International Journal of Progressive Research in Science and Engineering
Volume-2, Issue-2, February-2021
www.ijprse.com| ISSN(Online): 2582-7898

Film Proposals Utilizing Machine Learning With Root Mean Square Error

Umamageswari A[1], Phebe Persis P[2]

[1]Associate Professor, Department Of Computer Science and Engineering, DMI College Of Engineering, India.

[2]Assistant Professor, Department Of Computer Science and Engineering, DMI College Of Engineering, India.

Corresponding Author:r.umaramesh@gmail.com

Abstract: - In this movie, recommendation system is built based on the MovieLens10M dataset. We used recommendation method to predict user's movie rating and we can recommend movies to customers, which they potentially give high ratings according to prediction. The root-mean-square error (RMSE) is calculated to Carryout evaluation. A set of users at initial stage would have rated for example on the rate of 1to5 for some movies, which they have already seen. These ratings, which are given by these to users, is taken as in put to movie recommendation system. The movie recommendation system uses these ratings given by user to predict the ratings of other movies that each user would give. In some cases, user's ratings will not be available in such cases the movie recommendation system will not predict the ratings instead will predict the probability that user would choose to watch a movie other likelihood of the user.

Key Words: — Component, Knowledge Graph, Neural Network, Recommendation Algorithm.

I. Introduction

A. Machine Learning

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves. The process of learning begins with observations or data, such as examples, direct experience, or instruction, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. The primary aim is to allow the computers learn automatically without human intervention or assistance and adjust actions accordingly.

B. Machine Learning Methods

Machine learning algorithms are often categorized as supervised or unsupervised.

Supervised machine learning algorithms can apply what has been learned in the past to new data using labelled examples to predict future events. Starting from the analysis of a known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. The system is able to provide targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors in order to modify the model accordingly. In contrast, unsupervised machine learning algorithms are used when the information used to train is neither classified nor labelled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabelled data. The system doesn't figure out the right output, but it explores the data and can draw inferences from datasets to describe hidden structures from unlabelled data.

Semi-supervised machine learning algorithms fall somewhere in between supervised and unsupervised learning, since they use both labelled and unlabelled data for training – typically a small amount of labelled data and a large amount of unlabelled data. The systems that use this method are able to considerably improve learning accuracy. Usually, semi-supervised learning is chosen when the acquired labelled data requires skilled and relevant resources in order to train it / learn from it. Otherwise, acquiring unlabelled data generally doesn't require additional resources.

Reinforcement machine learning algorithms is a learning method that interacts with its environment by producing actions and discovers errors or rewards. Trial and error search and delayed reward are the most relevant characteristics of reinforcement learning. This method allows machines and software agents to automatically determine the ideal behaviour within a specific context in order to maximize its performance. Simple reward feedback is required for the agent to learn which action is best; this is known as the reinforcement signal.

C. Advantages of Machine Learning

Machine Learning undoubtedly helps people to work more creatively and efficiently. Basically, you too can delegate quite complex or monotonous work to the computer through Machine Learning - starting with scanning, saving and filing paper documents such as invoices up to organizing and editing images. In addition to these rather simple tasks, self-learning machines can also perform complex tasks. These include, for example, the recognition of error patterns. This is a major advantage, especially in areas such as the manufacturing industry: the industry relies on continuous and error-free production. While even experts often cannot be sure where and by which correlation a production error in a plant fleet arises, Machine Learning offers the possibility to identify the error early this saves down times and money. Self-learning programs are now also used in the medical field. In the future, after "consuming" huge amounts of data (medical publications, studies, etc.), apps will be able to warn a in case his doctor wants to prescribe a drug that he cannot tolerate. This "knowledge" also means that the app can propose alternative options, which for example also take into account the genetic requirements of the respective patient.

D. *Applications of Machine Learning*

1. *Virtual Personal Assistants*

Siri, Alexa, Google Now are some of the popular examples of virtual personal assistants. As the name suggests, they assist in finding information, when asked over voice. All you need to do is activate them and ask, “What is my schedule for today?”, “What are the flights from Germany to London”, or similar questions. For answering, your personal assistant looks out for the information, recalls your related queries, or send a command to other resources (like phone apps) to collect info.

You can even instruct assistants tasks like “Set an alarm for 6 AM next morning”, “Remind me to visit Visa Office day after tomorrow”. Machine learning is an important part of these personal assistants as they collect and refine the information based on your previous involvement with them. Later, this set of data is utilized to render results that are tailored to your preferences. Virtual Assistants are integrated to a variety of platforms. For example: Smart Speakers: Amazon Echo and Google Home Smartphones: Samsung Bixby on Samsung S8 Mobile Apps: Google All

2. *Predictions while Commuting*

Traffic Predictions: We all have been using GPS navigation services. While we do that, our current locations and velocities are being saved at a central server for managing traffic. This data is then used to build a map of current traffic. While this helps in preventing the traffic and does congestion analysis, the underlying problem is that there are a smaller number of cars that are equipped with GPS. Machine learning in such scenarios helps to estimate the regions where congestion can be found based on daily experiences. Online Transportation Networks: When booking a cab, the app estimates the price of the ride. When sharing these services, how do they minimize the detours? The answer is machine learning. Jeff Schneider, the engineering lead at Uber ATC reveals in an interview that they use ML to define price surge hours by predicting the rider demand. In the entire cycle of the services, ML is playing a major role.

3. *Videos Surveillance*

Imagine a single person monitoring multiple video cameras! Certainly, a difficult job to do and boring as well. This is why the idea of training computers to do this job makes sense. AI that makes it possible to detect crime before they happen powers the video surveillance system nowadays. They track unusual behaviour of people like standing motionless for a long time, stumbling, or napping on benches etc. The system can thus give an alert to human attendants, which can ultimately help to avoid mishaps. In addition, when such activities are reported and counted to be true, they help to improve the surveillance services. This happens with machine learning doing its job at the backend.

4. *Social Media Services*

From personalizing your news feed to better ads targeting, social media platforms are utilizing machine learning for their own and user benefits. Here are a few examples that you must be noticing, using, and loving in your social media accounts, without realizing that these wonderful features are nothing but the applications of ML. People You May Know: Machine-learning works on a simple concept: understanding with experiences. Facebook continuously notices the friends that you connect with, the profiles that you visit very often, your interests, workplace, or a group that you share with someone etc. Based on continuous learning, a list of Facebook users is suggested that you can become friends with. Face Recognition: You upload a picture of you with a friend and Facebook instantly recognizes that friend. Facebook checks the poses and projections in the picture, notice the unique features, and then match them with the people in your friend list. The entire process at the backend is complicated and takes care of the precision factor but seems to be a simple application of ML at the front end. Similar Pins: Machine learning is the core element of Computer Vision, which is a technique to extract useful information from images and videos. Pinterest uses computer vision to identify the objects (or pins) in the images and recommend similar pins accordingly.

5. *Email Spam and Malware Filtering*

There are a number of spam filtering approaches that email clients use. To ascertain that these spam filters are continuously updated, they are powered by machine learning. When rule-based spam filtering is done, it fails to track the latest tricks adopted by spammers. Multi-Layer Perceptron, C 4.5 Decision Tree Induction are some of the spam filtering techniques that are powered by ML. Over 325, 000 malwares are detected every day and each piece of code is 90–98% similar to its previous versions. The system security programs that are powered by machine learning understand the coding pattern. Therefore, they detect new malware with 2–10% variation easily and offer protection against them.

6. *Online Customer Support*

A number of websites nowadays offer the option to chat with customer support representative while they are navigating within the site. However, not every website has a live executive to answer your queries. In most of the cases, you talk to a chatbot. These bots tend to extract information from the website and present it to the customers. Meanwhile, the chatbots advances with time. They tend to understand the user queries better and serve them with better answers, which is possible due to its machine learning algorithms.

7. *Search Engine Result Refining*

Google and other search engines use machine learning to improve the search results for you. Every time you execute a search, the algorithms at the backend keep a watch at how you respond to the results. If you open the top results and stay on the web page for long, the search engine assumes that the results it displayed were in accordance to the query. Similarly, if you reach the second or third page of the search results but do not open any of the results, the search engine estimates that the results served did not match requirement. This way, the algorithms working at the backend improve the search results.

8. *Product Recommendations*

You shopped for a product online few days back and then you keep receiving emails for shopping suggestions. If not this, then you might have noticed that the shopping website or the app recommends you some items that somehow matches with your taste. Certainly, this refines the shopping experience but did you know that its machine learning doing the magic for you? On the basis of your behaviour with the website/app, past purchases, items liked or added to cart, brand preferences etc., the product recommendations are made.

9. *Online Fraud Detection*

Machine learning is proving its potential to make cyberspace a secure place and tracking monetary frauds online is one of its examples. For example: PayPal is using ML for protection against money laundering. The company uses a set of tools that helps them to compare millions of transactions taking place and distinguish between legitimate or illegitimate transactions taking place between the buyers and sellers.

II. RELATED WORK

Many RSs have been developed over the past decades. These systems use different approaches, such as CF, CBF, hybrid, and sentiment analysis to recommend the preferred items. These approaches are discussed as follows.

A. *Collaborative, Content-Based, and Hybrid Filtering*

Various RS approaches have been proposed in the literature for recommending items [48]. The primordial use of CF was introduced in [18], which proposed a search system based on document contents and responses collected from other users. Yang et al. [59] inferred implicit ratings from the number of pages the users read. The more pages read by the users, the more they are assumed to like the documents. This concept is helpful to overcome the cold start problem in CF. Optimizing the RS is an ill-posed problem. Researchers have proposed several optimization algorithms, such as gay wolf optimization [26], artificial bee colony [21], particle swarm Optimization [53], and genetic algorithms [6]. Katara et al.and Verma [26] developed a collaborative movie RS based on gay wolf optimizer and fuzzy c-mean clustering techniques. Both techniques are applied to the Movie lens data set and predicted a better RS. They improved the existing framework in [24] proposing an artificial bee colony and k-mean cluster (ABC-KM) framework for a collaborative movie RS to reduce the scalability and cold start complication. The combination of the hybrid cluster and optimization technique showed better accuracy in movie prediction compared with movie prediction by the existing frameworks. Dong et al. [11] proposed feature relearning with data augmentation for the Hulu Content-based Video Relevance Prediction Challenge. The result showed better improvement in TV shows and movie track in recall@100. Most approaches suffer from the sparsity problem in Social-aware Movie Recommendation systems (SMRs). Zhao et al. [63] developed a framework called SMR-multimodal network representation learning (MNRL) for movie recommendation to address this issue effectively. The result achieves better performance on a large-scale data set collected from the Chinese social-aware movie recommender site (Durban).

CBF [30], [39], [55], [57] is one of the most widely used and researched RS paradigms. This approach is based on the description of the item and a profile of the user's preferences. Nascimento et al. [35] discussed about discriminative power of the words for research articles recommendation. They deduced that title and abstract are multiple times stronger than the body text of the items and thus use the weightage scheme of the title, abstract, and body text to retrieve relevant articles. Contador et al. [9] made use of user and item profiles, described in terms of weighted lists of social tags to provide music recommendations [15], [23], [32]. Metermen and Sommerer [54] proposed a personalized RS to suggest articles for home improvement where the similarity between the user profile vector and a document was determined by using the combination of TF-IDF and the cosine similarity. Gossan et al. [19] proposed a new method for recommending news items based on TF-IDF and a domain ontology, i.e., CF-IDF. The performance of this method outperformed the TF-IDF approach on several measures, such as accuracy, recall, and the F1-measures when tested, evaluated, and implemented on the Athena framework. MA et al. [31] proposed a latent genre-aware micro video recommendation model for social media. Netflix data sets showed the effectiveness of the model. Recent research has demonstrated that the hybrid approach [5], [7],[40], [45], [50] is more effective than traditional approaches. The hybrid systems mitigate the drawback of individual technique due to the combination of multiple recommendation techniques. Melville et al. [34] developed a content-boosted CF system that used pure content-based features in a collaborative framework. This system further improved the prediction, first rate, and the sparsity problem. Zhang et al. [62] developed a framework based on user recommender interaction that takes input from the user, recommends N items to the user, and records user choice until none of the recommended items favour. Nogueira et al. [37] developed a mobile recommender system that combines a hybrid recommendation engine and a mobile 3-D GIS architecture. For testing the proposed framework, 27 users were selected with an age range of 24–48 years. To evaluate the performance of the RS, users were instructed to find restaurants, bars, and accommodation while walking and driving along a motorway. The user feedbacks demonstrated competent performance by the 3-D map-based interface that also overcame the limited screen size of most mobile devices. Hirokawa et al. [20] proposed a multimodal field-aware factorization machines (FFMs) algorithm to recommend the sentiment-aware personalized tweet. Users' interest is strongly influenced by sentiment factors in the tweet, and thus, this method models users' interest by deriving multimodal FFM that enables collaborative use of multiple factors in a tweet and improves performance. The experimental result of FFM evaluated through mean average precision, which showed a better result in comparison with other methods.

B. *Sentiment Analysis*

Sentiment analysis [8], [33], [41], [42] is a technique to computationally identifying and categorizing people's opinions expressed in the form of reviews or survey is positive, negative, or neutral. Sentiment analysis has been used TextBlob1 library to calculate the polarity and subjectivity of the review sentences. Past research has primarily focused on analysing the user-generated textual reviews and categorized the user reviews into positive or negative classes. In recent years, online reviews also include slang, emoticons, and some common words that help in finding the opinion of users more accurately. Hutto and Gilbert [22] proposed a valence-aware dictionary and sentiment reasoner (VADER) algorithm that is used to parse the user reviews and analyse them using a rule-based model to calculate the sentiment score of the tweets. This method is evaluated and validated in different domains, such as movie reviews, e-commerce product reviews, and news headlines. The result derived from the VADER method showed better performance than other sentiment analysis techniques. Rosa et al. [46] proposed a music recommendation framework for mobile devices where recommendations of songs for a user were based on the mood of the user's sentiment intensity. The studies were performed on 200 participants (100men and 10 women) to fill out their musical preferences choice in his or her profile. Later, the participant's profile was analysed and Movie lens using sentiment analysis from recommendation

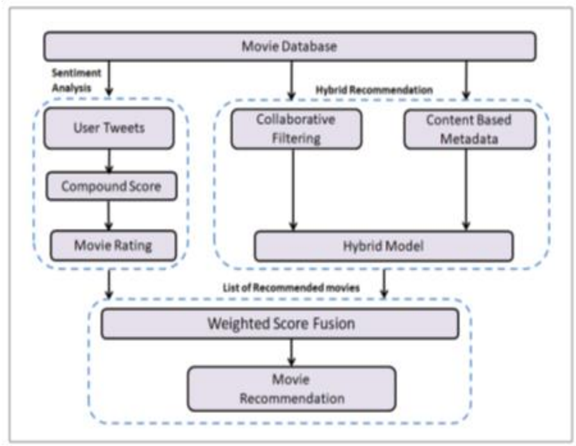

Fig. 1. Proposed movie recommendation framework

The results showed 91% user satisfaction rating. Li et al. [29] proposed the Bridge framework to solve the cold start problem in the CF system. Sentiment analysis was also used for microblogging posts in this framework. The polarity score of the post was assigned on a 1–5 rating scale. The result showed an enhanced RS by bridging the gap between user communication knowledge and social networking sites. Leung et al. [27] proposed a rating inference approach to transform textual reviews into ratings to enable easy integration of sentiment analysis and CF. our proposed model is a hybrid RS whose results are boosted using sentiment analysis score. Experimental evaluations, both quantitative and qualitative, demonstrate the validity and effectiveness of our method.

III. PROPOSED SYSTEM

The proposed sentiment-based RS is shown in Fig. 1. In this section, we describe various components of the proposed RS.A. Data Set Description. The proposed system needs two types of databases. One is a user-rated movie database, where ratings for relevant movies are present, and another is the user tweets from Twitter.

Public Databases: There are many popular public databases available, which have been widely used to recommend the movies and other entertainment media. To incorporate the sentiment analysis in the proposed framework, the tweets of movies were extracted from Twitter against the movies that were available in the database.

Experiments conducted using various public databases, such as the Movie lens 100K,2 Movie lens 20M,3 Internet Movie Database (IMDb,4) and Netflix database,5 that were not found suitable for our work due to the absence of microblogging data. After a thorough assessment of the abovementioned databases, the Movie Tweeting's database [12] was finally selected for the proposed system. Movie Tweeting's is widely considered as a modern version of the Movie Lens database. The purpose of this database is to provide an up-to-date movie rating so that it contains more realistic data for sentiment analysis. Table I displays the relevant details of the Movie tweeting's database.

Table.1. Details of the Movie tweeting's database

Metric	Values
Ratings	646410
Unique Users	51081
Unique Movie	29228
Start year	1894
End year	2017

2) Modified Movie Tweeting's Database: In the proposed work, the Movie tweeting's database is modified to implement the RS. The primary objective to modify the database was to use sentiment analysis of tweets by the users, in the prediction of the movie RS. The Movie Tweeting's database contains the movies with published years from 1894 to 2017. Due to the scarcity of tweets for old movies, we only considered the movies that were released in or after the year 2014 and extracted a subset of the database which complied with our objective.

$$\text{release\_year-movies} \geq 2014. \qquad (1)$$

Table.2. Example of a movie entry in the modified movie tweeting's database

Attribute	Value
Movie Id	0451279
Title	Wonder women
Runtime	141 min
Genre	Action, adventure
Director	Pithy Jenkins
Writer	Allan Heimberg
Actors	Gal Gadot, Chris pine
Rating	7.6
Production companies	Dc films, Tencent pictures
Popularity	524.772
Language	English
Budget	816303142

The subset of the database consisted of 292 863 ratings by51 081 users on 6209 different movies. The Movie Tweeting's database has three different components. The first component contains the mapping of users with their Twitter IDs. The second component contains the ratings of movies by users and their respective genres. The final component contains the information about the movies that were rated. In the proposed model, the socially filtered data, as well as the similarity of movies based on their attributes, has been used. The database had limited numbers of attributes for each movie, and thus, the Movie Database (TMDb) API was used to get more attributes of all the movies. TMDb6 is a premier source for extensive metadata for movies that have more than 30 languages. The movie attributes of the modified Movie tweeting's database are shown in Table II.

The modified database also contains some obscure movies from different countries and languages. The metadata for such movies was not available in TMDb, and therefore, those movies were discarded from the database. The final database had approximately 5000 movies.

2https://grouplens.org/datasets/movie lens/100k/

3https://grouplens.org/datasets/movie lens/20m/

4https://www.kaggle.com/orgesleka/imdbmovies

5https://www.kaggle.com/netflix-inc/netflix-prize-data/data

IJPRSE
Progressive Research
International Journal of Progressive Research in Science and Engineering
Volume-2, Issue-2, February-2021
www.ijprse.com| ISSN(Online): 2582-7898

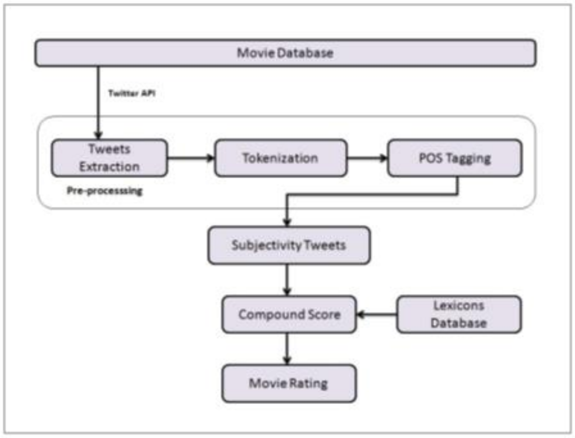

Fig. 2. Representative framework based on the VADER sentiment analysis system.

B. *Analysis of User Tweets*

As shown in Fig. 2, Twitter API7 was used to fetch the tweets for the movies that were present in the Movie tweeting's database. The extracted tweets consisted of tremendous amounts of noise, such as hashtags, emojis, repetitive words, and other irrelevant data that were removed using pre-processing techniques.

1) *Pre-processing of Tweets*: There are many short forms of words in the tweets, which converted into its original forms through gingerit8 library. To filter unusable data and uninformative parts in tweets such as stop words, punctuations, weblinks, and repetitive words, which did not add much value to sentiment analysis, as shown in Table III. After pre-processing, the text extracted from the tweets was used for sentiment analysis.

2) *Sentiment Analysis of User Tweets*: VADER is a lexicon and rule-based method that is used to find the opinion expressed by the users in the form of tweets. It maps the words to sentiment by looking up the intensity of a word in the lexicon. This method produces four sentiment components for each tweet. The first three components are positive, negative, and neutral. The last component is the normalization of all the above mentioned three components of the tweet. The sum of the first three components is always 1. Compound score lies between −1 to +1 where −1 represents extreme negative and +1 denotes extreme positive sentiment rating of the movie. For calculating the rating of the movie, the compound score is scaled in the range of 1–10 using (2), where x is a compound score.

$$\text{Rating} = [1 + (1 + x) \times 2] \times 2. \tag{2}$$

VADER performance is better than the other methods, as shown in Table V.

C. *Hybrid Recommendation*

In this section, we describe the combination of content-based similarity features with collaborative social filtering to generate a hybrid recommendation model. Let f = {f1, f2, . . ., fn} and q = {q1, q2…qn} are the content-based feature vectors and weight vectors, respectively. We construct the closeness C of two items i and j as:

C (i, j) =∑N n=1 *fn (Ani, An j),* for *i _= j* 0otherwise (3)

Where fn (Ani, Anj) corresponds to the similarity between feature values Ani and Anj corresponding to two items. In openness of the items is determined using the metadata or the relevant information related to the items. Fi j is constructed by combining the closeness vector C for all the items and multiplying it with the weight vectors q. Fi j is a feature matrix of dimension n × (M (M − 1)/2), where n and M are the number of feature attributes and number of items, respectively. The weight vectors q is evaluated using a social graph of items that indicate the user likeness of items. Let U = {u1, u2, . . ., un}, where ui is a user in the database A user item matrix is constructed for M items. An important property of the user-item matrix is that it has very high sparsity. Typical collaborative filtering [49] uses this user-item matrix to predict a user's rating of a particular item i by analysing the ratings of other users in the user's neighbourhood, normally, K neighbouring users. Neighbouring users are recognized by similarity measures, such as cosine similarity and Pearson correlation. After selecting K neighbouring users, the weighted aggregation of the ratings is as follows:

Rating (u, i) = 1/k∑ similarity (useru, uservk) · rating Ki (4)

Where u and vk are target user and K nearest neighbours, respectively. The procedure of CF is used to overcome the sparsity of the user-item matrix instead of directly using it to predict ratings. We employ the tweaked user-item matrix to construct a social graph using items as nodes. This graph represents the user's perception of similarity between the items. The determination of feature weights complies with the social graph. To determine the optimal feature weights q, we formulate a framework as described in the following equation:

$$S(i, j) = q \cdot F_{i\,j} \tag{5}$$

Which can be, expanded as

S (i, j) = q1. f1(A1i, A1 j) + q2 · f2(A2i, A2 j) +... +qn·fn (Ani, Anj). (6)

The procedure for determining the weights for the feature vectors used for calculating the similarity scores between two items have been constructed as a linear system, S (i, j). Here, S (i, j) are the number of users who are interested in both items I and j. Fi j denotes the feature vectors, which is constructed keeping in mind the similarity in metadata between two items.

International Journal of Progressive Research in Science and Engineering
Volume-2, Issue-2, February-2021
www.ijprse.com| ISSN(Online): 2582-7898

The similarity score using metadata of both items is calculated as described in (3). Therefore, the weight q here signifies the importance of a particular metadata when it is compared with the metadata of another movie. For example, the weight of the title of the movie will have more importance than the weight of the costume designer in determining the similarity between "The Dark Knight" and "The Dark Knight Rises." After having the weight matrix for the content-based metadata, we can calculate the similarity between an unknown movie A and a movie B, by using the weights present for B in the weight matrix computed from the user social graph.

For the entire database, S is a matrix of dimension1 × (M (M − 1)/2) and q is a matrix of dimension 1 × n, where n is the number of content-based features and dimensionality of F is n× (M (M − 1)/2). We calculate the weight vectors q for all the metadata feature attributes for all the items using the Moore–Penrose pseudoinverse as in the following equation:

$$q = S^{-1} \cdot F \qquad (7)$$

D. *Weighted Score Fusion*

To make the system robust, we use two data sources: one from the hybrid RS and another is from sentiment analysis. The hybrid RSs gives us the similarity between two movies based on their metadata (e.g., Actor, Director, Release Year, and Producer). The weights of these metadata for computing the similarity is computed under a linear system framework, as described in Section III-C The weights q signify the importance of particular metadata when a movie is compared with another (e.g., the genre of a movie has more importance than the runtime of the movie) movie. These precomputed weights from the collaborative social graph are used for computing the similarity with another new item for which we just have metadata information but no social user rating data. These weights q is normalized between [0, 1], and the concept of sentiment fusion is utilized in the proposed system. Through the retrieved user tweets, a sentiment rating is fabricated for all M movies. Let S€ {s1, s2,. ., sn},. Where si is the rating of movie, i calculated using (2). For calculating the sentiment similarity, a function G (i, j) for two movies i and j is defined based on their sentiment ratings si and s j as mentioned in (8) to determine how close are the movies in terms of the polarity of the user

$$G(i, j) = D - |s_i - s_j| \qquad (8)$$

Where D is a constant. The constant D, in (8) is taken as 10 because the ratings are on a scale of 1–10. Another function H (i, j) defined as

$$H(i, j) = q \cdot f_{ij} \qquad (9)$$

Where fi j is the feature similarity between movies i and j and q are the set of optimal weights as determined by (7). The final combined similarity CS (i, j) is described in (10). It is a weighted combination of the defined functions G and H

$$CS(i, j) = \omega 1 \cdot H(i, j) + \omega 2 \cdot G(i, j) \qquad (10)$$

$$\omega 1 + \omega 2 = 1, \omega 1, \omega 2\ \omega\ [0, 1] \qquad (11)$$

Where ω1 corresponds to the weight of the similarity score calculated from the hybrid model and ω2 corresponds to the weight of the sentiment similarity score. For a new movie item, we calculate this weighted similarity with all the movies present in the social graph for which we have the user rating data and then sort them by the computed similarity rating in descending order.

IV. Experimental Results And Analysis

In this section, quantitative, qualitative, and correlation coefficient, results are discussed.

A. *Correlation between Sentiment and IMDb Movie Ratings*

We conducted the statistical analysis between sentiment ratings X and movie rating Y to find the correlation coefficient. The correlation coefficient value varies from −1 to +1. Let D denotes a database of movies and N denote the number of total movies in the database. The statistical correlation coefficients are as follows: Spearman rank-order correlation coefficient (SROCC), Kendall rank correlation coefficient (KRCC), and Pearson linear correlation coefficient (PLCC). Table IV displays the values of different corelation coefficients utilized by us. In our experiments, we have found that sentiment and movie ratings are positively correlated. For PLCC, xi and yi are sentiment rating and IMDb movie rating, respectively, for the i the movie, whereas ¯x denotes the mean sentiment score and ¯y denotes the mean movie rating in the database. For SROCC, di is the difference between the sentiment rating and movie rating of the i th movie in the database. For KRCC, Nc and Nd represent the number of concordant and discordant pairs in the database, respectively.

B. *Evaluation Metric*

In many real-world applications, relevant recommendations are suggested by the system, instead of directly predicting rating values. This is known as Top-N recommendation [10], [47] and suggests specific items to users that are likable. The direct alternative methodologies are used for evaluation metric (e.g., precision). Precision is defined in terms of movies that are relevant (Lrel) and recommended (Lrec) by the model. In the proposed system, Precision @N is defined as follows:

$$\text{Precision@N} = L_{rel} \cap L_{rec} / L_{rec} \qquad (12)$$

International Journal of Progressive Research in Science and Engineering
Volume-2, Issue-2, February-2021
www.ijprse.com| ISSN(Online): 2582-7898

For the proposed model, the choice of weights in the fusion in (10) is determined by evaluating the Precision@5 and Precision@10 for a different combination of weights ω1 and ω2 conforming to (11).

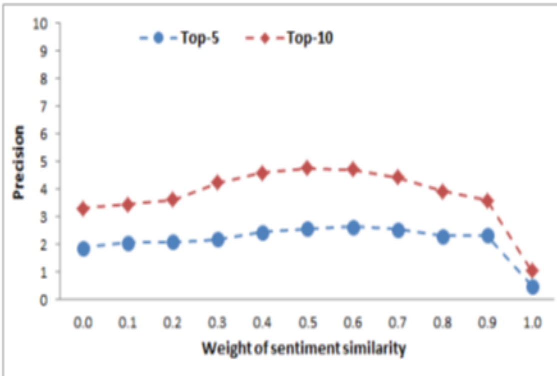

Fig. 3. Precision of Top-5 and Top-10 movies with varying sentiment similarity weights.

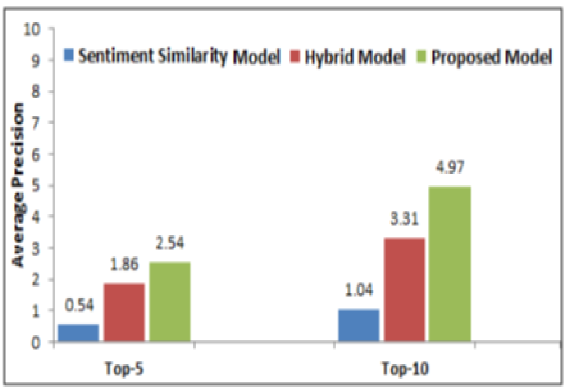

Fig.4.Comparison of the proposed model with baselines models.

C. *Weight Selection for Weighted Fusion*

For every movie, the Top-N recommendation list is evaluated using (10). The choice of the weights 1 and 2 in (10) is decided by experiments conducted on the metric mentioned in Section IV-B. The Precision@N is evaluated as in (12). The recommendations of all movies are collected from public databases, such as IMDb and TMDB. These recommended movies are considered as the ground truth. We compare the results of the Precision@5 and Precision@10for different values of ω1 and ω2. We choose the values of ω1 and ω2 for which the precision values are the From Fig. 3, an observation can be made that the maximum precision for weight values is between 0.5 and 0.6. Hence, ω1 and ω2 values are selected as 0.5 in the proposed system.

D. *Comparative Analysis*

In this section, we present a comparative analysis of our proposed system with the pure hybrid model (PH Model) and sentiment similarity models (SS Models). The PH Model is a combination of CBF and CF. The recommended movies are based on the similarity of attributes, such as genre, director, and cast. The similarities are evaluated using weights obtained by a social graph, as described in Section III-C. SS Model recommends movies based solely on the similarity of the movie tweets of the corresponding tuple of movies. We evaluate our proposed method using Precision@5 and Precision@10. Fig. 4 shows the quantitative comparative results of our proposed system with the baseline models. For Precision@5, the average precision values of the SS Model and PH Model are 0.54 and 1.86, respectively. Similarly, for Precision@10, the average precision values of SS Models and PH Model are 1.04 and 3.31, respectively. Our proposed model achieves a better precision value in both cases with 2.54 for Top-5 and 4.97 for Top-10 in comparison with the PH and SS Models. Thus, we can infer that our method will suggest at least two recommended movies out of five and five recommended movies out of ten. In addition, we have studied the FFM algorithm [20], [38] that uses personalized tweet recommendations for comprehending quick and accurate access to the desired information in the area of effective advertisements or election campaigns. This model is primarily effective when a fine-grained analysis is needed on the user's tweet along with its retweet to analyse multiple factors in a tweet, i.e., publisher, topic, and sentiment factors. Since this article is to propose a movie RS using approaches, such as hybrid, CF, and CBF against the modified Movie Tweeting database and sentiment analysis on the user's tweets, respectively, therefore, the FFM algorithm is not suitable in this article.

Comparison with Pertained Word Embedding and Attention Mechanism: Deep learning-based models are mostly used in the natural language processing and vision domain. The pertained word embedding (e.g., Glove algorithm) and attention mechanism models are used to compare our proposed system. Both models have used the IMDb database for training purposes. We have used the Glove algorithm to initialize the pertained vectors. Bidirectional LSTM, Adam optimizer, and dropout layer are the parameters used to train this model. After training this model, our database is used to calculate the polarity score that is eventually converted into a rating using (2). As shown in Table V, the movie's rating is the average rating of the movie's tweets. The pertained rating results are inferior to Vader rating due to ignorance of the tweet's context.

Table.5. Comparative Analysis Ratings

S.NO.	Movie Lists	Vader ratings	Naive Bayes ratings	Text lob ratings	PTWE ratings	Attention model ratings	IMDB ratings
1	Baby driver	7.61	7.37	7.25	6.35	6.0	7.6
2	Snowden	7.72	6.06	7.45	5.86	5.0	7.3
3	Arrival	8.45	7.7	7.68	7.87	8.0	7.9
4	Storks	7.54	6.39	7.24	6.39	7.24	6.8
5	Mother	6.63	6.15	6.32	6.09	6.0	6.6
6	Neerja	8.21	7.48	7.38	7.08	8.0	7.7
7	Alien covenant	6.77	6.19	6.03	3.45	7.0	6.4
8	Captain America	6.04	7.29	7.15	4.67	6.0	7.8
9	A dog purpose	7.54	6.37	7.38	6.67	6.0	7.0

IMDb	TMDb	Recommendation from the proposed system
Justice league	Guardian of the galaxy vol.2	Batman vs spiderman: Dawn of justice
Batman vs spiderman	Spiderman: Home coming	Suicide squad
Suicide squad	Logan	Thor – Ragnarök
Thor – Ragnarök	Thor – Ragnarök	Justice league
Deadpool	Pirates of the Caribbean	Doctor strange
Logan	Doctor strange	Guardian of the galaxy vol.2
Captain America	Baby driver	Kong: skull Island
Doctor strange	Kong: skull Island	The LEGO Harley Quinn
Guardian of the galaxy vol.2	Life	Batman and Harley Quinn

Qualitative Analysis of Wonder Woman Hollywood Movie: Model has been trained on 10 epochs with 32 batch size [4],[17], [44], [52]. Attention models are used bidirectional LSTM with attention layer [13], [58], [60], [61]. The result shows the inferior performance of this model than the performance of the VADER method. Deep learning requires a huge amount of relevant data to give an accurate result. In this article, the performance is inferior due to not having a large amount of data.

Effect of Path Length Paths with different lengths were filtered from all paths between user-items, i.e., L= {3,5,7}, and then which were send to the recurrent network for further processing. Figure 3 shows the results: as the path length increases, the accuracy rate gradually decreases. This fully shows that too long paths have more noise, and shorter paths have clearer semantics.

E. Qualitative Analysis

In this section, we show the qualitative results for some of the movies recommended by the proposed system. The results also include movies from both Hollywood as well as Bollywood, as shown in Tables VI and VII, respectively. It is interpreted from these tables that the recommendations from the proposed system have many intersecting movies, with the recommendations from both IMDb and TMDB.

V. CONCLUSION AND FUTURE WORKS

The recommendation system is of great significance for screening effective information and im-proving the efficiency of information acquisition. RSs are an important medium of information filtering systems in the modern age, where the enormous amount of data is readily available. In this article, we have proposed a movie RS that uses sentiment analysis data from Twitter, the proposed system Used weighted score fusion to improve the recommendations. Based on our experiments, the average precision in Top-5 along with movie metadata and a social graph to recommend movies. Sentiment analysis provides information about how the audience is respond to a particular movie and how this information is observed to be useful. And Top-10 for sentiment similarity, hybrid, and proposed model are 0.54 and 1.04, 1.86 and 3.31, and 2.54 and 4.97, respectively. We found that the proposed model recommends more precisely than the other models. In the future, we plan to consider more information about the emotional tone of the user from different social media platforms and non-English languages to further improve the RS. However, this model still needs to be improved, for example, the accuracy rate still has a huge room for improvement. This also proposes new ideas for future study.

REFERENCES

[1]. F. Abel, Q. Gao, G.-J. Hoeben, and K. Tao, “Analysing user modelling on Twitter for personalized news recommendations,” in Proc. 19th Int. Conf. Modelling, Adaption, Pers. (UMAP). Berlin, Germany: Springer-Verlag, 2011, pp. 1–12.

[2]. F. Abel, Q. Gao, G.-J. Hoeben, and K. Tao, “Twitter-based user modelling for news recommendations,” in Proc. Int. Joint Conf. Atif. Intel., vol. 13, 2013, pp. 2962–2966.

[3]. G. Dominicus and A. Zhilin, “Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions,” IEEE Trans. Know. Data Eng., vol. 17, no. 6, pp. 734–749, Jun. 2005.

[4]. O. Raque, I. Corcuera-Platas, J. F. Sánchez-Rada, and C. A. Iglesias, “Enhancing deep learning sentiment analysis with ensemble techniques in social applications,” Expert Syst. Appl., vol. 77, pp. 236–246, Jul. 2017.

[5]. E. Alsatian, M. Radames, and M. Jalili, “Hybrid recommender systems based on content feature relationship,” IEEE Trans. Ind. Informant., early access, Nov. 21, 2016, doe: 10.1109/TII.2016.2631138.

[6]. J. Bobadilla, F. Ortega, A. Hernando, and J. Alcalay, “Improving collaborative filtering recommender system results and performance using genetic algorithms,” Know. -Based Syst., vol. 24, no. 8, pp. 1310–1316, Dec. 2011.

[7]. R. Burke, “Hybrid recommender systems: Survey and experiments,” User Model. User-Adapted Interact., vol. 12, no. 4, pp. 331–370, 2002.

[8]. E. Cambria, “Affective computing and sentiment analysis,” IEEE Intel. Syst., vol. 31, no. 2, pp. 102–107, Mar./Apr. 2016.

[9]. I. Contador, A. Beloin, and D. Valet, “Content-based recommendation in social tagging systems,” in Proc. 4th ACM Conf. Rec. Syst. (Recasts), 2010, pp. 237–240.

[10]. P. Ceremonies, Y. Koran, and R. Turin, “Performance of recommender algorithms on top-N recommendation tasks,” in Proc. 4th ACM Conf. Rec. Syst. (Recasts), 2010, pp. 39–46.

[11]. J. Dong, X. Li, C. Xu, G. Yang, and X. Wang, “Feature re-learning with data augmentation for content-based video recommendation,” in Proc. ACM Multimedia Conf. (MM), 2018, pp. 2058–2062.

[12]. S. Dooms, T. D. Bessemer, and L. Martens, “Movie Tweetings: A movie rating dataset collected from

Twitter," in Proc. Workshop Crowdsourcing Hum. Compute. Rec. Syst., 2013, pp. 1–2.

[13]. C. Du and L. Huang, "Text classification research with attention-based recurrent neural networks," Int. J. Compute. Common. Control, vol. 13, no. 1, pp. 50–61, 2018.

[14]. X. Du, H. Yin, L. Chen, Y. Wang, Y. Yang, and X. Zhou, "Personalized video recommendation using rich contents from videos," IEEE Trans. Know. Data Eng., vol. 32, no. 3, pp. 492–505, Mar. 2020.

[15]. A. Flexor and J. Stevens, "Mutual proximity graphs for improved reachability in music recommendation," J. New Music Res., vol. 47, no. 1, pp. 17–28, Jan. 2018.

[16]. X. Fu and Y. Shen, "Study of collective user behaviour in Twitter: A fuzzy approach," Neural Compute. Appl., vol. 25, nos. 7–8, pp. 1603–1614, Dec. 2014.

[17]. M. Gratosoglio, M. G. Vocalist, K. Diamantaires, A. Vialing. Sargonids, and K. C. Cassava's, "Sentiment analysis leveraging emotions and word embeddings," Expert Syst. Appl., vol. 69, pp. 214–224, Mar. 2017.

[18]. D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," Common. ACM, vol. 35, no. 12, pp. 61–70, Dec. 1992.

[19]. F. Gossan, W. Intima, F. Frasnian, F. Hogeboom, and U. Kayak, "News personalization using the CF-IDF semantic recommender," in Proc. Int. Conf. Web Intel., Mining Semantics (WIMS), 2011, pp. 10:1–10:12.

[20]. R. Hirokawa, D. Takehisa, T. Ogawa, and M. Hassayampa, "Sentiment aware personalized tweet recommendation through multimodal FFM," Multimedia Tools Appl., vol. 77, no. 14, pp. 18741–18759, Jul. 2018.

[21]. C.-C. Hsu, H.-C. Chen, K.-K. Huang, and Y.-M. Huang, "A personalized auxiliary material recommendation system based on learning style on Facebook applying an artificial bee colony algorithm," Compute. Math. Appl., vol. 64, no. 5, pp. 1506–1513, Sep. 2012.

[22]. C. J. Hutto and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in Proc. 8th Int. Conf. Weblogs Social Media, 2014, pp. 216–225.

[23]. L. Jinn, D. Yuan, and H. Zhang, "Music recommendation based on embedding model with user preference and context," in Proc. IEEE 2nd Int. Conf. Big Data Anal. (ICBDA), Mar. 2017, pp. 688–692.

[24]. R. Katara, "Movie recommender system with Metaheuristic artificial bee," Neural Compute. Appl., vol. 30, no. 6, pp. 1983–1990, Sep. 2018.

[25]. R. Katara and O. P. Verma, "An effective collaborative movie recommender system with cuckoo search," Egyptian Informant. J., vol. 18, no. 2, pp. 105–112, Jul. 2017.

[26]. R. Katara and O. P. Verma, "Recommender system with grey wolf optimizer and FCM," Neural Compute. Appl., vol. 30, no. 5, pp. 1679–1687, Sep. 2018.

[27]. C. W. Leung, S. C. Chan, and F.-L. Chung, "Integrating collaborative filtering and sentiment analysis: A rating inference approach," in Proc. ECAI Workshop Rec. Syst., 2006, pp. 62–66.

[28]. F. Li, G. Xu, and L. Cao, "Two-level matrix factorization for recommender systems," Neural Compute. Appl., vol. 27, no. 8, pp. 2267–2278, Nov. 2016.

[29]. H. Li, J. Cui, B. Shen, and J. Ma, "An intelligent movie recommendation system through group-level sentiment analysis in microblogs," Neurocomputing, vol. 210, pp. 164–173, Oct. 2016.

[30]. P. Lops, M. de Gemma's, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in Recommender Systems Handbook, F. Ricci, L. Roach, B. Shapira, and P. Kantor, Eds. Boston, MA, USA: Springer, 2011, pp. 73–105.

[31]. J. Ma, G. Li, M. Zhong, X. Zhao, L. Zhu, and X. Li, "LGA: Latent genre aware micro-video recommendation on social media," Multimedia Tools Appl., vol. 77, no. 3, pp. 2991–3008, Feb. 2018.

[32]. Y. Matsumoto, R. Hirokawa, T. Ogawa, and M. Hassayampa, "Music video recommendation based on link prediction considering local and global structures of a network," IEEE Access, vol. 7, pp. 104155–104167, 2019.

[33]. W. Medhat, A. Hassan, and H. Kora shy, "Sentiment analysis algorithms and applications: A survey," Ain Shams Eng. J., vol. 5, no. 4, pp. 1093–1113, Dec. 2014.

[34]. P. Melville, R. J. Mooney, and R. Nagarajan, "Content-boosted collaborative filtering for improved recommendations," in Proc. AAAI, vol. 23, n 2002, pp. 187–192.

[35]. C. Nascimento, A. H. F. Laender, A. S. da Silva, and M. A. Gonçalves, "A source independent framework for research paper recommendation," in Proc. 11th Anna. Int. ACM/IEEE Joint Conf. Digit. Libraries (JCDL), 2011, pp. 297–306.