Towards Developing a Multi-Modal Video Recommendation System

Sriram Pingali   Prabir Mondal   Daipayan Chakder   Sriparna Saha
Department of Computer Science and Engineering
Indian Institute of Technology Patna, India
1801cs37@iitp.ac.in, prrabirmondal@gmail.com, {daipayan_2011cs02, sriparna}@iitp.ac.in

Angshuman Ghosh
Sony Research India
angshuman.ghosh@sony.com

2022 International Joint Conference on Neural Networks (IJCNN) | 978-1-7281-8671-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/IJCNN55064.2022.9892382

***Abstract*—With the surge of digitized entertainment systems in recent years, the element of personalised experience for users gives a competitive edge to entertainment businesses. Hence, the importance of recommendation systems, in particular for video or movie recommendation, is evident. However, majority of the recommendation systems are primarily learned in supervised fashion on empirical data like user ratings. This approach has certain short comings, specifically the cold start problem and the data sparsity problem. Moreover, existing recommendation systems do not utilize multiple information associated with videos/movies like their textual summary, meta-data information, audio and video signals etc. With the increase in the multi-modal information processing in different fields of artificial intelligence, we have proposed the task of multi-modal recommendation system in the current study. For representing the movie/video, feature vectors generated from text modality, meta-data modality, audio and video modalities are concatenated. A novel knowledge graph based approach is applied for generating feature vectors from text modality. Finally, a Siamese architecture based deep learning technique is proposed to perform regression over similarities using multi-modal user-item embeddings. As there was no data set available for solving the task of multi-modal recommendation, we have also enhanced the existing benchmark Movie-lens 100k dataset with text, video, audio information and utilized that for performing experimentation. Experimental results establish the efficacy of using multi-modal information for movie embedding generation**

***Index Terms*—Recommendation system, multi-modality, personalized recommendation, knowledge graph, siamese network.**

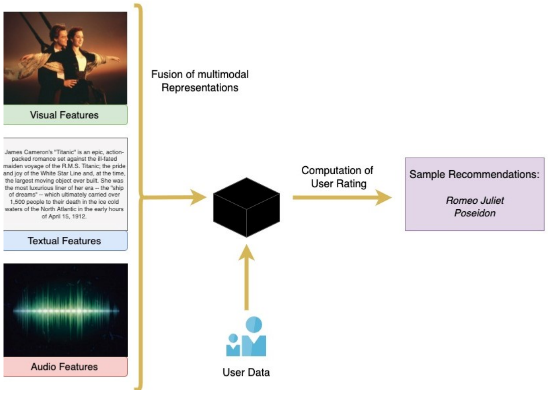
Visual Features
James Cameron's "Titanic" is an epic, action-packed romance set against the ill-fated maiden voyage of the R.M.S. Titanic; the pride and joy of the White Star Line and, at the time, the largest moving object ever built. She was the most luxurious liner of her era -- the "ship of dreams" -- which ultimately carried over 1,500 people to their death in the ice cold waters of the North Atlantic in the early hours of April 15, 1912.
Textual Features
Audio Features
Fusion of multimodal Representations
Computation of User Rating
Sample Recommendations:
Romeo Juliet
Poseidon
User Data

Fig. 1. Pictorial Representation of Problem Statement

I. Introduction

The explosive growth of web information and its content diversity has opened a challenging research area in digital information categorization, user's choice personification, and building a user-item recommendation system. With the increasing number of digital content platforms, there is an exponential rise in data being collected in the form of user behavior (user likes and dislikes). Utilizing these data in personalizing user experience gives a competitive edge to digital entertainment platforms and this helps in building a recommendation system which is one of the revenue-making strategies for the digital media service providing entities. Therefore, the popularity of recommendation systems in movies/videos is on the rise. Metadata like the movies watched by a user, his ratings to his watched movies, the movies' metadata, and user's other information help in developing a movie recommendation system where a new movie can be recommended to the user if the movie has a very close similarity with other liked movies of the user. However, due to the complexity of the task involved, combining exponentially rising users and content, it becomes increasingly difficult to accurately represent all the users, movies, and their properties from the user preferences, item features, user-item interactions, and sometimes user meta-data or temporal-spatial data.

Methods used for recommendation systems (RS) can be broadly classified into three categories, namely Collaborative Filtering (CF) [1], Content-Based Filtering (CBF) [2], and Hybrid Models. CF uses existing rating information to predict the missing rating data. Here users are grouped according to the similarity of their preferences or ratings and the preferred items of users of the same group are recommended to all users belonging to the same group but have not availed of the items. In CBF, the contents of the liking-disliking products of a user are considered in selecting a new product for the recommendation. For example in movie recommendation, CBF approaches use user-item representations for predicting the ratings. For user, this data could include, user's age, gender, occupation information, etc. and for items, those could be genres, directors, cast, number of ratings, etc.

However, contemporary works in the field of recommen-

978-1-7281-8671-9/22/$31.00 ©2022 IEEE

dation systems mostly focus on utilizing user ratings and sometimes user-movie meta information to represent the data. Relying on the users' ratings in recommendation may lead to a cold start problem [3] when a user rates less movies or when less number of movies in the platform have been rated. So relying on the users' ratings in video recommendation is not always recommendable. Instead of the ratings, movies as an entity contains a lot of information in the form of various modalities. Some of these may include elements like frames of the movie, audio, textual descriptions, etc. The visual features of the movie's frames can be considered as the visual representation of the movie. But dealing with all the frames of a video of 2 hours average duration is really a time taking problem because of its large number frames. But the trailer of a movie is shorter in duration and its audio-video content summarizes of the whole movie. So considering the movie trailer rather than the whole movie in recommendation system leads to a better approach. To the best of our knowledge, none of the approaches of recommendation system in the existing literature focuses on using this multi-modal information of movie/video. In the current paper, we have introduced the problem statement of multi-modal recommendation system where multi-modal information of movie like audio, video, text information are utilized for generating an efficient movie representation without utilizing any rating information.

In our proposed method, we have proposed an unsupervised way of representing the movie after considering various attributes associated with it like the video-audio features from the movie's trailer, textual summary, metadata of the movies. This unsupervised representation can help in solving the cold start problem of the recommendation system. As there was no data set available containing multi-modal information of movies, we have extended the bench-mark **Movie-lens 100k Dataset** with textual, audio, and video features. We have experimented with two different ways of extracting the textual features (i) Sentence BERT [4] which is a transformer-based architecture for extracting good quality features from textual summary of movies (ii) knowledge graph-based representation: here first a knowledge graph is constructed based on the textual summary and meta-data information of movies. Finally, embeddings are generated from this knowledge graph to represent the textual summary of movies. VGG-ish [5], a pre-trained tool has been used for generating audio embeddings and frame extraction process followed by RESNET-50 [6] based feature generation are used for generating video embedding. Finally, these textual, audio, video, meta embeddings are concatenated to represent the movie embedding. The user embedding is the average of those movie embeddings liked by the user the most. Finally, a siamese based network [7] is proposed to detect the similarity between user embedding and movie embedding and provide a score/rating. The siamese network is first trained using the available rating values of the movie-user pairs and finally, the trained network can be used for generating rating value for any unknown movie for a given user.

The major contributions of the current paper are as follows:

- To the best of our knowledge, we are the first to introduce the problem of a multi-modal recommendation system where different aspects of movies like textual summary, video information, audio information, metadata are utilized for generating an efficient movie representation without considering any user-movie rating information (problem statement is pictorially represented in Figure 1).
- This multi-modal representation of movies can help in solving the cold-start problem of recommendation system because here, we are not relying on the rating information of a movie which is not available for a new/unknown movie. The information used for generating the movie embedding like textual summary, audio information, video information, meta-data information are available with any movie and they capture different aspects of a movie.
- An efficient knowledge graph-based technique is used for generating embedding from textual summary of the movies. Knowledge graph is also used to generate embedding from the meta-data information of the movies.
- A Siamese architecture based deep-learning framework is proposed to finally generate rating value given a movie representation and a user embedding.
- We have also enhanced the benchmark Movie-lens dataset with multi-modal information (textual summary, audio and video modalities) to enable its usage for multi-modal video recommendation task. We have made this Dataset[1] publicly available for further research and improvement in the video recommendation system.
- Experimental results establish the efficacy of using multi-modal information for movie representation in solving the rating prediction task.

II. Related Work

In this section, the relevant works related to the proposed method have been discussed. The rapid growth of digital information on the web and the rigorous demand of user specific or item specific showcasing on the digital platform put the recommender system in the list of researchers' top challenging research areas. Personalized recommendation by information processing has been getting immense interest for decades and video recommendation is one of its parts. In [8], authors described the system of recommendation of short length YouTube videos. The author in [9] proposed an opinion based recommendation system for movies by adapting the textual side information of users as well as movies. But the commonly used collaborative filtering approach of RS experiences the lack of diversity in recommendation due to sparseness in data. The remarkable performance of deep learning also has contributed to RS. The Probabilistic Matrix Factorization(PMF) [10] and Stacked Denoising Autoencoder [11] are used in [12]. In rating prediction for RS, items and users feature matrices are decomposed by Matrix Decomposition [13]–[15] technique in the studies of [16].

[1] https://github.com/SriramPingali/Multi-Modal-Recommendation-System

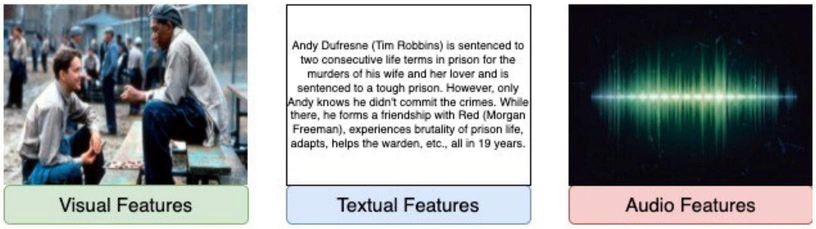
Andy Dufresne (Tim Robbins) is sentenced to two consecutive life terms in prison for the murders of his wife and her lover and is sentenced to a tough prison. However, only Andy knows he didn't commit the crimes. While there, he forms a friendship with Red (Morgan Freeman), experiences brutality of prison life, adapts, helps the warden, etc., all in 19 years.

Fig. 2. Example of a Data Sample in **Movie-Lens 100k**.

Here, a rating is predicted from low dimensional features. The content information and scoring matrix are merged by the collaborative filtering neural network used in [17] and the final prediction for recommendation is generated. The user's behavior, user's movie collection also in the form of text are modeled in [18], [19] and the deeper features are extracted by applying CNN. Recommender system relying on the user or item rating faces the critical issue of cold start problem. In overcoming the limitations, multi-criteria based approaches are explored in [20] for the rating prediction. As per our knowledge, most of the studies are on collaborative filtering based RS where user-item ratings and other textual information of them are utilized for generating deeper representative features. However, the rating based RS suffers from the data sparseness, cold start problem. The progress with the adaptation of deeper network and the remarkable impact of side information in the recommendation motivate us in exploring the essence of multi-modal model based approaches where not only the text information but also the audio, video and other meta information of items are included in the accurate prediction of recommendable items.

III. Dataset

In this section we shall discuss in detail regarding the novel dataset being introduced and the process involved in compiling the same.

A. *Collection of Dataset*

Since the task of multi-modal video recommendation system is quite novel, there is no dataset available that contains multi-modal movie data containing different features like video, audio, textual description etc. However, there are benchmark datasets like Movie-lens 100k dataset, that provides user rating information. Movie-lens dataset, in particular, has 100,000 user-item-rating tuples, sampled from a corpus of 943 users and 1682 movies. Additionally, some meta data information are also provided in the form of movie genres, user's age, gender, occupation etc. Nevertheless, for the sake of this paper we have enhanced the benchmark Movie-lens dataset with multi-modal information. Detailed explanations on how the dataset has been compiled are provided in the following sections. A sample of the data set is shown in Figure 2.

*1) **Movie Trailers**:* Considering that movies are large data sources spanning hundreds of minutes, it is quite difficult to extract any meaningful data off it, as it would lead to a huge overhead cost in computation and time. To overcome this, we propose a meaningful alternative. Movie trailers are meant to provide users of what to expect from a movie, and therefore contain the same theme that would run throughout the movie in a small gist spanning couple of minutes. And considering the fact that trailers are likely to be readily available in video-content platforms like YouTube, unlike full Movie videos, we generated a corpus from the same. YouTube data API was used to download movie trailers of 1682 movies. These videos are further segregated into frames and audio files.

*2) **Movie Summaries**:* Secondly, there are descriptions available for movies in sites like IMDb which provide story summaries. These summaries provide the essence of the movie in couple of sentences. And hence this is a very concise representation of the movie story itself. We use a web scraper with the help of IMDb API to scrap the *"Storyline"* section of IMDb pages of all these 1682 movies.

*3) **Movie Meta-Data**:* Finally, some meta-data of movies are provided by movie-lens which include movie genres, directors, cast, rating in IMDb, number of ratings etc. These information could also provide meaningful cues. IMDb scraper & API are used to extract information about Director, Cast, Rating, Number of ratings etc.

IV. Problem Statement

In this paper, we aim to solve the task of predicting user ratings for user-item pairs from the benchmark Movie-lens-100k dataset. Given a movie and user pair, we are required to predict the value (from 1 to 5) of what the user might rate the movie. To do so, the complete task shall be split into two sub-tasks, (i) User & Item representation generation and (ii) Regression over User-Item similarity.

After a thorough literature review, we identified the shortcomings of collaborative filtering approaches which mostly rely on user-item rating information for generating the representations and when the rating information is not available for any unknown movie/new movie then it leads to the issue of cold-start problem. Hence in this paper, we shall explore the possibility of utilising multi-modal information of movies to generate user-item representations. After generating the representations, the user-item pairs are used to perform regression over the user ratings depending on how (dis)similar the pairs are.

We shall now establish the notations that shall be followed throughout the paper. $U \in \mathbb{R}^{NX1220}$, $I^{Video} \in \mathbb{R}^{MX2048}$, $I^{Audio} \in \mathbb{R}^{MX128}$, $I^{Text} \in \mathbb{R}^{MX384}$, $I^{Meta} \in \mathbb{R}^{MX1220}$,

$R \in \mathbb{R}^{NXM}$. Where U is the user feature matrix, $I^{Modality}$ is the Movie/Item feature matrix under various modalities and R is the pairwise user-item rating matrix. And N, M are the number of users and movies, respectively. Formally, we structure the problem statement as, $R_{nm} = F(U_n, [I_m^{Video}, I_m^{Audio}, I_m^{Text}, I_m^{Meta}])$. Here F is the regression function that predicts user rating for a given user-item pair. The problem statement is pictorially represented in Figure 1.

V. Proposed Methodology

In this section, firstly we have described the embedding techniques utilized for extracting feature vectors from different modalities like text, audio and video. These embeddings are then concatenated to generate user and movie representations which are finally used in a siamese based network for predicting the rating for a user-movie pair.

A. *Generating Embeddings*

In our multi-modal RS, we have generated embedded feature vectors from text, video, audio, and metadata modalities. Two different techniques are explored for generating embedding vectors from text + metadata part: (a) knowledge-graph based approach (b) Sentence BERT based approach. For generating the knowledge graph based embedding, the textual form of the movie's summary sentences and the movie's meta information like genre, director, etc. are utilized. On the other hand, for generating the video and audio embeddings, the candidate frames of the movie trailer and its audio version in .wav file extension have been considered as the input of the embedding. The embedding generation techniques of different modalities are as follows.

*1) **Text Embedding**:* The textual part is the summary of the movie with multiple sentences. For generating semantically meaningful sentence embedding of every movie's summary sentences, the Sentence-BERT [4] model has been employed. The model transforms every sentence into $d^{1\times384}$ dimensional vector and we have feature matrix $Text^{Sentences}$. Finally, the feature wise mean values of all the sentences are computed for generating textual feature representation, I^{text} of every movie. $I_i^{text} := Mean[Text_{[1:S][i]}^{Sentences}]$, $i = 1$ *to* 384, $Text^{Sentences} \in \mathbb{R}^{S\times384}$ and, $I^{text} \in \mathbb{R}^{1\times384}$, where S is the number of sentences present in the movie summary that has been embedded by the transformer.

*2) **Video Embedding**:* Here, before generating the embedding, the candidate frames of a movie trailer are extracted in the form of images. As it is not possible to consider all the frames of the video which will, in turn, increase the time complexity of the system, we have proposed a candidate frame extraction technique to select a subset of frames for feature extraction. The candidate frame extraction technique and the corresponding embedding process are described below.

- **Candidate Frame Extraction** In our experiment we empirically concluded that the frame appears at the middle position in the trailer's sequential frame series is quite informative and the frames at the beginning as well as at the ending either contain a very less number of informative image objects or contain information about the movie's cast-and-crew names, movie title, sponsoring partners' advertisement, disclaimer, etc. in the form of text. This observation motivated us in discarding the frames from the beginning and ending position of the video and selecting the middle frame as the pivot frame of our candidate frames which will be used for generating the embeddings. For minimizing the number of candidate frames we have chosen only those frames having a distance of a specified timestamp from the pivot frame positioned in the middle. In our case, we have chosen the frames arriving in the interval of every one second before and after the pivot frames.

  $$t_{c_i} := |t_p \pm (i * fps)|, \quad i \in \mathbb{N}$$

  where t_p is the arrival time of pivot frame, t_{c_i} is the i^{th} candidate frame's arrival time and fps is frame rate in second for the video. The frame that has appeared in the t_cth second is considered as the candidate frame in our approach.
- **Video Frame Embedding** For bringing out the hidden visual features of the video, the pool layer of pretrained ResNet50 [6] image classification model (trained using ImageNet [21] dataset) has been used. For the reduction of time complexity, only the candidate frames extracted from the movie trailer have been employed for this visual feature extraction and embedding. Finally, for the visual representation of the video, the feature wise means of all $d^{1\times2048}$ dimensional feature vectors of all candidate frames are used: $F^{Frames} \in \mathbb{R}^{N\times2048}$, $I^{video} \in \mathbb{R}^{1\times2048}$. If F^{Frames} is the feature matrix of dimension $d^{N\times2048}$, containing all $d^{1\times2048}$ dimensional features of N number of candidate frames, then I^{video} is the $d^{1\times2048}$ dimensional visual representation of the video obtained after taking the feature wise mean from F^{Frames}.

*3) **Audio Embedding**:* The audio features of the movie trailer have also been incorporated in our proposed multi-modal RS. The audio file of .wav extension of the movie trailer is embedded in this part. For generating the audio embedding, the variant of VGG, VGGish model [5] has been utilized. In our proposed technique, the full audio of the movie trailer is sampled in every one second time stamp and every sample of this waveform is passed through the model where each sample is embedded into a 1×128 dimensional feature vector. Finally, the column wise feature mean values of all the samples are generated from the audio feature matrix, $Audio^{Samples}$, and the audio representation, I^{audio}, of the movie is generated. $I_i^{audio} := Mean[Audio_{[1:S_a][i]}^{Samples}]$, i=1 to 128; $Audio^{Samples} \in \mathbb{R}^{S_a\times128}$ and $I^{audio} \in \mathbb{R}^{1\times128}$, where S_a is the number of samples of the audio file.

*4) **Knowledge Graph based Embedding**:* : For generating embedding vector for text modality, we have utilized a novel knowledge graph based embedding technique. Firstly a knowledge graph is constructed from the textual information available and then an embedding is generated from that. The semantic network also known as knowledge graph based network [22] defines the relationships among the real-world entities. Two types of textual meta information are used for this graph, 1) tabulated meta information like movie name, director, genre, etc., and 2) information in the form of a sequential descriptive sentences. The formation of the semantic network using the tabulated data is quite straight forward as shown in Fig. 3, but for the descriptive information, the part-of-speech (POS) information of the sentences are considered as the entities of the graph. Using the open source library Spacy[2] of NLP model, the POS present in the sentences are extracted in building the graph. In our experiment, we have considered the first three simple sentences of the story summary and its subject, verb, object form the entities of the network. In Fig. 4. a single path knowledge graph based network has been presented for the sentence "John Hammond invited four individuals." where the subject, verb and object are the head, root, and tail of the network, respectively.

[3]

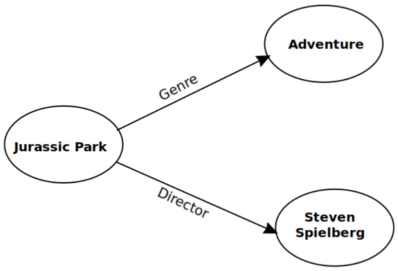

Fig. 3. Example of Knowledge Graph Based representation of a **tabulated meta information**.

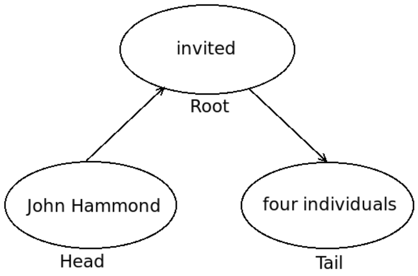

Fig. 4. Knowledge Graph Based representation of a sentence **" John Hammond invited four individuals"**.

In this graph based embedding, the distance information of the head-root-tail edge is evaluated after the word entities of the graph are transformed into word2vec representation [23].

$$I^{meta} = ||h + R - t|| \tag{1}$$

To elicit the knowledge graph based features of the textual meta information, the difference in distance between the head+root and the tail as shown in Eq.1 is calculated. So, for all the first three simple sentences from the textual summary of the movie and the tabulated meta information of it, the head(h), root(R), and tail(t) are fit to the formula defined in Eq.1. Hence we have $d^{1\times 1220}$ dimensional feature vector I^{meta} after concatenating all the meta information of the item/movie.

We previously establish our problem statement as a regression task on user ratings by identifying the (dis)similarity between user representation and item representation. We experimented with a popular deep learning based technique, Siamese Architecture which specializes in capturing similarities between inputs from same vector space (in this case, users and movies). However, in order to do so, we need to generate accurate user and item representations.

Our model contains three main components: Modular Encoders, Siamese Network and Rating Predictor.

B. *Modular Encoders*

From previous sections, we have seen how embeddings from different modalities are collected. However, these embeddings in raw format are highly variant in length and properties. Therefore, it is imperative to process these embeddings into same sized encoding. We achieve this by using modular encoders for each modality, i.e., we used 4 encoders, namely the Video Encoder Module, Audio Encoder Module, Text Encoder Module, Meta Encoder Module. After post processing, each modality encoding shall be of same length. Formally, the encoder operations can be represented as the following. $i_m^{Video} = Enc^{Video}(I^{video})$, $i_m^{Audio} = Enc^{Video}(I^{audio})$, $i_m^{Text} = Enc^{Video}(I^{text})$, $i_m^{Meta} = Enc^{Video}(I^{meta})$. Where, encoders are fully-connected layers meant to condense the raw embeddings to same vector space. Following are their specifications.

Enc^{Video} = Dense(2048 → 1600 → 500 → 200)
Enc^{Audio} = Dense(128 → 256 → 200)
Enc^{Text} = Dense(384 → 100 → 200)
Enc^{Meta} = Dense(1220 → 100 → 200)

Therefore, $i^{Modality} \in \mathbb{R}^{MX200}$ is the latent feature matrix for the movies in the given modality. These notations are considered further on.

1) ***Movie Representation:*** Movie properties could be captured by all the various modalities involved, therefore, various combinations of the modality latent vectors are considered to be the movie embeddings. For example, consider a movie M. The movie representation of M can be formulated as follows.

$$i_m = Concatenate([i_m^{Video}, i_m^{Audio}, i_m^{Text}, i_m^{Meta}])$$

It is important to note that, during the experimentation part, item representation sometimes involved only few modalities and not all. This is done so as to comprehend the impact of each modality on the result.

2) ***User Representation:*** While most of the existent works in recommendation systems use user ratings to represent users, we hypothesise that we can do the same through a pooling of the movies that he/she likes. For this experiment, we consider the baseline pooling mechanism to be a simple average.

[2] https://spacy.io
[3] https://grouplens.org/datasets/movielens/

Specifically, we consider the user representation to be the mean of meta-data embeddings of all the movies that the particular user has rated 4 or above. We have considered meta-data as the most appropriate modality for the user because, unlike video and audio that are specific to the movie, genre and director information with textual description might be more representative of users' preferences. This is proven by the better performance of meta data modality compared to other modalities in our experiments.

Subsequently, the user representation vector is passed through a modular user encoder that encodes the user representation into same size as item encodings. $U_n = \frac{\sum_m r_{nm}.i_m^{Meta}}{\sum_m r_{nm}}$ where i^{Meta} is the encoded meta data feature matrix of the movies. And r is the binarised rating matrix where,

$$r_{nm} = \begin{cases} 0, & \text{if } R_{nm} <= 3 \\ 1, & \text{if } R_{nm} > 3 \end{cases}$$

To extract latent features from user embeddings, $u_n = Enc^{User}(U_n)$ where, Enc^{User} = Dense(1220 $\rightarrow$ 1024 $\rightarrow$ 800)

C. *Siamese Network*

Siamese Neural Networks were first introduced in the early 1990s by Bromley and LeCun [7] to solve signature verification as an image similarity problem. Siamese architectures are popular for their ability to capture similarities in feature vectors of input pairs. It has two input fields for comparing two patterns and an output node that represents the similarity amongst the two patterns. There are two separate stem networks for each of the inputs. These sister stem networks have shared weights and have the same parameter updates throughout the training. The network therefore learns a similarity function, which takes the two inputs and decides how similar they are. The sharing of weights enables the network to project two extremely similar objects in nearby locations in vector space. And different objects further apart in the vector space.

Since we are sampling user and movie representations from same vector space (Movie embeddings), therefore the Siamese neural network should be able to compute their similarities. We pass the user embeddings in one stem and item embeddings in the other stem of a Deep Siamese Neural Network architecture and obtain two vectors for each user and movie in the same vector space. Siamese network utilised in this paper has the following specifications: $Siamese$ = Dense(800 $\rightarrow$ 512 $\rightarrow$ 256 $\rightarrow$ 100). Outputs are calculated as follows, $S_{U_n} = Siamese(u_n)$, and $S_{I_m} = Siamese(i_m)$. The architecture of the proposed siamese architecture based recommendation system is presented in Figure 5.

D. *Rating Predictor*

In order to perform regression, we concatenate the outputs from the siamese network and pass them through dense layers with the last layer being the regression output.

$$y_{nm}^{pred} = RatingPredictor(Concatenate(S_{U_n}, S_{I_m}))$$

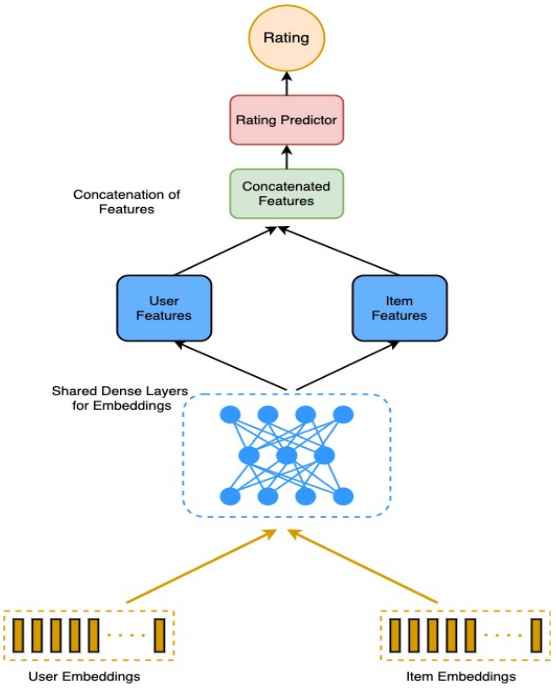

Fig. 5. Architecture of the **Siamese Network** module stacked with the **Rating Predictor** module.

where rating predictor is fully-connected regression layer with the specifications, $RatingPredictor$ = Dense(200 $\rightarrow$ 164 $\rightarrow$ 50 $\rightarrow$ 1)

VI. Experiments and Results

In this section, we describe the training setup of the proposed framework and report its performance under different setups. Analyses for the same are provided along with the probable reasoning. We have experimented with different combinations among the modalities and compared amongst themselves to see the impact each modality can create on the overall result. Also, experiments are done to examine the impact of data sparsity on performance, by masking at various degrees, the rating data.

A. *Training Setup*

The complete framework, as represented in the Figure 5, considers inputs as various modality embeddings to output regression value for rating prediction. This framework is trained using a combination of two losses. First is cosine embedding loss between the outputs from the two stems of Siamese Network. This loss enables the Siamese architecture to learn weights that transpose inputs to a vector space where dissimilar inputs are pushed further away and similar inputs are close by. Second is mean square error loss between predicted labels and actual ratings. These two losses are added to compute the final loss that shall be used to train the end to end pipeline.

$$\begin{aligned} Loss = CosineEmbeddingLoss(S_U, S_i) + \\ MeanSquareErrorLoss(y^{pred}, y^{true}) \end{aligned} \tag{2}$$

where cosine embedding loss is given by,

$$Loss_{cse} = \sum L_{nm}$$
$$L_{nm} = \begin{cases} 1 - \cos(S_{U_n}, S_{i_m}), & \text{if } I_{nm} = 1 \\ \max(0, \cos(S_{U_n}, S_{i_m})), & \text{if } I_{nm} = 0 \end{cases} \tag{3}$$

and mean square error loss is given by, $Loss_{mse} = \sum(y_{nm}^{true} - y_{nm}^{pred})^2$. Optimiser used for training purpose is **Adam**, with a learning rate of 0.001. Epochs are set to 200. All the dense layers are activated by a LeakyReLU function (with 0.01 leakage in negative side) apart from the first layers of Enc^{User} and first two layers of Enc^{Video} which utilize sigmoid function, for normalisation purpose.

B. Performance Metrics

For these experiments, we use the U1 splits, that are the prescribed train test splits provided by movie-lens dataset. Following this, we get 80,000 user-item-rating tuples as the training split and 20,000 tuples as the testing split.

Root mean square error [24] is considered as the primary performance metric. Furthermore, we consider binary class precision, recall as the secondary metrics. The secondary metrics are calculated as follows, the true labels are binarized using 4 as threshold. And the predicted ratings are binarised with 3.5 as threshold. These binarised true and predicted labels shall be used to calculate precision and recall. All the experiments mentioned in this section are conducted 5 times and best values are reported here. All the results are statistically significant.

C. Ablation Study on Modalities

Previously, we discussed that the item encodings are generated by concatenating the encodings from various modalities. In this section, we shall report on the performances with various combinations of the same. $i_m = Concatenate([i_m^x, i_m^y, ...])$. The results obtained by varying different modalities are reported in Table I.

TABLE I
MULTI-MODALITY ABLATION STUDY

Modality	***RMSE***	***Precision***	***Recall***
Text Only	1.054	**68.582**	77.036
Meta Only	1.025	69.419	85.132
Audio Only	1.153	58.182	73.698
Video Only	1.151	59.033	74.448
Meta + Text	1.032	68.129	80.715
Meta + Audio	1.034	70.102	71.811
Meta + Video	1.034	70.205	71.473
All	**1.028**	69.911	73.912

We can see that the best performing modality is a combination of all the existent representations (in terms of the primary metric - RMSE). However, we can see that the Meta + Text is the next best performing combination with no significant gap between the two. Hence, it could be concluded that the architecture and procedure followed in this paper work for the most directly related information like description of the movie and meta data information like genre and directors. While video and audio are intuitively important signals, the way they are currently being utilised in this work is sub-optimal and is subject to more investigation.

D. Rating Masking

An advantage of generating user-item representations from multi-modal data is that we don't need to rely on user ratings to generate the embeddings, i.e., we do not need users to rate hundreds of movies in order to generate accurate representation of users. Even if a user rates few movies, we should theoretically be able to make a user representation by taking the means of the movie representations which are liked by the user. However, it is important to establish that, by masking such ratings and reducing the sampling space of movies for user representation, there shouldn't be a significant drop in the performance. We have masked 50% of the user ratings in the first experiment and 80% of the user ratings in the second experiment. In each case, the metrics are computed and reported in Table II.

TABLE II
RESULTS WITH VARYING DEGREES OF MASKING

Table Modality	**50% Masking**			**80% Masking**		
	RMSE	***Prec.***	***Rec.***	***RMSE***	***Prec.***	***Rec.***
Meta + Text	1.048	70.653	75.149	1.049	68.221	78.321
Meta + Audio	1.054	67.348	78.942	1.057	68.119	77.817
Meta + Video	1.053	68.993	74.462	1.059	67.079	75.576
All	1.048	69.771	76.745	1.053	67.111	78.118

We can notice that masking doesn't significantly effect the metrics. In fact there is an improvement in performance in some modalities. This could be attributed to the fact that since the movie latent vectors might be highly variant, sampling a larger sample size might cause randomness in the user representation. However, if you sample from a smaller, but better representative segment, the representation might be bit more purer. This is not the case however with using rating vectors for representation. With less amount of rating data, the performance plummets. This is known as the infamous cold-start problem in recommendation system, and with the introduction of multimodal embedding for video representation we have made an attempt in overcoming this issue.

E. Comparison with State of the Art Models

Note that it is not fair to compare the results of the proposed approach with the existing systems available for Movie-Lens-100K data set. Existing approaches utilize rating information available with the data set to generate user and move embeddings which are further utilized for developing the recommendation models. Unlike these approaches, the proposed approach utilizes multi-modal features of the movies to generate a movie representation and movie representations are utilized to generate a user representation. Thus a direct comparison with the existing systems is not fair. But in a part of our study, we have performed the following experiment. We have replaced the user embedding and movie embeddings by rating vectors. Performance of the rating vectors independently using our proposed siamese based model is as follows: RMSE: 0.918, precision: 75.013, recall: 78.033. Note that these results are better than many of the state of the art techniques [25]–[27] published in recent years. This illustrates that the proposed siamese based architecture can provide state-of-the-art values when trained using supervised rating vectors. Note that the best RMSE value of the proposed multimodal approach (0.932) is higher than this. But the proposed model can help us in overcoming the cold-start problem of RS. This performance gap can be attributed to the lack of task-specific embedding vectors for different modalities.

VII. Conclusion and Future Works

In our current study, we have proposed the movie representation technique by using its different modalities like textual summary, video, audio and meta information and then the Siamese architecture has been incorporated for proposing regression over similarities among multi-modal user-item embeddings. The novelty of the task is building a multimodal movies/videos RS and the system does not require users' ratings for user/movie representation. Knowledge graph representation of meta-data information is the best performing modality with empirical improvement in performance, hence establishing the importance of graph-based representation of data. Apart from this, we have also enhanced the existing benchmark Movie-lens 100K for future research in the relevant domain. As per the observation, the result with all modality is not quite promising as the video and audio embeddings have put nominal impact in making the model accurate. Thus in future, efforts will be made in improving the audio and video embeddings to make them optimal/effective for the given task. Moreover, the feature vectors from different modalities are not extracted using an end to end setting making them task-specific. Future work will include building an end-to-end recommendation system which can provide the state-of-the-art performance values.

Acknowledgment

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

References

[1] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 1, pp. 5–53, 2004.

[2] R. Van Meteren and M. Van Someren, "Using content-based filtering for recommendation," in *Proceedings of the machine learning in the new information age: MLnet/ECML2000 workshop*, vol. 30, 2000, pp. 47–56.

[3] B. Lika, K. Kolomvatsos, and S. Hadjiefthymiades, "Facing the cold start problem in recommender systems," *Expert Systems with Applications*, vol. 41, no. 4, pp. 2065–2073, 2014.

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

[5] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold *et al.*, "Cnn architectures for large-scale audio classification," in *2017 ieee international conference on acoustics, speech and signal processing (icassp)*. IEEE, 2017, pp. 131–135.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[7] I. Melekhov, J. Kannala, and E. Rahtu, "Siamese network features for image matching," in *2016 23rd international conference on pattern recognition (ICPR)*. IEEE, 2016, pp. 378–383.

[8] J. Davidson, B. Liebald, J. Liu, P. Nandy, T. Van Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, B. Livingston *et al.*, "The youtube video recommendation system," in *Proceedings of the fourth ACM conference on Recommender systems*, 2010, pp. 293–296.

[9] A. Da'u, N. Salim, I. Rabiu, and A. Osman, "Recommendation system exploiting aspect-based opinion mining with deep learning method," *Information Sciences*, vol. 512, pp. 1279–1292, 2020.

[10] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," *Advances in neural information processing systems*, vol. 20, 2007.

[11] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion." *Journal of machine learning research*, vol. 11, no. 12, 2010.

[12] H. Wang, N. Wang, and D.-Y. Yeung, "Collaborative deep learning for recommender systems," in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2015, pp. 1235–1244.

[13] M. F. Aljunid and D. Manjaiah, "Movie recommender system based on collaborative filtering using apache spark," in *Data management, analytics and innovation*. Springer, 2019, pp. 283–295.

[14] S. Li, J. Kawale, and Y. Fu, "Deep collaborative filtering via marginalized denoising auto-encoder," in *Proceedings of the 24th ACM international on conference on information and knowledge management*, 2015, pp. 811–820.

[15] X. He, H. Zhang, M.-Y. Kan, and T.-S. Chua, "Fast matrix factorization for online recommendation with implicit feedback," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, 2016, pp. 549–558.

[16] H.-J. Xue, X. Dai, J. Zhang, S. Huang, and J. Chen, "Deep matrix factorization models for recommender systems." in *IJCAI*, vol. 17. Melbourne, Australia, 2017, pp. 3203–3209.

[17] F. Strub, R. Gaudel, and J. Mary, "Hybrid recommender system based on autoencoders," in *Proceedings of the 1st workshop on deep learning for recommender systems*, 2016, pp. 11–16.

[18] Z. Qin and M. Zhang, "Towards a personalized movie recommendation system: A deep learning approach," in *2021 2nd International Conference on Artificial Intelligence and Information Systems*, 2021, pp. 1–5.

[19] M. F. Aljunid and M. D. Huchaiah, "Multi-model deep learning approach for collaborative filtering recommendation system," *CAAI Transactions on Intelligence Technology*, vol. 5, no. 4, pp. 268–275, 2020.

[20] N. Nassar, A. Jafar, and Y. Rahhal, "A novel deep multi-criteria collaborative filtering model for recommendation system," *Knowledge-Based Systems*, vol. 187, p. 104811, 2020.

[21] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[22] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 28, no. 1, 2014.

[23] Y. Goldberg and O. Levy, "word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method," *arXiv preprint arXiv:1402.3722*, 2014.

[24] T. Chai and R. R. Draxler, "Root mean square error (rmse) or mean absolute error (mae)," *Geoscientific Model Development Discussions*, vol. 7, no. 1, pp. 1525–1534, 2014.

[25] J. Hartford, D. Graham, K. Leyton-Brown, and S. Ravanbakhsh, "Deep models of interactions across sets," in *International Conference on Machine Learning*. PMLR, 2018, pp. 1909–1918.

[26] F. Monti, M. Bronstein, and X. Bresson, "Geometric matrix completion with recurrent multi-graph neural networks," *Advances in neural information processing systems*, vol. 30, 2017.

[27] N. Rao, H.-F. Yu, P. K. Ravikumar, and I. S. Dhillon, "Collaborative filtering with graph information: Consistency and scalable methods," *Advances in neural information processing systems*, vol. 28, 2015.