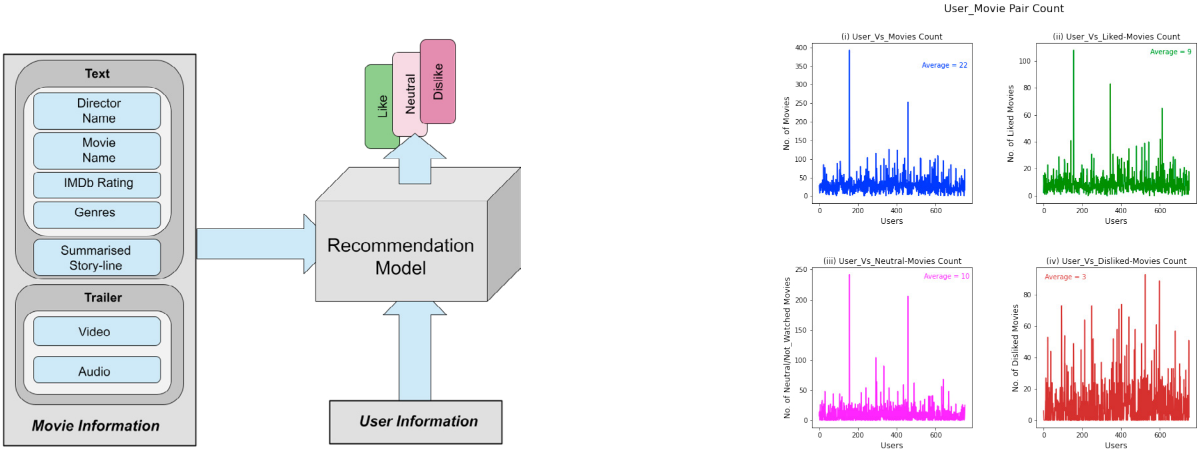
Nowadays the Recommendation System, a subclass of information filtering system does not require any introduction, and the movie recommendation system plays a vital role in the streaming platform where many movies are needed to analyze before showcasing a perfectly matched subset of them to its users. Most of the available datasets contain the rating information of user-movie pairs and this is the reason of regression-based works that predict the rating value for a user-movie pair. We have also found that there is no work on the Indian regional language-based dataset containing no users' feedback in the rating scale.

In this paper, we have introduced a recommendation system for the Indian language-based multi-modal Hindi movies' dataset where users' feedback is from the three different classes, *i) Dislike, ii) Like, and iii) Neutral*. Here, we have used the Flickscore dataset and added the audio-video information of the trailers of its movies for making it multi-modal. Besides that, we have investigated the performance of a classification-based model having two modules, (i) Task-Specific (TS) and (ii) Graph Convolutional Network (GCN). The performance of different combinations of these modules is tested on different modalities of the dataset. We have tested its performance in cold-start scenarios also. Modality wise different embedding processes have been introduced here and the experimental results tried to conclude how the model works in uni-modal, bi-modal, and all-modal information of movies in an information system where no rating information is present.

*Keywords:* Recommendation system, Multi-modal; Sentence-BERT; wav2vec; TimeSformer; Graph Convolutional Network; Task-Specific; Flickscore; Indian Regional Language Movies

[*] Corresponding author. Tel.: +91-8100438082
*E-mail address:* prabirm.phd22.cs@nitp.ac.in

10.1016/j.procs.2023.08.197



(a) Problem Statement.

(b) Feedback-wise User-Movie count.

Fig. 1: **(a)Problem Statement:** Graphical Representation. **(b)Feedback-wise User-Movie count:** (i):*User-Vs-Movie Count for all feedback*. (ii):*User-Vs-Movie Count for the Liked Movies*, (iii):*User-Vs-Movie Count for the Neutral/Not-Watched Movies*, (iv): *User-Vs-Movie Count for the Dislike Movies*

## 1. Introduction

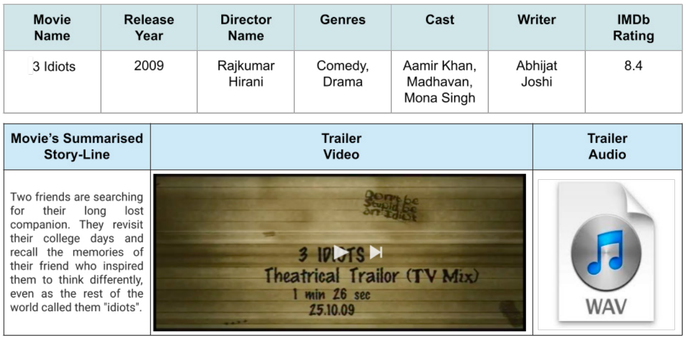
Increasing internet data volume in day-to-day demands has become quite challenging to the platform where users have zero decision time but accurate data finding desire. Understanding the users' preferences is highly demanding and the popular research topic in AI/ML field for now. The recommendation system is one of the best suggestions for this real-time problem of the digital world and the movie recommendation system is also in the front row. Most of the regression-based approaches are being done to predict the rating value as the preference score of a user-movie pair. A movie has different modalities in different forms like *Text, Video, and Audio*. Models' performances on different combinations of modalities are used to measure the effectiveness of multi-modal models and thus a lot of investigations in research are required to develop a real-time system for the purpose.

Here, we have investigated the performance of a Task-Specific (TS) and Graph Convolutional Network (GCN)[12] based model over the Indian regional Hindi language-based movie dataset Flickscore [1]. This dataset is made multi-modal by adding the audio-video information of movies. A thorough investigation has been done to check how the model works in different combinations of modalities and also to find how effective the TS-GCN combination is in the model. The model's performance is also measured in the cold start movie dataset.

Our thorough literature survey finds that as per the availability of the datasets, recent studies on Movie/Video recommendation system is mostly related to English language-based Hollywood movies. The accessible datasets are uni-modal and basically textual content based only. So, our first motivation is building a recommendation system where Indian regional language-based content is used to train the model and use the multi-modal information of movies rather than uni-modal to make the model more robust.

Secondly, most of the research is done in such a setup where movies have some rating information given by their users in a specified range, say 1 to 5. 1 means **do-not-like** and 5 means **Outstandingly-good**. In a practical scenario, busy users like to give their feedback in as simple as possible way but expect that the recommendation will be very preference specific, and accurate. Giving feedback in **Dislike/Like/Neutral** manner is simpler than giving it in a numerical form. But it is quite difficult to understand the users' preferences precisely with this simple form of feedback and this also motivates our research where the classification-based approach is learned to the model instead of regression.

Above all, a thorough investigation has been conducted to check how the task-specific and GCN work in the said challenge.



| Movie Name | Release Year | Director Name | Genres | Cast | Writer | IMDb Rating |
|---|---|---|---|---|---|---|
| 3 Idiots | 2009 | Rajkumar Hirani | Comedy, Drama | Aamir Khan, Madhavan, Mona Singh | Abhijat Joshi | 8.4 |

| Movie's Summarised Story-Line | Trailer Video | Trailer Audio |
|---|---|---|
| Two friends are searching for their long lost companion. They revisit their college days and recall the memories of their friend who inspired them to think differently, even as the rest of the world called them "idiots". | | WAV |

Fig. 2: **A Movie Data sample** of Multi-modal Hindi Movie Flickscore Dataset

## 2. Related Works

Analyzing the users' feedback and finding out the correlation of it with the items is one of the ways to Recommendation system building. Authors in [20] proposed a zero user profile-based RS where multi-modal information is fused and attention technique is applied to the modality information by considering the user feedback. The power of deep learning has boosted the analyzing scope of multi-modal data rigorously and authors in [5] introduced Deep Neural Network Based trusted filter (DNN-filter) in predicting the rating values for cold-start users. Predicting the rating value is the major target in recent regression-based work. A method in [15] also tries to predict the ratings of extended multi-modal MovieLens[1] dataset users. Work proposed by [11] finds the preference of users by tracking their affect or emotion by its ARTIST model. It uses different sensors for capturing human reaction signals and tries to predict the preferences towards the items. Movie Recommendation system by content-based approach[14] has been developed in the work of [16] and it uses the genre content-wise correlations among the MovieLens movies for its model.

There are also extensive works being done in the RS area with the incorporation of Graph Neural Network[18] based approach. By introducing the Graph Attention Network[19] in the proposed model, authors in [4] tried to handle the cold-start problem[13] over the multi-modal MovieLens dataset. Similarly in [8], the graph attention is used in uniforming the user's as well as the item's auxiliary information for its RS. The two layers of the graph network help the author[10] in analyzing the correlations among high-degree and low-degree associations of the book with its user in book RS.

It is quite clear that most of the works are regression-based and this motivates us to investigate how RS performs in classification.

## 3. Dataset

Most of the publicly available movie-information-containing datasets are basically of Hollywood movies and the users' preferences are from rating value space of specified range. But in this paper, by following our motivation, we have used the Flickscore[1] dataset. This dataset contains the meta information on 2,851 movies of 14 different genres (Biography, Musical, Mystery, Adventure, Action, War, Music, Fantasy, Romance, Thriller, Family, History, Crime, and Drama), 18 different Indian regional languages (Hindi, Bengali, Assamese, Tamil, Nepali, Punjabi, Rajasthani, Malayalam, Bhojpuri, Kannada, Haryanvi, Manipuri, Urdu, Marathi, Telugu, Oriya, Gujarati, Konkani) and their 919 number of viewers' meta information. In this dataset, the Hindi language movie has more samples, more users, and more availability of movie trailers online. For this reason, among all movies, we have considered only those movies belonging to the Hindi language for our proposed experiment and made this text-based dataset multi-modal by adding their video-audio information. Here, the users have given their feedback to the movies in three different ways, ***(-1)***

[1] https://movielens.org/

***for Dislike, (1) for Like, and (0) for Neutral or Not Watched***. No conventional rating values in a specified range are present in this dataset.

***(a)Movies' textual information:*** Out of 2,851 Indian-regional-language movies this dataset has 615 Hindi movies with 902 users. The movie's textual information contains the movie's summarised storyline written in pure English sentences and meta information like release year, IMDb rating, writer, director, cast, genre, and movie name. It also contains users' language, occupation, home state, date of birth, and gender as the users' side information.

***(b)Movies' Video-Audio Information:*** To make our dataset multi-modal we required the video-audio information of movies along with the provided textual information. In general, a full-length Hindi movie has a duration of 120 minutes to 180 minutes. If its average frame rate in seconds is 24 then there are minimum $120 \times 60 \times 24 = 172,800$ frames in a full-length movie with 120 minutes of the audio file.

Processing the video and audio information of a full-length movie is very costly in respect of time and as well as space. Motivated by [6], we have considered the movies' short-length trailers rather than full-length movies. A movie trailer presents the overall theme of the movie in a very short period, in nearly about 2% of total frames of the full-length movie's total frame and with very small-length audio information.

The dataset has 615 Hindi movies' textual information but some of them are too old to find their trailers in the online streaming platform like YouTube[2], IMDb[3], RottenTomatoes[4] etc. and finally we were able to collect the trailers of 510 Hindi movies out of 615. So, in our dataset, we have 510 Hindi movies with their video-audio information extracted from trailers and textual information as described above.

The 510 Hindi movies have 753 users and a total of 16,667 user-movie pairs with their feedback. The user-movie pair is a triplet consisting of ***user-id, movie-id, and user's feedback***. In this Hindi Movies multi-modal dataset there are 1887, 6967, and 7813 user-movie pairs for the preferences *Dislike, Like, and Neutral/Not-Watched*, respectively. Fig.1b(i). shows that each user has given feedback on an average of 22 movies, in which 9 for Liked-Movies, 10 for Neutral/Not-Watched Movies, and 3 for Disliked Movies as shown in Fig.1b(ii)., Fig.1b(iii). and Fig.1b(iv). respectively. Fig.2 shows a data sample of Hindi movies' multi-modal Flickscore dataset.

## 4. Problem Statement

Here, we are dealing with a classification problem where of three classes to classify and our recommendation model tries to predict whether a user will dislike or like, or be neutral in giving his feedback when a candidate movie is recommended to him. In Fig.1a, the problem statement has been pictorially presented.

In this proposed model a user-movie pair is passed as input and as an output, it triggers one of the options among three classes of preference. We can formulate it as $F(concat(m_t, m_v, m_a)|user_m) = Pref_{user_m}$. Where $Pref_{user_m} \in (Dislike, Like, Neutral)$. We have tried to incorporate the multi-modal information of a movie while embedding a movie (m) and it is then passed with its user ($user_m$) to the proposed model to predict the preference of the pair. In our experiment, the three modalities (*Text(t), Video(v) and Audio(a)*) of the movie are concatenated and the function $F$ is learned to predict the preference of the user to the movies. Besides developing the model, the investigation of the model's performance in different scenarios and finding out the corner cases to cover in developing an RS model is another task of this work.

## 5. Proposed Methodology

We have introduced a model that predicts the user's preference for a movie after analyzing its two inputs, the user embedding and the movie embedding. Before performing the task, the model requires the appropriate embeddings of both, the user and the movies. Here we have used three different modalities of movies, ***Text, Video, and Audio*** to incorporate the multi-modal aspects in RS. For generating the appropriate embeddings of the movie's different modalities, we have considered modality-specific techniques as described in the next.

[2] **YouTube:** https://www.youtube.com/
[3] **IMDb:** https://www.imdb.com/
[4] **RottenTomatoes:** https://www.rottentomatoes.com/

***(a)Generating embeddings:*** The multi-modal information of movies is embedded first to make it fit the model

*(i)Text embedding:* We have two forms of textual information of movies, ***(1) Structured:*** *the meta information like Genre, Director-Name, Movie-Name, IMDb-rating, etc. and* **(2) Non-Structured:** *the summarised story-line of movies written in purely English sentences.*

Among all the structured textual meta-information of movies, the genre and IMDb information are used in a very straight way to embed. In this considered dataset, there are 14 different genres and as the movies in the dataset belong to multiple genres, the genre embedding of the movie is multi-hot encoding with dimension 14. The IMDb rating of the movie is a single value between 0.5 to 10 range and its normalized value is taken as its embedding.

The movie name and director name of Hindi movies are mostly followed the Hindi naming convention. Here, we first *(1) Translate the Hinglish(Hindi+English) word into Hindi word* using the open-source Python library **googletrans 3.0.0**[5], and then *(2) Generate the embedding of the Hindi word*. For generating the Hindi word embedding, we used the open-source word-embedding and sentence-classification library **fastText**[6] which is trained on 157 different language data collected from **Common Crawl**[7] and **Wikipedia**[8]. It generates two separate vectors of 300 dimensions for each movie name and director name.

The non-structured textual story-line information of the movie is in the English language. So, using Sentence-BERT (SBERT)[17] a vector 384-dimensional vector for the summary is generated. Unlike BERT [7], SBERT uses a siamese-architecture [9], where it contains two BERT architectures that are essentially identical and share the same weights.

So, we have embeddings of a 14-dimensional genre, one-dimensional IMDb rating, 300-dimensional Movie-name, 300-dimensional Director-name, and 384-dimensional storyline. Finally, our movie's textual embedding dimension is (14 + 1 + 300 + 300 + 384) = 999.

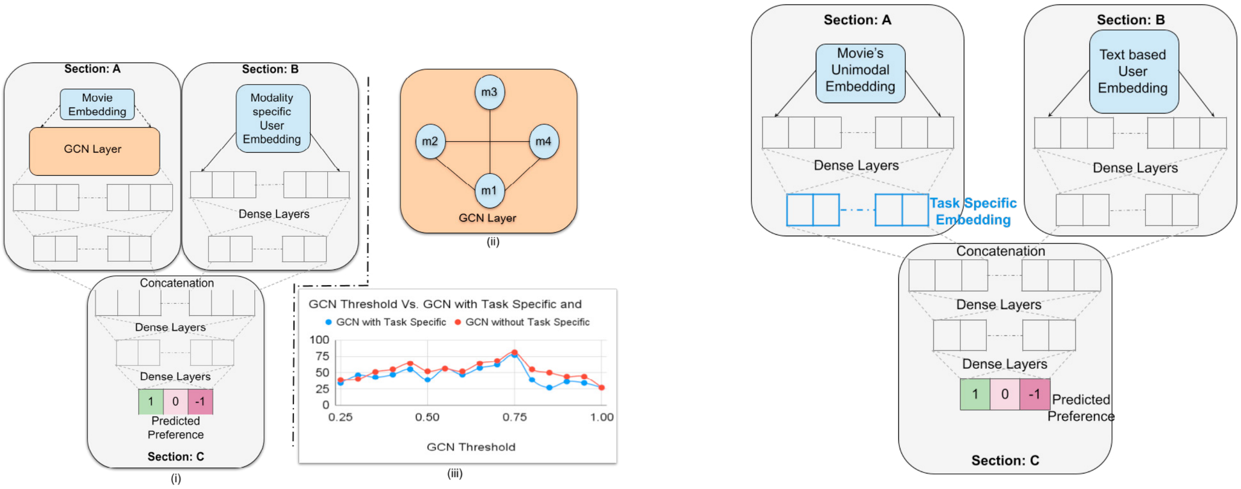
*(ii)Video embedding:* The video-audio information is extracted from the movie's trailer and scrapped from online platforms like YouTube, IMDb, etc. Video embedding generation is not quite straightforward. It also follows two steps, ***(1) Frame Extraction and (2) Embedding Generation***. In the frame extraction step, we extracted 16 frames at equal intervals from the trailer. Once the frames have been extracted, they are resized to 224x224x3 and the pixel values of the images are normalized. After extracting and normalizing the resized frames of the trailer, the video embedder is considered for embedding generation. For this purpose, the TimeSformer [3], a powerful convolution-free approach, trained on a large video-dataset of videos(K600 dataset) has been used following the transfer learning for video genre classification.

We made some modifications to the TimeSformer model. First, we removed the classification layers of the existing model and added a new layer with 512 neurons. Next, we added a classification layer with 14 neurons, which were used to perform multi-label classification. This was necessary as video genre classification is often a multi-label problem, where a video can belong to multiple genres. Finally, our modified TimeSformer model is trained with the movie's 16 frames in the input layer and 14-dimensional multi-label genre prediction in the output layer. During training, the weights of the last two layers are trained by keeping the rest of the layers' weight non-trainable as done in transfer learning and once the training is done, the embedding generated by the second last layer of dimension 512 is considered as the video embedding of our proposed main model.

*(iii)Audio embedding:* We have 510 Hindi movie trailers with audio files. The audio embedding generation of these movies is quite straightforward. Here we extracted the audio file in .wav format from the trailer first and then used **wav2vec2**[2] framework for the embedding generation. wav2vec2 is one of the current state-of-the-art self-supervised training models and has been used as an audio feature extractor in our experiment. It converts the .wav file to a vector of a dimension of 512 and hence we have a 512-dimensional feature vector of each movie's audio information.

*(iv)Movie representation:* The movie's representation depends on the considered modality. In uni-modality, movie embedding is nothing but the embedding of the text or audio, or video itself. But in bi-modality, the movie embedding is the concatenation of considered two modalities, and in tri-modality, all modalities are concatenated. So the dimension of the movie embedding depends on the considered modality only.

[5] `https://pypi.org/project/googletrans/`
[6] `https://fasttext.cc/`
[7] `https://commoncrawl.org/about/media/`
[8] `https://www.wikipedia.org/`



(a) Main Model.

(b) Task-Specific (TS) Embedding.

Fig. 3: **(a)Main model:** (i)The GCN Based main model architecture, (ii) example of movies (m1, m2, m3, m4) connectivity in GCN Layer based on cosine similarity among them, (iii) GCN-Threshold Vs. GCN-Based TS and Non-TS Model's accuracy. **(b)Task-Specific (TS) Embedding** model structure

*(v)User representation:* The user representation is the feature-wise average of all movies liked by the user. In our dataset, a user has three options in his preference feedback, *Dislike, Like, and Neutral* but while generating the user's embedding, only the movies liked by the user are considered and their feature-wise average is the user's representation. So here, the user embedding dimension is as same as the movie's dimension which also depends on the considered modality.

*(vi)Preference encoding:* In our dataset, there are three classes from the preference option, namely *Dislike, Like, and Neutral*. So, it requires to be a three-class classification problem while training the model to predict the preference and hence the preferences are encoded in the one-hot encoding manner with dimension three.

***(b)GCN in movie representation:*** For representing a movie by correlating it with other movies' information, the semi-supervised learning model, Graph Convolutional Network (GCN)[12] has been coupled up with our main model as shown in Fig.3a(i). Here the network is a shallow GCN with a single layer that takes the generated embedding of movies' information as input, uses the weighted average aggregation among the nodes, and passes its generated representation of movies' information to the next layer of the proposed model. The Binary Adjacency Matrix has been used here and those two nodes have been considered connected whose cosine-similarity score reaches the specified threshold (termed as GCN-Threshold in Fig.3a(iii)). Several ablation studies have been performed in setting the hyper-parameter GCN-Threshold. Fig.3a(iii) shows the corresponding accuracy of the GCN-based Task-Specific (**TS**) and Non-Task-Specific (**N-TS**) models. It is observed from the experiments that the GCN-Threshold at value 0.75 gives the better adjacency matrix for the GCN model and hence the value has been set at the threshold. Fig.3a(ii). presents a simple schematic diagram of 4 movies' connections in the GCN layer just for better visualization. This layer is trained with the main model and the movies' updated embeddings are passed to the deeper layers of the model for predicting the preference.

## 6. Experiments and Results

This section details the experimental setup and the result of the four different combinations of our proposed model.

***(a)Experimental setup:*** After generating the embeddings of different modalities individually, we performed our experiment in two different settings, *Task-Specific(TS) embeddings and Non-Task-Specific(N-TS) embeddings*. For N-TS, the embeddings of the modalities are used directly in the proposed main model but for TS, we need to introduce the

Table 1: Experimental setup of different models

| **Model** | **# Layers<br>(with dimension)** | **Activation<br>Function** | **Optimizer** | **Loss Function** | **# Training<br>Data** | **# Test Data** | **Vector<br>Dimension** | **Remarks** |
|---|---|---|---|---|---|---|---|---|
| **Text Embedding<br>(Task Specific)** | Sec. A: 3 (999, 666, 333)<br>Sec. B: 2 (999, 666)<br>Sec. C: 4 (755, 1024, 128, 3) | (tanh)<br>(tanh)<br>(tanh, sigmoid, sigmoid, softmax) | Adam | Categorical<br>cross-entropy | 12,000<br>triplets | 4,667<br>triplets | Input: 999, 999<br>Output: 3<br>Extracted Embedding: 128 | Test<br>Accuracy:<br>89.47% |
| **Video Embedding<br>(TimerSformer)** | TimeSformer (last layer eliminated),<br>512, 14 | sigmoid at last layer | Adam | Categorical<br>cross-entropy | 408 movies | 102 movies | Input: (224X224) with 16 frames<br>Output: 14<br>Extracted Embedding: 515 | Test<br>Accuracy:<br>89.71% |
| **Video Embedding<br>(Task Specific)** | Sec. A: 3 (512, 256, 128)<br>Sec. B: 2 (999, 666)<br>Sec. C: 4 (794, 1024, 128, 3) | (tanh)<br>(tanh)<br>(tanh, sigmoid, sigmoid, softmax) | Adam | Categorical<br>cross-entropy | 12,000<br>triplets | 4,667<br>triplets | Input: 512, 999<br>Output: 3<br>Extracted Embedding: 128 | Test<br>Accuracy:<br>67.34% |
| **Audio Embedding<br>(Task Specific)** | Sec. A: 3 (512, 256, 128)<br>Sec. B: 2 (999, 666)<br>Sec. C: 4 (794, 1024, 128, 3) | (tanh)<br>(tanh)<br>(tanh, sigmoid, sigmoid, softmax) | Adam | Categorical<br>cross-entropy | 12,000<br>triplets | 4,667<br>triplets | Input: 512, 999<br>Output: 3<br>Extracted Embedding: 128 | Test<br>Accuracy:<br>73.45% |
| **Main Model** | Sec. A: 4 [d_m, d_m, 2/3*(d_m), 4/9*(d_m)]<br>Sec. B: 2 (d_u, 2/3*(d_u))<br>Sec. C: 4 (con, 1024, 128, 3)<br>Here,<br>d_m = movie embedding dimension<br>d_u = user embedding dimension<br>con = [(4/9)*d_m + 2/3*(d_u)]/2 | (tanh)<br>(tanh)<br>(tanh, sigmoid, sigmoid, softmax) | Adam | Categorical<br>cross-entropy | Training-Testing split<br>= 80:20<br>with five fold cross validation<br>and<br>Training-Validation split<br>= 90: 10 | | Input: d_m, d_u<br>Output: 3 | learning_rate = 0.001,<br>$\beta_1 = 0.9$,<br>$\beta_2 = 0.999$,<br>$\epsilon = 10^{-7}$ |

Task-Specific model as shown in Fig.3b. In this TS model, the modality-specific movie embedding (text embedding, video embedding, or audio embedding) is passed through some dense layers and then concatenated with the user embedding. Here the user embedding is generated from the textual modality of movies. The concatenated embedding is then passed through some dense layers and finally, the output layer tries to predict the corresponding preference (-1: Dislike, 1: Like, 0: Neutral) of the user-movie input pair. Once the model is trained, the **Section: A** is taken as the TS-Embedding generator, and the embedding from its last layer (Task Specific Embedding layer in Fig.3b) is considered the TS embedding of the modality. In this way, the TS embedding is generated.

*(i)Embedding generation setup:* The generation of N-TS embedding is straightforward as described in the Embedding generation section, but for TS embedding, the TS-model shown in Fig.3b is used. For generating the TS embedding of Text, Video, and Audio, the corresponding N-TS embeddings with text-based user embedding are passed to the TS model, and the corresponding TS embedding is taken out. This model takes the N-TS text embedding as input and generates the TS embedding of it. For presenting in a better way, the TS-model has been partitioned into three sections namely *Section: A, Section: B and Section: C*. In Table 1 the experimental setup for N-TS video embedding generation, the section-wise and modality-wise TS embedding generation have been tabulated.

*(ii)Main model training setup:* The main model presented in Fig.3a(i) is also partitioned into three sections where *Section: A* takes the movie embedding and passes it to the GCN layer. The output of the GCN layer is then passed to some dense layers and the last layer output of this section is concatenated with the last layer output of *Section: B*. The user embedding generated from the same modality of the movie embedding is fed into *Section: B* as input and after passing through some dense layers is concatenated with *Section: A*'s output to form the input of *Section: C*. In *Section: C*, a sequence of some dense layers ended in a 3-dimensional classification layer. The section-wise training setup for the main model has been detailed in Table 1. Here the dimension of the model's inputs are the same and the dimension value depends on the modality of the movie embedding used.

*(iii)Train test split:* The main model is trained in five-fold Stratified cross-validation but before that, the user-movie-preference triplet for the **Dislike** class has been up-sampled to 7,000 from 1,887 to make the dataset balanced. Here the split of training-testing is 80:20 whereas the training: validation split is 90:10. For the other models, the training-testing split is almost 70:25 and has been tabulated in Table.1.

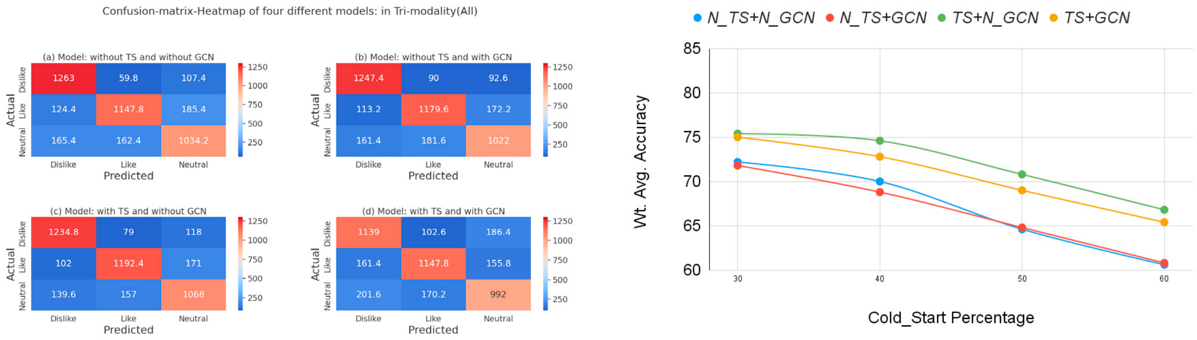
*(iv)Performance metrics:* The classification based models' performances are measured with Precision(**P**), Recall(**R**), F1-score(**F1**) and their weighted average (**Wt. Avg.**) as shown in Tables.2, 3, 4, 5, 6 with the mentioned metrics abbreviations. In these tables, the *Dislike, Like, and Neutral* classes are termed C0, C1, and C2, respectively. For space limitation, the modality names are also abbreviated as **T, A, V** for *Text, Audio, and Video*, respectively. In the same way, **TS, N-TS, N-GCN, TA, TV, AV and All** are the respective abbreviations of *Task-Specific, Non-Task-Specific(model without Task-Specific), Non-GCN(model without GCN), (Text+Audio), (Text+Video), (Audio+Video), and (Text+Audio+Video)*. The results shown in these tables are the average of five-folds of the best model trained in 32 epochs. Irrespective of TS or the main model, all are the best model's results of 32 epochs and statistically significant. The highest values in the comparisons are made bold to highlight and the comparison is based on the F1-score.

Table 2: **(N-TS and N-GCN Model):** modality (uni-modal and bi-modal) and class-wise Precision, Recall, F1-score and weighted averages

| | **T Modality** | | | | **V Modality** | | | | **A Modality** | | | | **TV Modality** | | | | **TA Modality** | | | | **VA Modality** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Class** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** |
| **P** | 82 | **85** | 76.2 | 81.2 | 82.6 | 80.8 | 77.7 | 80.4 | **82.8** | 83.8 | **78.2** | **81.6** | 82.4 | **84.2** | **79.6** | 82 | 79.8 | 83.4 | 79.2 | 80.6 | **83.6** | 84 | 78.8 | **82.2** |
| **R** | 88.40 | 76.6 | **77** | 80.6 | 85.2 | **81.2** | 74 | 80.6 | **89.6** | 80 | 75 | **81.6** | **89.2** | **79.8** | 77 | **82** | **89.2** | 78.4 | 74 | 80.6 | 89.2 | **79.8** | **77.2** | **82** |
| **F1** | 85.2 | 80.4 | **76.6** | 80.6 | 84 | 81 | 75.8 | 80.2 | **86** | **81.8** | **76.6** | **81.6** | 85.4 | **81.8** | **78** | 81.8 | 84 | 80.8 | 76.6 | 80.6 | **86.2** | **81.8** | 77.8 | **82** |

Table 3: **(N-TS and GCN Model):** modality (uni-modal and bi-modal) and class-wise Precision, Recall, F1-score and weighted averages

| | **T Modality** | | | | **V Modality** | | | | **A Modality** | | | | **TV Modality** | | | | **TA Modality** | | | | **VA Modality** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Class** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** |
| **P** | 75 | **83.2** | **74.8** | **77.6** | **76.6** | 76.4 | 69.4 | 74 | 73.8 | 78.8 | 72 | 74.8 | **80** | 80.6 | **77.8** | 79.4 | 79.8 | **81.8** | 77.6 | **79.8** | 78.8 | **81.8** | 73 | 78.2 |
| **R** | **85.2** | **74** | **72.2** | **77.2** | 83.4 | 73.2 | 65.8 | 74.2 | 84.6 | 72.4 | 67 | 74.6 | 85.2 | **79** | 73.2 | 79.4 | **86.6** | 78.4 | 73.4 | **79.6** | 85.2 | 75 | **73.8** | 78 |
| **F1** | 79.4 | **78** | **73.6** | **77** | **80.2** | 74.8 | 67.6 | 74 | 79 | 75.2 | 69.6 | 74.6 | 82.4 | **79.8** | 75 | **79.2** | **83** | **79.8** | **75.6** | 79.6 | 82 | 78.2 | 73.4 | 78 |



(a) Confusion-Matrix-Heatmap.

(b) Cold Start.

Fig. 4: **(a)Confusion-Matrix-Heatmap** of four different models in (TVA) modality. **(b)**The weighted average accuracy of different combinations of the model with All-modality in different cold start percentages

Table 4: **(TS and N-GCN model):** modality (uni-modal and bi-modal) and class-wise Precision, Recall, F1-score and weighted averages

| | **T Modality** | | | | **V Modality** | | | | **A Modality** | | | | **TV Modality** | | | | **TA Modality** | | | | **VA Modality** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Class** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** |
| **P** | **82.6** | **85.5** | **81** | **83.4** | 80.6 | 83.2 | 74.2 | 79.4 | 76.2 | 79.4 | 74.4 | 76.6 | **84.6** | 84.8 | **80.4** | **83.4** | 83.2 | **86.2** | 77.8 | 82.6 | 80.4 | 83.6 | 80.2 | 81.6 |
| **R** | **90.6** | **80** | **78** | **83** | 86.2 | 76.6 | 75.2 | 79.2 | 82 | 76 | 71.2 | 76.6 | 89.6 | **81.4** | 78.8 | **83.2** | 88 | 79.2 | **80.2** | 82.4 | **90.8** | 79.8 | 73.4 | 81.4 |
| **F1** | **86.4** | **82.8** | **79.2** | **83** | 83.2 | 79.6 | 74.4 | 79.2 | 79 | 77.6 | 72.8 | 76.6 | **86.8** | **83** | **79.4** | **83.2** | 85.6 | 82.6 | 79.2 | 82.4 | 85.2 | 82 | 76.6 | 81.2 |

***(b)Result analysis:*** In our model, we have two approaches, *(TS)* embedding generation and *(GCN)*.

In Tables. 2, 3, 4, 5, 6 the modality-wise performance matrices of different models have been tabulated.

*(i)N-TS and N-GCN model:* The analysis for the model where neither TS nor GCN has been used is reported in Table. 2. In uni-modal, the Audio (**A**) modality has the overall highest score whereas Video+Audio (**VA**) shows the best performance in bi-modality and also outperforms the uni-modality.

*(ii)N-TS and GCN model:* Table.3 reports the performance of the model with GCN but no TS. Here, uni-modal Text (**T**) and bi-modal Text+Audio (**TA**) are the best-performing modalities. Though the bi-modal also shows a better score here than the uni-modal, the model is not giving better results than that of the (N-TS and N-GCN) model. Hence, GCN is not helping in predicting the preference in the proposed model.

*(iii)TS and N-GCN model:* A model with TS but without GCN is another combination of our investigations and its results have been reported in Table. 4. Here also **T** is the best modality in uni-modal but for bi-modal, Text+Video(**TV**) is the best modality. Generally, in this model, the performance of the bi-modality is not too higher than that of the uni-modality but its best modality outperforms the model (N-TS and N-GCN). Hence TS has an effective role in the proposed model.

Table 5: **(TS and GCN Model):** modality (uni-modal and bi-modal) and class-wise Precision, Recall, F1-score and weighted averages

| | **T Modality** | | | | **V Modality** | | | | **A Modality** | | | | **TV Modality** | | | | **TA Modality** | | | | **VA Modality** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Class** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** |
| **P** | **78** | **81.2** | **77.2** | **79** | 72 | 76.8 | 73.8 | 74.4 | 68 | 75.4 | 67.2 | 70.2 | 75.2 | 78.8 | 75 | 76.4 | **76** | **79.6** | **79.4** | **78.2** | 75.2 | 78.8 | 75 | 76.4 |
| **R** | 84 | **78** | **74** | **78.8** | **85.6** | 74 | 61.6 | 74.4 | 73.6 | 73.4 | 62.8 | 70.2 | 76.8 | 79.2 | **71.8** | 75.8 | **83.8** | **80.2** | 70 | **78.2** | 76.8 | 79.2 | **71.8** | 75.8 |
| **F1** | **80.8** | **79.6** | **75.8** | **78.8** | 78 | 75.2 | 67 | 73.6 | 70.6 | 74.4 | 64.6 | 70 | 76 | 78.8 | 73 | 75.8 | **79.8** | **80** | **74.2** | **78** | 76 | 78.8 | 73 | 75.8 |

Table 6: **(Model with All-Modality:** model and class-wise Precision, Recall, F1-score and weighted averages

| | **N-TS and N-GCN** | | | | **N-TS and GCN** | | | | **TS and N-GCN** | | | | **TS and GCN** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Class** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** | **C0** | **C1** | **C2** | **Wt. Avg.** |
| **P** | 81.4 | **83.8** | 78 | 81.2 | 82 | 81.4 | **79.4** | 81 | **83.6** | 83.6 | 78.8 | **82** | 75.8 | 80.6 | 74.8 | 77 |
| **R** | **88.4** | 78.8 | 75.8 | 81 | 87.4 | 80.6 | 74.8 | 81 | 86.4 | **81.2** | **78.2** | **82** | 79.8 | 78.4 | 72.8 | 77 |
| **F1** | 84.6 | 81.2 | 77 | 81.2 | 84.4 | 81 | 76.7 | 80.8 | **85** | **82.2** | **78.6** | **82** | 77.6 | 79.4 | 73.4 | 77 |

*(iv)TS and GCN model:* Like the (N-TS and GCN) model, this model also has **T** and **TA** as its best uni-modal and bi-modal respectively. In this model, the bi-modality is not being proved better than the uni-modal and the model is not also outperforming both models presented in Tables 2, 4. Hence, the presence of GCN is not helping the model in its objective but TS might be a better choice in future research.

*(v)All-modality Vs. all models:* Here the three modalities (*Text+Video+Audio(**TVA**)*) have been jointly termed as **All-modality**. In Table.6 the results of all four combinations of the model are shown and the (TS and N-GCN) model is giving the best result. This table also shows that the Tri-modality is also not outperforming the bi-modality. Fig.4a shows the heatmap representation of the five-fold-average confusion matrix of the four model's in the All-modality setting and from Fig.4a(d), the True-Positive distribution of three classes proves again the presence of GCN is not effective for the model.

*(vi)Cold start movies handling:* The performances in movies' cold start problems are also scrutinized in our experiment. A certain percentage of movies are made new or cold start movies to the model during testing. This is done by not passing some movies to the model during training and asking the trained model to predict the users' preferences of them. 30%, 40%, 50%, and 60% movies of our dataset are made cold start movies, and the performances of all four combinations of the model are checked. In Fig.4b, it is shown that even at 60% cold start percentage, all the models' weighted average accuracy is not less than 60% and the (TS and N-GCN) models perform the best in this scenario also.

***(c)Error analysis:*** From the above analysis, besides the effect of TS and GCN, it is also observed that the joint modality is not working well consistently in the proposed model. This could be the reason of many factors which need to be rectified. The reasons could be, **(i) Video Frame Extraction Technique:** Here a certain number of frames at equal intervals are extracted from the movie's trailer and this might be a cause of losing the key-frames that represent the overall theme of the video. **(ii) Modality Fusion Technique:** The considered concatenation fusion technique is quite simple. In future research, some new processes of fusion can be incorporated. **(iii) Low Volume Dataset:** We have only 510 movies with multi-modal information and 753 users. This low-volume dataset is not helping properly both the GCN as well as the main deep learning model in finding the correlations between user-movie pair and user preference. **(iv) GCN-Threshold:** It might be a reason that the Cosine-Similarity based threshold is not able to form the appropriate Adjacency matrix for the graph. **(v) Movie's Textual Information:** The textual summary is a foreign material of the movies written by any third party with his observation whereas the audio-video property is the actual and raw material of the movie. Considering the movie's Transcripts rather than the summary might be a good and logical choice in the future.

## 7. Conclusion and Future Works

In this proposed experiment, an investigation of a movie recommendation model with two modules, Task-Specific (TS) embedding generation and Graph Convolutional Network (GCN) has been reported. The model takes user embedding and different modality embeddings of movies like Text, Video, and Audio as input to predict the preference. For the experiments, we have used Flickscore, an Indian Language-based Hindi movie dataset and the results of different combinations of two modules of the model conclude that the model with TS only is performing well but the presence of GCN is not effective for this model.

Alternate information and techniques for generating embeddings of movie information would be incorporated into our future research. For text information, incorporating the movie's transcripts, and for video, introducing a new approach for key frame extraction is on our to-do list.

To support the novel idea of GCN, increasing the volume of the dataset, and applying different threshold methods for creating an adjacency graph would be another approach in the future. This investigation on the new model and the new dataset does not have any competitive work in state-of-the-art. The conclusion of the investigation has opened the corner cases to cover in building an Indian language-based movie recommendation system. Developing the Recommendation system by covering all the corner cases found in this investigation and comparing them with the state-of-the-art would be our primary objective in future work.

## Acknowledgment

The authors gratefully acknowledge the support from Sony Research India for conducting this research.

## References

[1] Agarwal, P., Verma, R., Majumdar, A., 2018. Indian regional movie dataset for recommender systems. arXiv preprint arXiv:1801.02203 .

[2] Baevski, A., Zhou, Y., Mohamed, A., Auli, M., 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. Advances in neural information processing systems 33, 12449–12460.

[3] Bertasius, G., Wang, H., Torresani, L., 2021. Is space-time attention all you need for video understanding?, in: ICML, p. 4.

[4] Chakder, D., Mondal, P., Raj, S., Saha, S., Ghosh, A., Onoe, N., 2022. Graph network based approaches for multi-modal movie recommendation system, in: 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE. pp. 409–414.

[5] Choudhury, S.S., Mohanty, S.N., Jagadev, A.K., 2021. Multimodal trust based recommender system with machine learning approaches for movie recommendation. International Journal of Information Technology 13, 475–482.

[6] Deldjoo, Y., Elahi, M., Cremonesi, P., Garzotto, F., Piazzolla, P., Quadrana, M., 2016. Content-based video recommendation system based on stylistic visual features. Journal on Data Semantics 5, 99–113.

[7] Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 .

[8] Feng, C., Liu, Z., Lin, S., Quek, T.Q., 2019. Attention-based graph convolutional network for recommendation system, in: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE. pp. 7560–7564.

[9] Guo, Q., Feng, W., Zhou, C., Huang, R., Wan, L., Wang, S., 2017. Learning dynamic siamese network for visual object tracking, in: Proceedings of the IEEE international conference on computer vision, pp. 1763–1771.

[10] Huang, Z., Chung, W., Ong, T.H., Chen, H., 2002. A graph-based recommender system for digital library, in: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries, pp. 65–73.

[11] Kaklauskas, A., Gudauskas, R., Kozlovas, M., Peciure, L., Lepkova, N., Cerkauskas, J., Banaitis, A., 2016. An affect-based multimodal video recommendation system. Studies in Informatics and Control 25, 6.

[12] Kipf, T.N., Welling, M., 2016. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 .

[13] Lam, X.N., Vu, T., Le, T.D., Duong, A.D., 2008. Addressing cold-start problem in recommendation systems, in: Proceedings of the 2nd international conference on Ubiquitous information management and communication, pp. 208–211.

[14] Pazzani, M.J., 1999. A framework for collaborative, content-based and demographic filtering. Artificial intelligence review 13, 393–408.

[15] Pingali, S., Mondal, P., Chakder, D., Saha, S., Ghosh, A., 2022. Towards developing a multi-modal video recommendation system, in: 2022 International Joint Conference on Neural Networks (IJCNN), IEEE. pp. 1–8.

[16] Reddy, S., Nalluri, S., Kunisetti, S., Ashok, S., Venkatesh, B., 2019. Content-based movie recommendation system using genre correlation, in: Smart Intelligent Computing and Applications: Proceedings of the Second International Conference on SCI 2018, Volume 2, Springer. pp. 391–397.

[17] Reimers, N., Gurevych, I., 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084 .

[18] Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G., 2008. The graph neural network model. IEEE transactions on neural networks 20, 61–80.

[19] Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y., et al., 2017. Graph attention networks. stat 1050, 10–48550.

[20] Yang, B., Mei, T., Hua, X.S., Yang, L., Yang, S.Q., Li, M., 2007. Online video recommendation based on multimodal fusion and relevance feedback, in: Proceedings of the 6th ACM international conference on Image and video retrieval, pp. 73–80.