ISSN: 0971-1260
Vol-22-Issue-3-July-2019

Design and Implementation of Movie Recommendation System using Meta-Hybrid Approach

Neha Sharma
School of Computer Science & Engineering
Lovely Professional University, Phagwara, Punjab, India
neha.23413@lpu.co.in

Aditya Sisodiya
School of Computer Science & Engineering
Lovely Professional University, Phagwara, Punjab, India
aditya.sisodiya6743@gmail.com

Abstract – A recommendation system is a system based on some data set that provides users with recommendations for certain tools such as books, films, songs, etc. Generally, film recommendation systems predict what movies a consumer will like based on the attributes present in previously liked films. These recommendation systems help organisations that collect data from large amounts of consumers and want to provide the best possible recommendations efficiently. Other considerations can be taken into account when creating a film recommendation system, such as the type of the film, the actors involved in it or even the film director. The proposed paper studies and reviews w the techniques used for recommendation of movies and which among them is best and based on the result proposed our own architecture for the hybrid recommendation system. This paper includes details of implemented model, its architecture and methodology. The research paper also defines how our proposed model is robust, efficient, and effective in working with good performance in turn returning best result.

Keywords- Recommendation System, Hybrid system, Collaborative filtering, content filtering

I. INTRODUCTION

Every time, when you sit back and access internet for searching the online resources such as movies, TV shows, music, games, books, and newsyou always end up with website providing you bulk of data that is available online. From this bulk data, then you need to search for what you are looking for or what you may like. This consumes your lot of time, and many times you end up with no result. Also, as today data available online is increasing with an exponential rate which also makes the representation and providing quality results. Even though many of these websites are successful in providing an excellent user interface but however, they fail to provide more personalized user experience and ends up with losing users from their website. Another problem which arises is due to rapid increase in the number of user online. Answer to need of personalized experience was come out to be the **Recommendation/ Recommender System.** So, what is a recommendation system?

Wikipedia define recommendation system[1] as,a subclass of information filtering systemthat seeks to predict the "rating" or "preference" a user would give to an item. In general term, a recommendation system uses the experience of a user, to predict what other item/content that user may like. This recommendation can be based on user-user similarity or item-item similarity[2]. This system is part of machine learning and has been a hot topic of research ever since introduced. The reason why it is so popular is to improve the recommendation quality or if explained in the term of machine learning to reduce the loss of training or testing.

The popularity of the recommendation system can also be proved based on the competition organized by the Netflix, one of the giant hosting shows and movie online and provide best in class personalised experience to user. The challenge is known by the name Netflix Prize, which was an open challenge. It was started in year 2006 and ended in year 2009. The challenge was to create a collaborative filtering algorithm which beats any previous algorithm that exists. Since then there have been various similar challenges hosted till now so that we can come up with a perfect recommendation system. Today recommendation system finds in application in every field, entertainment, finance, E-commerce, Travel, Advertisement etc.

II. LITERATURE REVIEW

Along with help of Natural Language Processing, Michael Fleischman et al. identified the problem of less available information regarding user choices as one of product similarity calculation. They described their two algorithms as a new word-space method and more of a urbane approach and evaluated the performance in comparison to gold, commercial and baseline approaches [3]. A. Saranya et al. implemented a probabilistic matrix factorization system and provided users with the ability to solve problems with cold start. Their system takes into account the different ranges of film ratings given by review experts [1]. Eyrun A. Eyjolfsdottir et al. introduced a system that was implemented using a hybrid recommendation approach to clustering and machine learning. Our program enters the personal information of the user and predicts the tastes of the user's movies with the aid of the well-trained support vector machine (SVM) models [4].Chen et al discuss process-oriented system architectures, main models, key structures, tests, and standard context-aware RS applications [5].

Yang et al propose a CF-based RS model based on a variety of user data including user ratings and user historical behaviour, contrasting several standard CF algorithms [5]. CF-based RS recommends things to an active user based on their neighbour'sviews [6].Mobile cloud computing (mcc) can save energy, enhance user experience and application. All the above-mentioned systems have their own advantages and problems but are not yet up to the level to solve all security, energy and user experience issues [7].Some of the works also concentrate on the implementation of cooperative filtering techniques in the film recommender system. It increases the reliability of recommendations and therefore the performance of them [8][9]. Many programs take user reviews into account to optimize the movie list.Wakil et al. focused on the human emotion-based Movie Recommender System (MRS).The challenge is that MRS needs to accurately capture the customer's profile and film characteristics, because film is a dynamic domain and emotions are a human interaction domain, so hard to combine in the current Recommenders System (RS)[10]. Rahul Katarya reviewed various techniques used for recommendation systems like context, collaborative, social, group, tag, hybrid and various others [11].

III. APPROACHES FOR RECOMMENDATION SYSTEM

a. CONTENT FILTERING

Content-based filtering methods can be defined on the basis of description of the item and user's preferences. These methods are best suited to situations where there is data which is known on an item (name, location, tags,description, etc.)[12], but not on the user.Recommendations based on content view recommendations as a user-specific classification problem and learn a classifier based on product features for the user's likes and dislikes. Keywords are used in this framework to identify the items and a user profile is created to show the type of item that the user wants.Based on the research performed, we found that there exist two distinct methods for the content-based system.

- Item – Item based, in which similarity between two items are used for recommendation.
- Item – User based, by generating User and Item profile based on the history of user and then provides recommendation to user as shown in Figure 1.

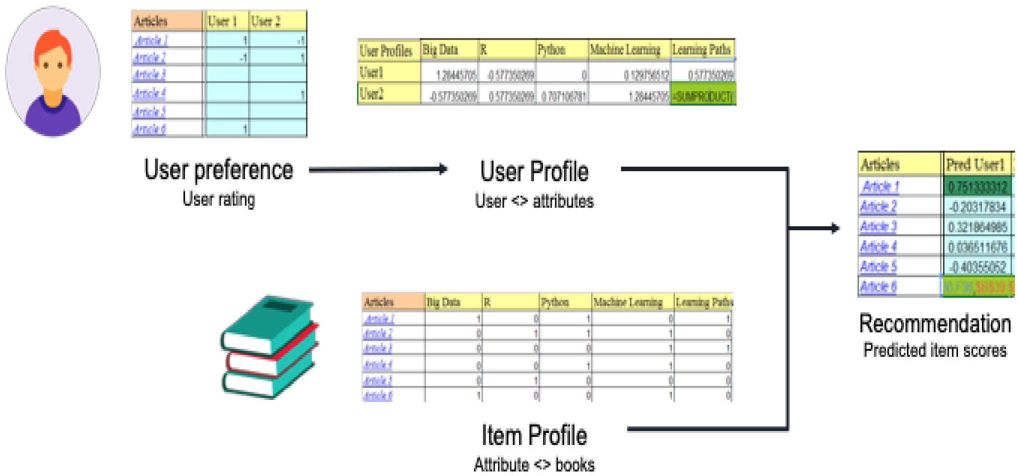

Figure 1 User - Item based Model for Content-based System

b. COLLABORATIVE FILTERING

Collaborative filtering[8][13] is a process of filtering for pattern using technique involving collaboration of agents. Collaborative system is based on assumption that people like things like other things they like, and things like by other people with same taste.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

Basically, it finds collaboration in between users to provide the recommendation to user. It follows User – User similarity concept which is quite like approach one of content filtering explained above.

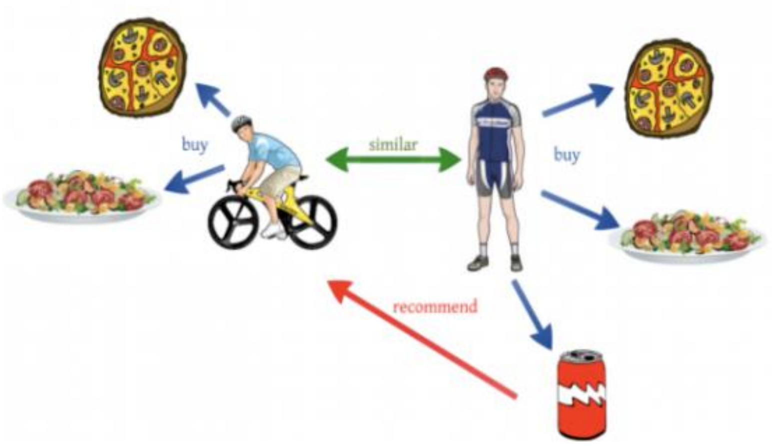

Figure 2 User-User Behaviour

As shown in **Error! Reference source not found.**for instance, a user A likes pizza and salad, and there exist another user B which likes the same thing but also likes coke, then collaborative filter will see similarity between the two user A and B and would say that user A may also like coke and hence is going to recommend that to user A.

Based on research done for collaborative filtering, we found that, there exist multiple ways in which a collaborative filtering system. The most popular are as follow –

- Probabilistic Matrix factorization
- Single Value Decomposition
- KNN
- Deep Learning Based model

The rest description in detail is explained inside.

Using any of the two approaches of recommendation an engine can be built to provide recommendation. However, each model suffers from various disadvantages. And to remove this con of both models, we club them to form a Meta (hybrid) recommendation.

c. HYBRID RECOMMENDATION

A **Hybrid Recommender system** is a system build by combining the two approach of recommendation engine[14][15], content and collaborative. Various other approaches available can also be applied to this. For hybridization of the two models there exist various techniques namely,

- Switching
- Weighted

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

- Mixed
- Feature – Augmentations
- Meta – level

IV. EXISITING SOFTWARE

There exists vivid type of recommendation system since there in action since a long time ago. But however, many stills fail to provide quality recommendation. In technical terms, the trained model has low performance that is high loss. Even the approaches that are available have changed since, so much.

- **Benefit**
  Existing software uses light model which have only enough data to build high performance dataset. The web app is build using PHP based language at backend as these are quick in response resulting in low loading time.

- **Challenges/Problems**
  Most system today working on recommendation neglected the fact the data available (taking in respect to movie dataset), on which they train their model is sparse or the fact that there is no enough rating for a group of movies, due to which bias is inducted to our rating prediction. There is a sparsity of -99.46%

  print("Sparsity –"+ str(round((1.0 – len(rating)/float(n_users * n_movies)) *100, 4)) + "%")

The sparsity in Movie through exploratory data analysis can be seen as shown in Figure 3.

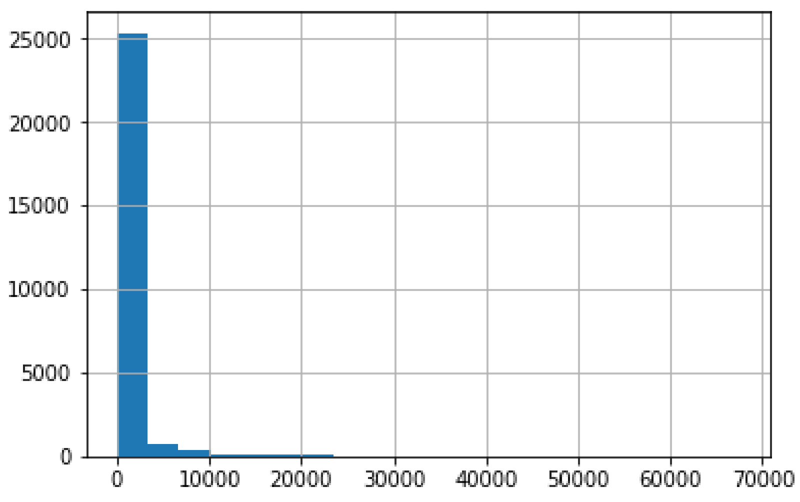

Figure 3Sparsity in Movies

The above figure shows the sparsity on the number of rating each individual movie holds from the Figure 3we can see the sparsity that is some movies have move rating count of

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

up to 10000 while for some this count tends to zero this give sparsity when represented in matrix form.

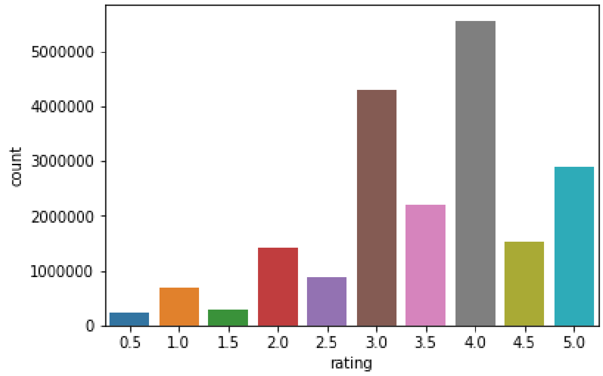

Figure 4 Rating Count for Movies

The count for each rating value can also be plotted and the graph can be seen in Figure **4**.The outliers and scatter/ density plot for movies can be plotted and is displayed as shown in Figure 5.

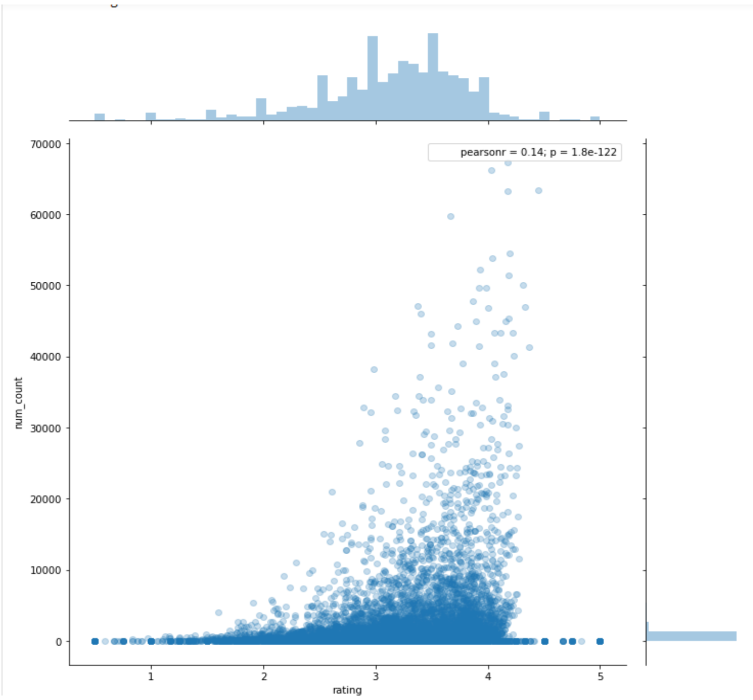

Figure 5 Density and Scatter Plot for Movie Dataset

Sparsity of user data refers to the difficulty of finding sufficiently trustworthy similar users, as active users typically rated only a small portion of items. User rating sparsity is shown in Figure **6**.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

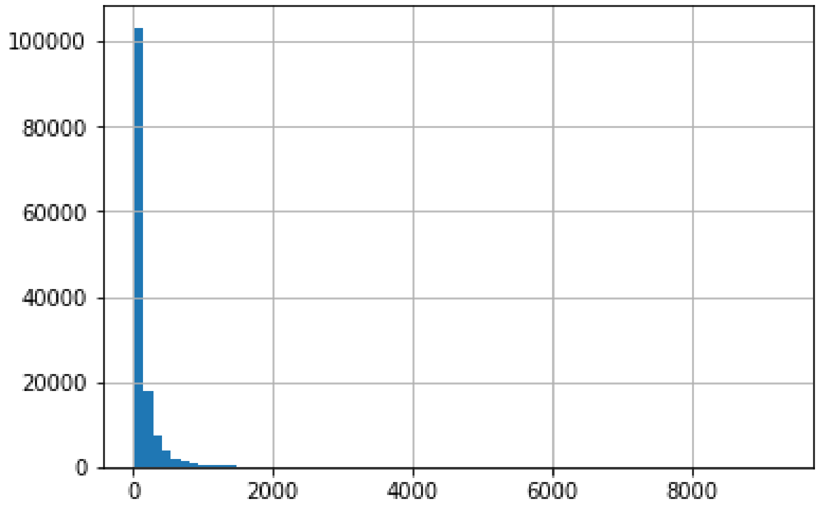

Figure 6 Sparsity for User Rating

Based on this conclusion form the exploratory data analysis, we conclude that most data we have for movie and user is sparse this reason using any single model of two going to yield poor result. This is the problem with the existing system that is no regard to this scarcity.

Based on the above analysis we conclude that the existing system are not robust and need to improve. Hence the hybrid model comes to play to generate a robust model

V. NOVELTY

Based on the existing system and the analysis done, we have research and found that the above problem of sparsity can be dealt with the help of hybrid approach for recommendation system. Hybrid system uses approach of both the models in Figure 7.

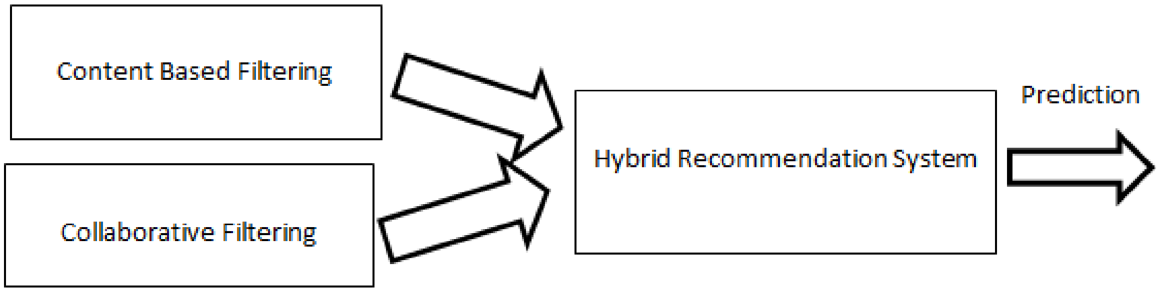

Figure 7 Hybrid System

This would help us tackle the sparsity. Furthermore, to improve the accuracy of the system we would use different approach for content and collaborative filtering.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

- Collaborative Filtering

Based on what we have define earlier, there exist various approach on which a collaborative engine can be some of which are

- **Memory Based,** majorly work on finding similarity between either items or users. For similarity, it uses Cosine Similarity or Pearson correlation coefficient.
- Using **K-Means** approach, it's a model-based approach, using clustering method to find which group a user belong and recommend the movies accordingly.
  - **Single value decomposition,** here for the purpose if recommendation we work as, we break down our original sparse matrix into a 2 low orthogonal matrix product and then predict the value.
  - **Probabilistic Matrix Factorization,** the principle behind these models is that a small number of unknown variables will decide the user's attitudes or preferences. We can call these factors as Embeddings**.** Matrix decomposition can be reformulated as an optimization problem with loss functions and constraints. Embedding can be understoodas low dimensional hidden factors for items and users**.**
- **Deep Learning Multi-layer model,** there is a ton of research material on collaborative filtering using matrix factorization or similarity matrix. But there is lack on online material to learn how to use deep learning models for collaborative filtering. In this we determine the movie embedding based on the deep learning model.

Out of this approach for the Collaborative filtering, we need to find the performance and determine the best mode that we will be using for our approach.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

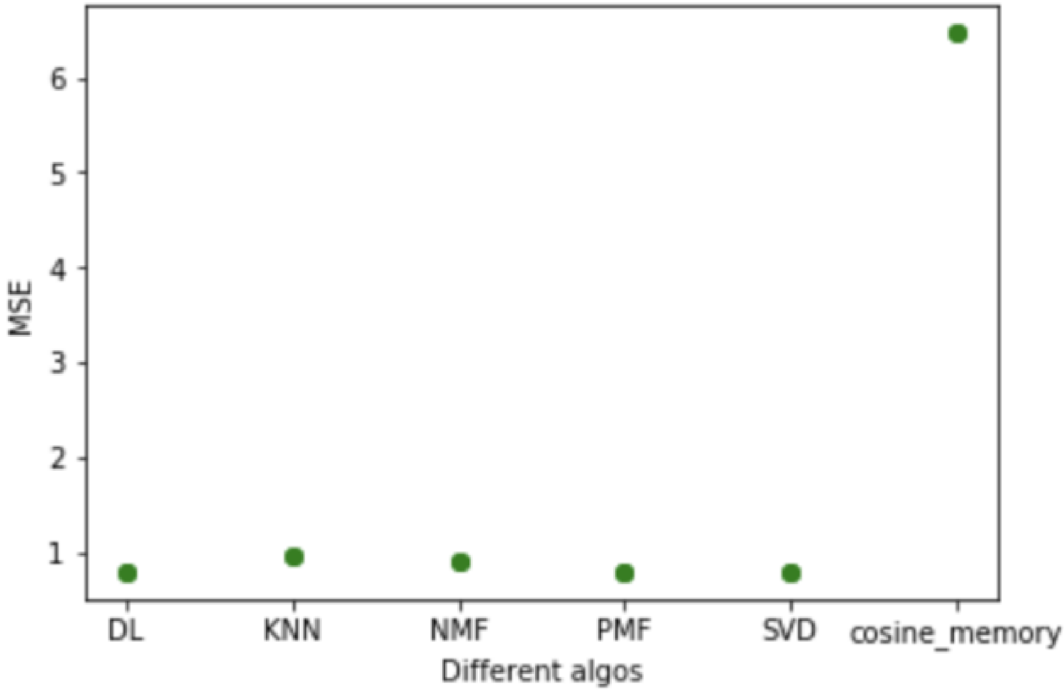

Figure 8 Loss Comparison between different Collaborative Approaches

The graph as shown in Figure **8** represents the mean square error of all the approaches discussed above. From this graph we can conclude that deep learning-based model or SVD model have the lowest error rate among all.

Based on the above conclusion, for our project we have decided to use the deep learning-based model for our project

- **Content – Based Filtering**
  As in collaborative based, in similar manner, content filtering must approach
  - Item-Item Based
  - Item-User Based

For the project, we have implemented Item-User based similarity along with an Auto encoder**.** Auto encoder is application of machine learning of deep-learning, in which the input and the output size is same. Below is the graphical representation of the autoencoder**.**

VI. METHODOLOGY

For the recommendation system we have proposed that is meta-hybrid recommendation system, we decided to use item-user based of content recommendation system, and a deep neural network model for collaborative system, and combine the result of both to construct the hybrid use meta model, according which the result o one model is sent to the other constituting the hybrid.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

For building the content model, we use review of users given for the movie as the keywords or bag of word or document which is converted into vectors using tiff vectorizer and then encode the result using auto encoder. An autoencoder is deep – learning based model which helps encode or decode the feature. The reason for using this is that, the result generated from tiff vectorizer is very large and sparse represented in 19456 x 19456 matrix and to reduce this sparsity we have proposed the use if autoencoder. The autoencoder returns the encoded matrix based on which we find the similarity between different movies which intern helps us to predict the best recommendation for the movie. The architecture of model is shown in Figure **9**.

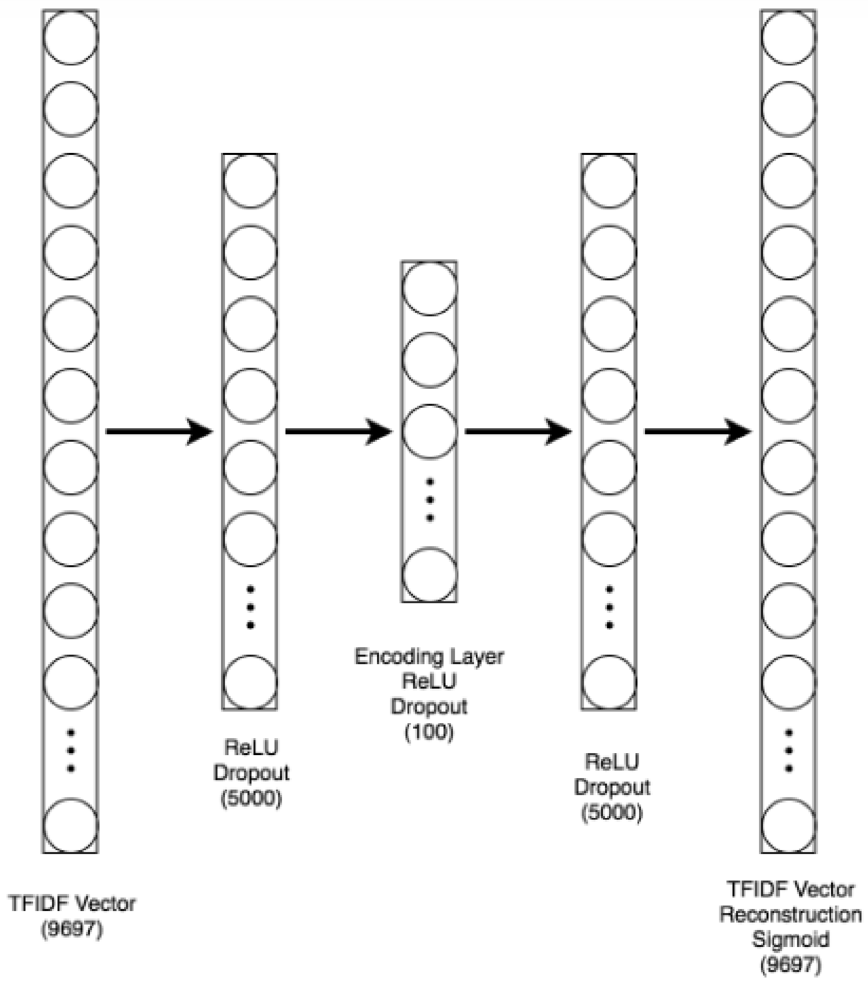

Figure 9 Architecture of Collaborative Model

Once done with that, we build the collaborative filtering model, defining the proper architecture for model with 3 hidden layers having 128,64,32 neurons respectively and then training the deep learning model using Adam algorithm, to optimize it.

The result generated from the content based is then passed to collaborative which based on user-user based similarity and then finding the result based on cosine similarity.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

VII. PSEUDO CODE

i. Content Based Filtering

Step I Import the require dataset and libraries
Step II Generate the bag of word/ Document which contain all the tags.
Step III Convert the text data to numeric data using tf-idf vectorizer
Step IV Encode the generated matrix using auto encoder
Step V Generate the Embedding

ii. Collaborative based Filtering

Step I Import the data and libraries
Step II Initialize the deep learning model with 5 layers, having input layer, layer with 200, 100, 50 neurons respectively, and output layer with single neuron.
Step III Define each layer with ReLu activation function.
StepIV Train the model with train data using Adams algorithm, and save the trained model
Step V Generate the Embedding and predict he recommendation

iii. Similarity Function of Hybrid model

Step I Get the Embedding from both the model
Step II Predict the rating for all movies
Step III Find hybrid result based on any one of the methods, by finding average or cosine similarity.
Step IV Sort the value and generate top recommended movies for the user and forward result to the flask app.

VIII. RESULTS

In this research we were successfully able to build the proposed architecture using both content and collaborative filtering with and with mean absolute loss for measuring the performance of the model, as .665 which means that the model, we have trained for the recommendation is effective and efficient.

IX. CONCLUSION

In the end of the research, we learned about various type of implementation of recommendation system, how they are implemented and how to improve the model based on hyperparameter tuning. Also, on the Meta model proposed in this paper. Turns out to have meant absolute loss, technique to determine the loss of the model, is

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

0.66 for collaborative filtering and .64 for content-based model and 0.665 for the complete model which help us to provide the effective recommendation to any user.

X. FUTURE SCOPE

The recommendation system has endless scope and various branches which are yet to explore. Slowly we are moving towards an era where recommendation will play an important part in eachdevice, we use in our day to day life, whether it's our wardrobe, television, smartphone, etc. fulfilling its respective purpose. Other than that, we can also extend the recommendation engine we have built by extending the level of hybridization. For instance, we can add the layer of sentiment analysis to identify the mood of user at an instance of time and then use this mood in content-based system and further to collaborative system. Other than that, we can also define new architecture for the deep learning model. For this we can apply the hyperparameter tuning, which would be able to learn network property, and tell us best architecture so that we can improve the accuracy of model.

REFERENCES

[1] M. T. Scholar, "Movie Recommendation System," vol. 04, no. 01, pp. 59–66, 2018.

[2] A. Anandhan, L. Shuib, M. A. Ismail, and G. Mujtaba, "Social Media Recommender Systems: Review and Open Research Issues," *IEEE Access*, vol. 6, no. Figure 1, pp. 15608–15628, 2018.

[3] M. Fleischman, E. Hovy, and M. Rey, "Recommendations Without User Preferences : A Natural Language Processing Approach," pp. 242–244, 2003.

[4] M. Singh, "On Content Based Recommender System Using Structured Data Singh Under the guidance of Professor . O . P . Vyas By : Monica Allahabad," no. May 2014, 2019.

[5] R. Chen, Q. Hua, Q. Gao, and Y. Xing, "A Hybrid Recommender System for Gaussian Mixture Model and Enhanced Social Matrix Factorization Technology Based on Multiple Interests," vol. 2018, 2018.

[6] P. Moradi and S. Ahmadian, "Expert Systems with Applications A reliability-based recommendation method to improve trust-aware recommender systems," *Expert Syst. Appl.*, no. May, 2015.

[7] S. Basu, G. Kannayaram, S. Ramasubbareddy, and C. Venkatasubbaiah, *for Monitoring of Virtual Machines in Cloud Environment*, vol. 2, no. February. Springer Singapore, 1927.

[8] A. S. Shree and R. S. Sharma, "of the Hovering Quadcopter," pp. 15–26, 2017.

[9] D. Roy and A. Kundu, "Design of Movie Recommendation System by Means of Collaborative Filtering," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 3, no. 4, pp. 1–6, 2013.

[10] K. Wakil, R. Bakhtyar, K. Ali, and K. Alaadin, "Improving Web Movie Recommender System Based on Emotions," *Int. J. Adv. Comput. Sci. Appl.*, vol. 6, no.

ISSN: 0971-1260
Vol-22-Issue-3-July-2019

2, 2015.

[11] R. Katarya, "A systematic review of group recommender systems techniques," *Proc. Int. Conf. Intell. Sustain. Syst. ICISS 2017*, no. Iciss, pp. 425–428, 2018.

[12] R. Chen, Q. Hua, Y. S. Chang, B. Wang, L. Zhang, and X. Kong, "A survey of collaborative filtering-based recommender systems: from traditional methods to hybrid methods based on social networks," *IEEE Access*, vol. 6, no. c, pp. 64301–64320, 2018.

[13] X. Wang, S. Member, and L. E. I. Lei, "A Survey of Collaborative Filtering-Based Recommender Systems for Mobile Internet Applications," vol. 4, pp. 3273–3287, 2016.

[14] B. K. Ye, Y. J. Tu, and T. P. Liang, "A hybrid system for personalized content recommendation," *J. Electron. Commer. Res.*, vol. 20, no. 2, pp. 91–104, 2019.

[15] K. N. Jain, V. Kumar, P. Kumar, and T. Choudhury, "Movie recommendation system: Hybrid information filtering system," *Adv. Intell. Syst. Comput.*, vol. 673, pp. 677–686, 2018.

