IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Movie Recommendation System Using Varian of Recurrent Neural Network (Bidirectional GRU & ROA)

[1] Ms. Gadakh Archana. K., [2] DR. Pawar U. B.

[1] PG Student, [2] Asso. Professor,

[1]Department of Computer Engineering,

[1]SNDCOERC, Yeola, India

Abstract: The Recurrent Neural Network (RNN) deep learning algorithm, which mainly learns and predicts sequential and time series data, is mainly used in language modeling, stock price prediction, and chatbot. In this paper, we propose a movie prediction and recommendation method based on considering the movie consumption patterns of users. We measure similarity between users based on movie rating data, classify users with similar movie preferences, and identify the consumption patterns of each similar user group to improve prediction accuracy by taking into account the change in preferences over time. To show the effectiveness of the proposed method, we apply a collaborative filtering algorithm, a simple RNN, and our modified RNN and compare their prediction accuracies.

Expressing reviews in the form of feelings or ratings on an item used or a movie watched is part of human habit. These reviews are easily available on various social networks. Based on the user's interest pattern, it is important to recommend items to the user. To this day, many recommendation systems are designed using several machine learning algorithms. The ever-increasing convergence speed, prediction accuracy, and appropriate optimization are obstacles for recommender systems that must be solved by hybrid algorithms. In this paper, we propose a system that uses Bidirectional Gated Recurrent Unit (BiGRU), the latest variant of Recursive Neural Network (RNN) in collaboration with Remora Optimization Algorithm (ROA).

***Index Terms* - movie recommendation, user similarity, consumption pattern, sequence data, Recurrent Neural Network, BIGRU, ROA**

I. INTRODUCTION

People are heavily connected with social media to share their emotions and reviews on various websites. These emotions are in the form of feelings or evaluations of a product or service. As a result, a huge amount of data is generated and studied to predict, recommend any product or service to the user for their interest. A movie rating database with users and various movie parameters is available on several popular websites such as Kaggle. Decisions made with the support of several stronger historical impressions to solve a problem are always better than decisions made using a single impression by any user. Instead of collecting all reviews or ratings, only the users for whom the rating is more important are collected.

Many researchers have tried to improve the recommender system using various machine learning techniques such as GA, Neural Network (NN), Support Vector Machine (SVM) and many others. In this work, we proposed a movie recommendation system based on BiGRU algorithm optimized through ROA. The BiGRU algorithm is applied to the 5000 IMDB dataset. User ID, Movie ID, and Rating parameters are only taken into account during pre-processing. The similarity between users is obtained to rate the similarity of the same movies, as well as the weight difference between the same movies. The weights to find the similarity between users are optimized by ROA and finally the top 10 movies are recommended to the user based on his interest pattern. The results are compared with the output from GA, MMDL and FFNN. It was observed that BIGRU shows better results obtained for all the test parameters we used for comparison. Recently, many studies have been conducted to predict and recommend products to buy in the near future through customized analysis of individual users, and applications such as Netflix recommending movies and recommending Amazon products are on the rise. In order to predict future consumption, user-based or item-based filtering algorithms are typically used. However, these methods are typically based on user-provided rating data, meaning they cannot be predicted without rating data. Therefore, a product without rating data cannot be included in the recommendation list, causing a shortage problem. Also because these conventional methods do not take into account the changes of time, it does not reflect the changes in the consumption pattern of users.

In this paper, we propose a movie prediction and recommendation method based on considering the movie consumption patterns of users. We measure similarity between users based on movie rating data and classify users with similar movie preferences. We then use RNN to learn movie consumption patterns of similar user groups and later predict or recommend movies. To show effectiveness

In the proposed method, we apply a collaborative filtering algorithm, a simple RNN, and our modified RNN to the latest Movie Lens dataset and compare their prediction accuracies.

II. LITERATURE REVIEW

Currently, the increment in e-commerce has a great improvement in personal checks. Customers have changed the way they shop. Purchase choices must be based on the facts about the properties listed on the website. Online carriers are trying to overcome this tipping point by allowing the user to share merchandise checks online (Park 2007). Within the net age, the most serious problem for an individual who wants to buy something on the internet is not always just the way to get enough statistics or desire, and from time to time they try to make an accurate choice with massive data. Nowadays, individuals are constantly searching the internet to discover great viable items and deals that they need. Intentionally or unknowingly, they depend on the recommendation system (RS) to conquer the overload of facts. RS has been proven to be a huge answer to record overload issues, offering customers more and more proactive and tailored statistical services.

RS are software program gadgets that suggest items that are likely to be user intrigue. Gadget recommendation is characterized by a help system that is used to help customers find data offers or gadgets such as internet destinations, TV apps, songs, movies, books and virtual products.

At the suggestion of alternative customers. It provides customized recommendation services to many users. A recommendation engine is a statistical filtering engine, otherwise called a design engine, that is used to prescribe lighting objects. They can be seen all over the region: there are RS for songs, motion pictures, travel, books, articles, news and basic stuff and have actually become a remarkable segment on websites like Netflix, Amazon. , YouTube, Google and others (Ricci 2015). RS uses different strategies, such as collaborative filter (CF), content-based ones, and hybrid RS (Porcel et al. 2018). CF techniques are all of the larger ones commonly used; they do not need any previous data about customers or matters; as an alternative, they create pointers with cooperation between them. Despite the fact that they are effective and simple, they revel in the ill effects of a whole host of problems, such as accuracy of expectations, cold starts, and the absence of the ability to accept difficult patron-problem collaboration (Fu et al. 2018). The referral system includes the following three elements.

Items: Recommendation outputs are called items. Things that are recommended are referred to as items. Items may be displayed as high value items for complexity or service. In the vast majority of recommendation systems, rating things can be certain if it is useful to the user. Things that are disliked are represented by negative attributes.

Through human-computer interactions, giving up customers of the advisory machine receive instructions. RS users are people; their interest differs from one person to another, and the pointers given to each person can be additionally exchanged. Since users have numerous dreams and attributes, it is important to tailor tips that require a wide range of facts. Statistics can be stored, checked and handled in various ways, and decisions can be made about the method and device with which we will use the facts. Facts can be prepared in exceptional ways and also determine the model with which information is based on RS.

Transaction: Exchange is a legitimate unit of work. In recommender systems, the exchange is a recorded connection between the user and the recommender system. These exchanges are recorded in log documents to store all data on human-RS interactions. These log records are listed as contributions to pattern detection algorithms. These patterns are used by the recommendation system to predict things for users.

1) (Cheng et al., 2020) proposed a movie recommendation model based on recurrent neural network (RNN) and KG-RKAN (knowledge graph-recurrent knowledge attention network), which uses auxiliary information in KG to search for potential user interests for personalized recommendations. They solve the problem of individual user interests by designing an attention module in RKAN, using different weights to converge user interests. For testing purposes, they mapped data collected from the real Movie lens dataset and IMDB to a new dataset for testing. Their model significantly improved the accuracy of recommendations.

2) (Pongpaichet et al., 2020) proposed a rating prediction algorithm using singular value decomposition (SVD). They extended a movie recommendation algorithm based on singular value decomposition (SVD) using parallel stochastic descent (PSGD) and improved its speed. They compared their proposed algorithm with a state-of-the-art rating prediction algorithm based on a traditional user-user collaborative filtering algorithm on the Movie Lens dataset, and their proposed algorithm outperforms the baseline in terms of accuracy in both rating prediction and movie. recommending tasks.

3) (W. Wang et al., 2020) proposed a combined LSTM and CNN recommendation model. Their model combines CNN to fully mine local information from movie data and uses LSTM to capture the context of user ratings. They used the Movie Lens 1M dataset. Compared with the traditional recommendation model and other deep learning-based recommendation models, the combined LSTM and CNN recommendation model proposed in this paper has a reduction in MSE loss of 4.4%~18.7% and a reduction in MAE loss of 3.0%~52, 2%. .

4) The TimeFly algorithm is a new behavior-inspired recommendation algorithm that works on the concept of changing user behavior with respect to time. Their proposed model considers two recommendation problems (fluctuating user interest over time and long computation time when datasets change from sparse to abundant) and shows a real-world implementation of the approach in the field of recommendation engines. On the Movie Lens 1M dataset, they compared the results of the TimeFly algorithm with the results of other known algorithms. They found that using TimeFly resulted in more accurate predictions in less time. (Sinha et al., 2020)

5) (Shen et al., 2020) used a collaborative filtering algorithm to implement a movie recommendation system. They used the Movie Lens dataset for their experiments. Their system achieved high efficiency and reliability in large data sets.

6) A system that uses Hadoop technology can meet the needs of big data and cloud computing environments (Shen et al., 2020). KG provides an efficient way to design recommender systems in a big data environment. As an emerging type of auxiliary

III. SYSTEM ARCHITECTURE

The common RS architecture includes the following three basic components as shown in Fig. 1.1

a) Candidate Generation This is the number one section in RS that takes opportunities from out-of-motion consumers as information and recovers a small subset of (numerous) recordings from a huge corpus. There are usually two main age categories of competitors around:

• Content-based filtering Full content-based isolation involves designing things that depend on the properties of the things themselves. Gadget prescribes such things as what the consumer used to value.

• Collaborative CF filtering relies on consumer-object interplay and is based on the concept that comparison customers like comparison things, for example, customers who bought one element could have bought any other factor.

b) Scoring This introduces a second level of RS, where any other model places and rates the applicant as a whole on a scale of 10. As an example, as a result of YouTube, the positioning device achieves this commitment by assigning a score to each video as indicated through the correct job objective using a rich combination of video and shopper main points. Maximum increased scoring recordings are presented to the buyer and placed according to their score.

c) Reordering Within 1/3 degree, the machine considers more necessities to ensure decent range, freshness and propriety. For example, the structures exclude a substance that was not previously explicitly preferred by the consumer and take into account each new item on the web page.

There are sequences of steps that are followed when attempting the referral framework process. The individual steps are as follows:

• Data Collection
• Data storage
• Data analysis
• Data filtering (Rehman 2019)

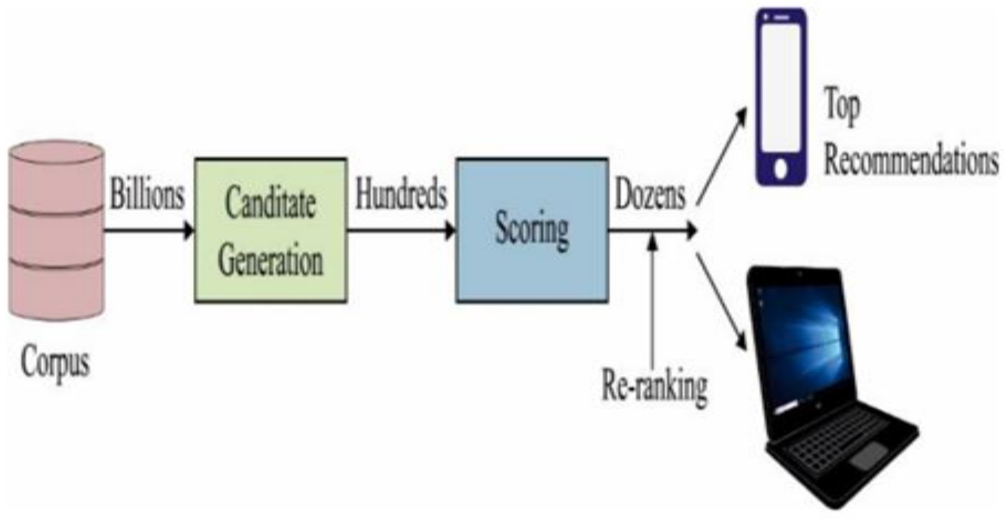

Figure. 1.1. Common Architecture for Recommender Systems Processes of recommender systems

i. Data collection: In this step, the artificial intelligence (AI) device uses information that can be obtained and found in many systems, regardless of whether it is implicit or specific. Information that is known to be specific can be collected by engaging in surveys and sharing comments from users of approximately exceptional gadgets. Be that as it may, information of an implicit nature is recognized with the pursuit protocol and records of the facts that are obtained in various bureaucracies. This information can be effectively obtained by looking at the consumer's search log or historical background, which records or collects records approximately the consumer's pastimes and items.

ii.Data storage: The previous information collected is deferred or stored in the systems which helps in providing suggestions later. Thus, saving the given data helps in making recommendations that can be made about the system and can support recommendation help about users.

iii.Data Analysis: After completing the above steps, one can additionally find that the records are moving in the direction of the analysis phase, in which the records are thoroughly analyzed.

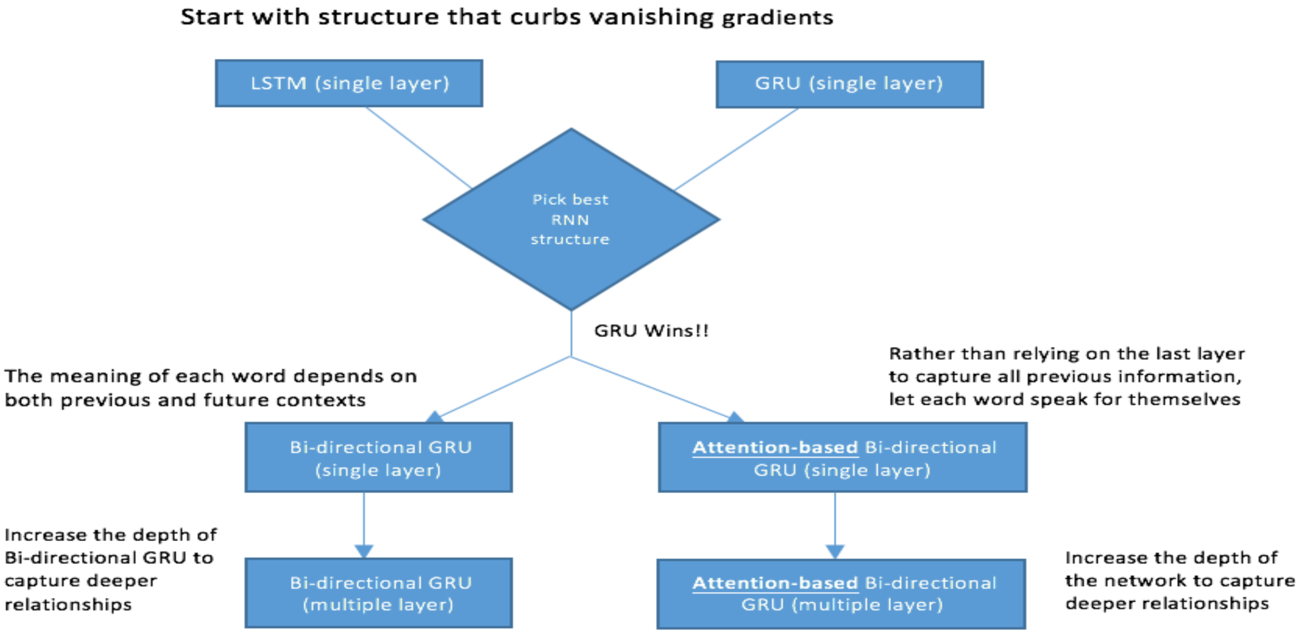

Figure 1.2: Model Selection Flow

First, we compare the performance between GRU and LSTM on this specific prediction task, the result indicates that the GRU framework performs slightly better than LSTM. Using GRU as an RNN cell, we implement a one-way, double-, triple-, and quadruple-stacked two-way model; the same implementation procedure is also used to implement four layered two-way attention-based structures.

The BiRNN consists of a forward and a backward RNN (GRU cell) structure. In a feedforward RNN, the input sequence is ordered from the first word to the last word, and the model computes a sequence of feedforward hidden states. A backward RNN takes the input sequence in reverse order, resulting in a sequence of backward hidden states. To calculate the final prediction, we average the output from the RNN in both directions and then use a linear transformation to generate the input to the SoftMax prediction unit.

IV. METHODOLOGY & ALGORITHM

A) Bidirectional Gate Recurrent Unit (BiGRU)

BiGRU is a newer version of RNN than LSTM, which is gaining a lot of popularity these days. RNN has a very deep computation graph because it repeats the same operation at each time point. The neural network long-term and short-term memory technique is designed to solve RNN problems, but its structure becomes more complicated and it is difficult to converge at a higher speed. The speed of BiGRU is much higher than that of LSTM.

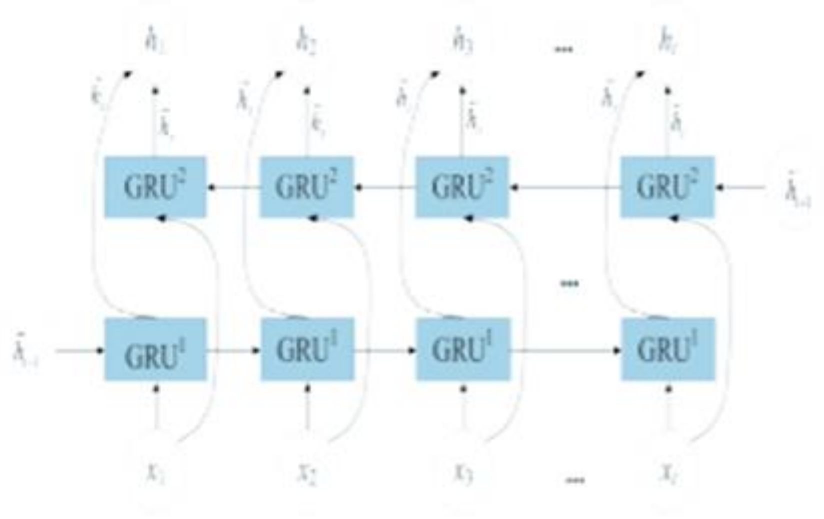

Figure 1.3 BIGRU Working Architecture

B) Remora Optimization Algorithm (ROA)

ROA [52] is a well-known bionics-based meta-heuristic algorithm inspired by the parasitic behavior of remora during foraging in the ocean. Unlike other fishes, remora usually attach to other hosts (humpback whales or sailfishes) to complete long and short-distance movement in the ocean. Like other MAs, ROA also has three different phases: initialization, exploration, and exploitation.

To minimize error, we are using optimization. Remora is the latest one and its convergence is better than rest of another optimization algorithm. So we preferred Remora. There are following steps in remora optimization algorithm.

The pseudo-code of basic remora optimization algorithm

1. Initialization
2. Initialize the remora population size (N) And maximum number of iterations (T)
3. Initialize the positions of all search Agents Xi (i=1, 2, 3… N)
4. Set the remora factor C
5. Main loop {
6. While (t ≤ T)
7. Calculate the fitness of each remora
8. Find the best position and best Fitness, Xb
9. Calculate the a, b, A, B
10. for the ith remora
11. If H (i) = 0
12. Generate position VI by Eq (3)
13. Else if H (i) = 1
14. Generate position Vi by Eq (1)
15. End if
16..Generate candidate position Vi' by Eq (2)
17. if f(Vi') < f(Vi)
18. Xi = Vi'
19. H(i) = round(rand)
20. else
21. Update position Xi by Eq (7)
22. End if
23.End for
24. t = t + 1
25. End While}
26. Return bestFitness, Xb

ROA [52] is a well-known bionics-based meta-heuristic algorithm inspired by the parasitic behavior of remora during foraging in the ocean. Unlike other fish, remora usually attach to other hosts (humpbacks or sailfish) to complete long- and short-distance movements in the ocean. Like other MAs, ROA also has three distinct phases: initialization, exploration, and exploitation.

• Initialization

Like other various metaheuristic algorithms, ROA initializes the search agents using a random access in the search space calculated by:

$X_i = lb + rand \times (ub - lb);\ i \in \{1, 2, \ldots, N\}$ (1)

where rand denotes a random variable between [0, 1]. ub and lb denote the upper and lower bounds of the search space. i represents the number of Remora and N indicates the population size.

- Generate an initial population

The population here is a parameter for the search agent. In our system the remora search agent is that we need to initialize the remora or search agent number.

- Define the network weight:

Here the neural network weight values will be defined. These are randomly generated weight values close to 0.

- Change search agents

In all optimization algorithms, we first define the solution space. We also define the search area and the number of search agents. If the search agents exceed the threshold value in the search space, we adjust the value of the search agents. Only search agents below the threshold will be allowed in the search space.

- Minimizing errors

Here we use error function or loss function to minimize the error.

- Store Fitness and current search agent position

There are different network weights in the search space. The optimal weight of the network is saved.

- Find near-optimal weight

The first weight value is known, and the next weight value, which is the optimal weight, can be found by evaluating the fitness value of each weight value we initially chose. For each iteration, we evaluate the fitness function.

- Evaluate fitness (error minimization)

For each network weight, we evaluate the fitness value. We compare the new weight value with the previous weight value.

- Stop criteria:

Initially, we define the number of iterations that will be our stopping criteria.

V. INCORPORATED PACKAGES

- Flask
- gunicorn
- Jinja2
- MarkupSafe
- Werkzeug
- numpy
- scipy
- nltk
- scikit-learn
- pandas
- beautifulsoup4
- jsonschema
- tmdbv3api
- lxml
- urllib3
- requests
- pickleshare

VI. RESULT AND IMPLEMENTATION

This segment portrayed the examinations and their outcomes during testing of the model. It likewise results a few related techniques to begin investigation on existing works and our suggested work to precisely lay out possible benefits of proposed structure.

- **Dataset**

Here IMDB 5000 Movie Dataset has been considered and partitioned into test information. The investigations are executed on two unique releases of the dataset. In the trial and test information, all information comprises of IMDB 5000 Movie Dataset.

- **Evaluation**

. It was observed that BIGRU shows better results obtained for all the test parameters we used for comparison. Recently, many studies have been conducted to predict and recommend products to buy in the near future through customized analysis of individual users, and applications such as Netflix recommending movies and recommending Amazon products are on the rise. In order to predict future consumption, user-based or item-based filtering algorithms are typically used. However, these methods are typically based on user-provided rating data, meaning they cannot be predicted without rating data. Therefore, a product without rating data cannot be included in the recommendation list, causing a shortage problem.

- **Implementation**

1. When the user presses the "Generate Recommendation" button it will recommend movies based on his previous ratings.
2. If he is a new user and has not rated any movies then he is expected to search for a random movie or any movie of his interest in the "search" box as shown in figure 1.4
3. Since the user is new and has not rated any movies, he searches for the word "sister" in the search box and all the movies with words 'sister' in them will appear on the screen as shown in figure 1.5
4. Also using this recommendation system user can see some selected reviews and sentiments using sentiment analysis prediction model as shown in figure 1.5

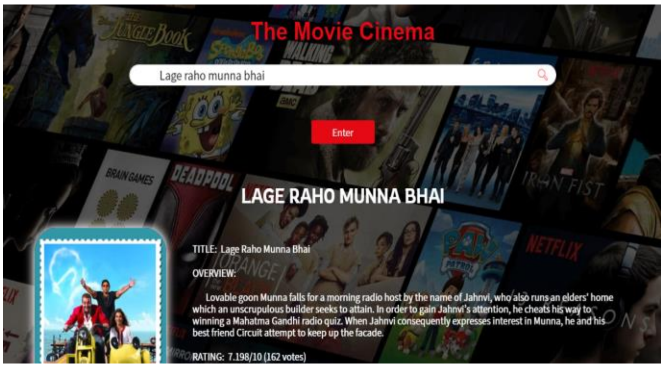

Figure 1.4: search

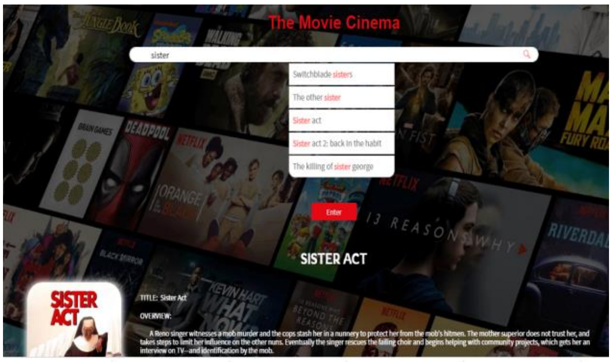

Figure 1.5: search result

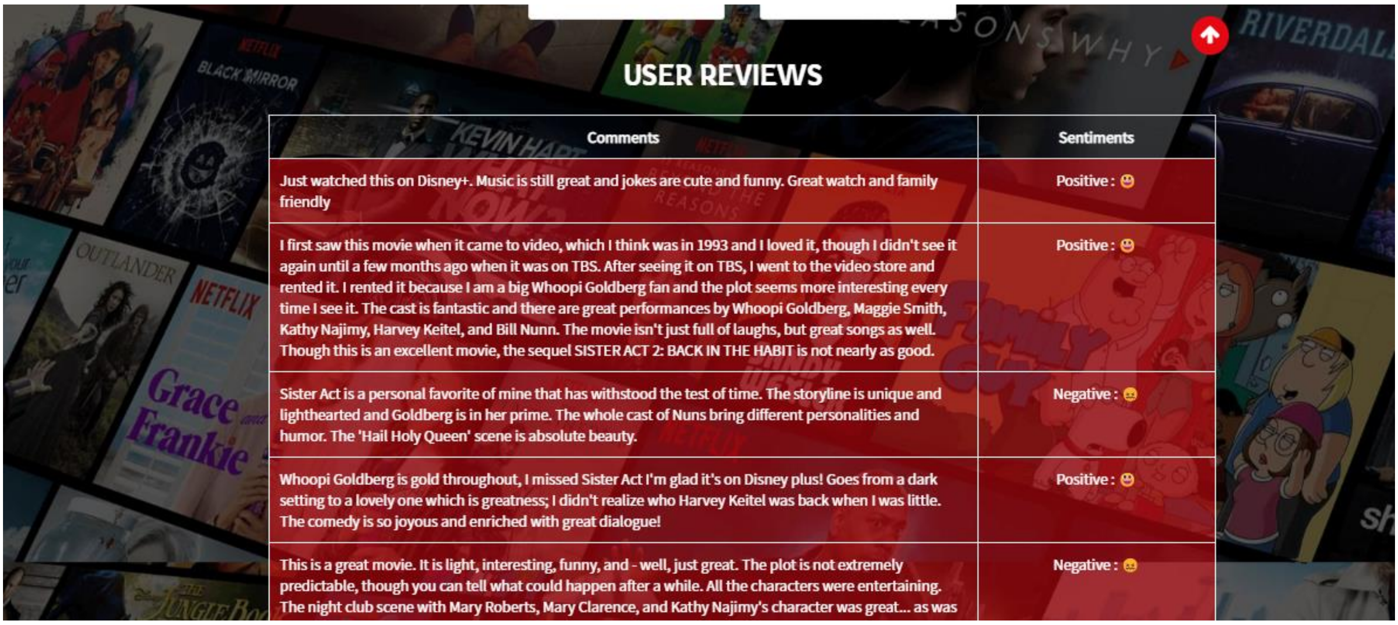

Figure 1.5: sentiment analysis

VII. CONCLUSION & FUTURE SCOPE

We designed ROA for BiGRU to recommend movies to the user based on his interest pattern. Then the obtained results are tested for parameters such as MAE, RMSE, precision, F measure, precision and recall. We compared these results with GA, MMDL, FFNN and found that BGRU with ROA has 97% precision, 97.5% F-measure, 97% precision, 98% recall, which is greater than the rest of all and MAE 0.03, RMSE 0.17, which are the lowest of all the rest. So, we conclude that BiGRU with ROA has better performance for movie recommendation. In the future, ROA can be used for the latest machine learning algorithm for movie recommendations.

In this paper, we classify similar groups of users with similar taste preferences through a dataset of movie ratings for “movie” items and apply an RNN learning method to them. In this way, the movie consumption pattern of a similar group of users is discovered and a model is proposed to recommend movies to users. This model overcomes the sparsity problem, which is the biggest problem of current recommendation systems based on existing rating data. It can also recommend movies with dynamic changing in mind

consumption patterns over time. In addition to the "movie" item, the recommendation model can also be used as a recommendation system in areas such as books and clothing, which have individual tastes and are likely to change over time.

In this article, we predicted one movie that a given user is likely to consume out of 45,000 movies. For more practical applications, multiple recommendation methods that recommend multiple similar movies simultaneously with respect to genre, actor, director, etc. will be more desirable.

VIII. REFERENCES

[1] Mira Kartiwi, Teddy Surya, Sukuk Rating Prediction using Voting Ensemble Strategy, International Journal of Electrical and Computer Engineering (IJECE),Vol. 8, No.1, ISSN: 2088-8708, February 2018, pp. 299-303

[2] Zhang Yuan, Neural Network Based Movie Rating Prediction, Association for Computing Machinery, Shenzhen China ACM ISBN 978-1-4503-6426-3/18/04

[3] Muhammad Ibrahim,Bakhtiar Kasi, A Neural Network-Inspired Approach for Improved and True Movie Recommendations, Hindawi Computational Intelligence and Neuroscience Volume 2019, Article ID 4589060,2019

[4] Prakash P. Rokade, Aruna Kumari D., Business recommendation based on collaborative filtering and feature engineering – proposed approach, International Journal of Electrical and Computer Engineering (IJECE) Vol.9, No.4, ISSN: 2088-8708, August2019, pp. 2614-2619

[5] Warda Ruheen Bristi, Zakia Zaman, Predicting IMDb Rating of Movies by Machine Learning Techniques, 10th ICCCNT 2019 July 6-8, 2019, IIT - Kanpur, India IEEE – 45670.

[6] Son, J., Kim, S.B., Kim, H. and Cho, S., "Review and Analysis of Recommender Systems," J. of the Korean Institute of Industrial Engineers, Vol.41, No.2, pp.185-208, 2015.

[7] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. “Item-Based Collaborative Filtering Recommendation Algorithms,” in Proc. WWW, pp.285-295, 2001.

[8] Mikolov, T., Karafiat, M., Burget, L. Cernocky J. and Khudanpur, S., “Recurrent neural network-based language model,” in Proc. ISCA (International Speech Communication Association), pp.1045- 1048, 2010.

[9] Sutskever, I., Vinyals, O. and Le, Q.V., “Sequence to Sequence Learning with Neural Networks,” in Proc. NIPS(Advances in Neural Information Processing Systems), pp.3104-3112, 2014.

[10] Zaremba, W., Sutskever, I. and Vinyals, O., “Recurrent Neural Network Regularization,” CoRR abs/1409.2329, 2014.

[11] Sundermeyer, M., Schluter, R. and Ney, H., “LSTM Neural Networks for Language Modeling,” in Proc. ISCA(International Speech Communication Association), pp.194-197, 2012.

[12] Yu, F., Liu, Q., Wu, S., Wang, L. and Tan, T. "A Dynamic Recurrent Model for Next Basket Recommendation," in Proc. SIGIR, pp.729- 732, 2015.

[13] Liu, D.Z. and Singh, G. "A Recurrent Neural Network Based Recommendation System," 2015.