Clustering Based Collaborative Filtering Using and Incentivized Penalizing Model

JAKKAMSETTI KEERTHI [#1], V.SARALA [#2]

[#1] MSC Student, Master of Computer Science,

D.N.R. College, P.G.Courses & Research Center, Bhimavaram, AP, India.

[#2] Assistant Professor, Master of Computer Applications,

D.N.R. College, P.G.Courses & Research Center, Bhimavaram, AP, India.

ABSTRACT

In current days the collaborative filtering combined with various kinds of deep learning models is considered as the best model for recommendations for getting strong positive results. A lot of users still try to ignore these reviews due to the limitations that are present in the recommendation model. Also in the current recommendation model all the social reviews may not generate the accuracy and following ratio compared with similar category products. Here we try to collect the opinion on products based on rating prediction method by making the use of social users' sentiment to infer ratings. First, we extract product features from user reviews. Then, we find out the sentiment words, which are used to describe the product features. Here the admin will add a set of pre-defined words which comes under three categories and based on the matching of words from the database, the review was classified in one of the three categories. In this proposed work there is a method to get the comparative report for K-nearest Neighborhood between set of same category products.

Key Words:

Online Social Networks, Deep Learning, Social Reviews, Recommendation Systems, Users' Sentiment, Rating Prediction.

1. INTRODUCTION

K-Nearest Neighborhood Algorithm (KNN) can be used for solve the two main problems like classification and regression problems. This K-NN is mostly used in the real time industry and this is mainly having three aspects like:

1. It is very easy to retrieve the output

2. The calculation time for prediction takes very less time and gives accurate result

3. It is having good predictive power than compared with many other classification methods.

Now let us elaborate some example to use KNN in overall scale:

	Logistic Regression	CART	Random Forest	KNN
1. Ease to interpret output	2	3	1	3
2. Calculation time	3	2	1	3
3. Predictive Power	2	2	3	2

From the above tabular representation the KNN algorithm is having very clear and fair reviews for all the different parameters which we considered to test the efficiency of classification. Hence in this proposed application we try to apply K-NN algorithm for getting the rating of reviews which are posted by the admin to get the match score for that appropriate product based on social reviews. The K-NN algorithm is assumed to be as one of the best instance-based learning, which is mainly used for finding the approximate distance from one object group to a new object which is found to be added recently. This will try to find out the distance between the new object and for the existing group and then check in which group this new object should be assigned. As we all know that this algorithm has very positive impact compared to many primitive algorithms of classification, this may increase its accuracy for some more level by normalizing the data.

Data Mining (DM) is the process of extracting the useful information from a large data source in order to extract the useful information from that raw data. In general the data mining process will require some integration of techniques from multiple area's such as statistics, machine learning, database technology, and spatial data analysis. The process of DM requires a very keen observation by taking a set of algorithms and also the task which is accomplished for that user. Almost several type of algorithms are used to fit the model in best way. In this proposed application we try to use deep collaborative filter for mining the text reviews given on set of products and also try to identify the rating present for that products or post.

In general the data mining algorithms can be classified based on the following ways like:

1. **Model Based Approach:**

   This approach is mainly used to identify the purpose of the algorithm which is required to fit the data.

2. **Preference Based Approach:**

This is another form of approach in which we try to identify the preference or criteria that is used to execute the task.

3. **Search Based Approach:**

This is the third model in which we try to identify the data based on search time complexity.

So based on the above methods we can able to classify the algorithms and then perform the process of information extraction

2. LITERATURE SURVEY

Literature survey is that the most vital step in software development process. Before developing the tool, it's necessary to work out the time factor, economy and company strength. Once this stuff is satisfied, ten next steps are to work out which OS and language used for developing the tool. This literature survey is mainly used for identifying the list of resources to construct this proposed application.

MOTIVATION

Two well-known authors M.Michael and D. Ekstrand [6]: proposed the concept of Collaborative Filtering Recommender Systems. In this paper the authors mainly concentrated on the various personalities that are exhibited by different recommender algorithms to show that a recommendation is not fixed to one nature. They also discussed about the deployment factors which are required for construction the collaborative recommendation model for getting the recommendations in an accurate manner. In this paper they mainly discuss about the wide variety of choices which are present for the researchers to get the clear idea about this novel recommendation systems.

Two well-known authors Rajeev Kumar and Shyam Sunder Rastogi [7]: proposed the concept of Social Popularity based SVD++ Recommender System. In this the authors mainly discussed more about the incredible growth of Web 2.0 web sites and the challenges that are raised for the traditional recommender systems. In the primitive recommender systems they always use to ignore the social interactions which are present for the end users and almost all the recommendations are one to one. In this paper the authors mainly discussed about the social popularity which is incorporated in SVD++ factorization method to improve accuracy and scalability of recommendations.

Two well-known authors Pierre Baldi and Peter Sadowski [8]: proposed the concept of Understanding Dropout. In this the authors mainly discussed more about the dropout function and how this is relatively connected for training neural networks. Here they introduced a concept like identifying the dropout on either units or connections, with arbitrary probability values, and use it to analyze the averaging and regularizing properties of dropout in both linear and non-linear networks. In general for characterization of deep neural networks, we provide a detailed estimation and bound function for

performing the simulation results. Also we try to discuss about the probability of drop out how much it will really impact the social networks.

3. PROPOSED DEEP COLLABORATIVE SYSTEM USING K-NN ALGORITHM

In this section we mainly discuss about the proposed collaborative system using K-NN algorithm for rating prediction and recommendations of social reviews.

PRELIMINARY KNOWLEDGE

The K-NN algorithm works as follows:

Step-1: Initially try to select the number of neighbors I.e. K- neighbors

Step-2: Next try to calculate the Euclidean distance of **K number of neighbors**

Step-3: Try to take the the K nearest neighbors as per the calculated Euclidean distance.

Step-4: Try to verify how many data points are present in each category.

Step-5: Next try to assign the new data point for that category and check the maximum number of neighbors present in each group.

Step-6: Our model is ready.

DISTANCE CALCULATION IN K-NN

$$\text{dist. } (X^T, X) = \sum_{i=1}^{n} \text{dist}A_i\,(X^T \cdot A_i,\ X \cdot A_i)$$

Where X^T is the test tuple, X is a nearest neighbor, and Ai (i=one to n) represents the attributes of the data points

The weighted sum of local distances is known as global distance. The attributes Ai can be assigned specific weights wi to depict their level of importance in deciding the appropriate classes for the samples. The weights usually range between 0-1. Irrelevant attributes are assigned a weight 0.

$$\text{dist}(X^T, X) = \sum_{i=1}^{n} w_i \times \text{dist}A_i(X^T \cdot A_i\ ,\ X \cdot A_i)$$

The average distance is given as follows:

$$avgdist(X^T, X) = \sum_{i=1}^{n} w_i \times distA_i(X^T \cdot A_i , X \cdot A_i)$$

$$\sum_{i=1}^{n} w_i$$

4. IMPLEMENTATION PHASE

Implementation is the stage where the theoretical design is converted into programmatically manner. In this stage we will divide the application into a number of modules and then coded for deployment. The front end of the application takes JSP, HTML and Java Beans and as a Back-End Data base we took My SQL data base. The application is divided mainly into following 2 modules. They are as follows:

1. Admin Module
2. User Module

1. ADMIN MODULE

In this module, the Admin has to login by using valid user name and password. After login successful he can do some operations such as View All Users and Authorise,Add posts, View All posts with ratings, View All Movie Recommended Posts, View All service reviewed posts, View All Search History, View Collaborative Filtering based on Recommendation, Find top-K hit rate in the chart.

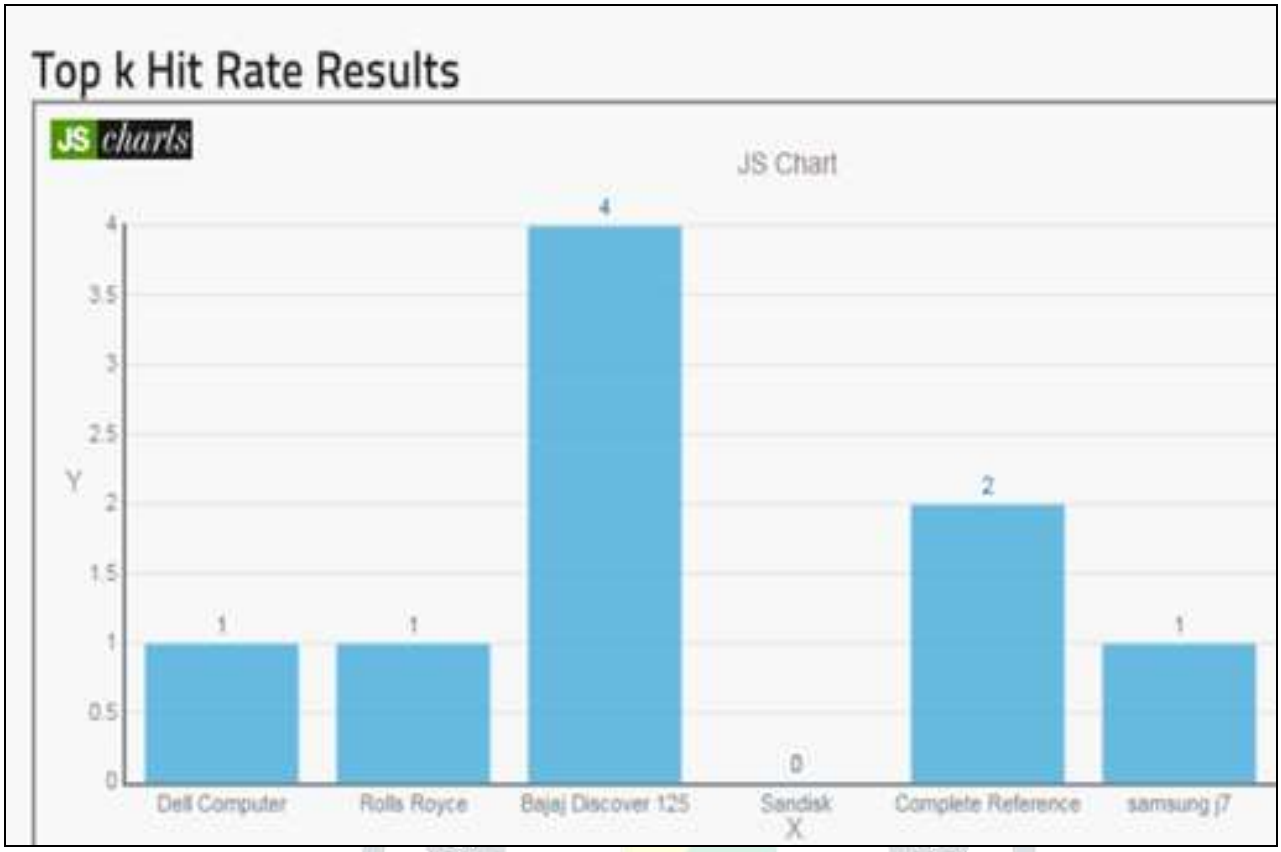
2. USER MODULE

In this module, there are n numbers of users are present. User should register before doing some operations. After registration successful he has to login by using authorized user name and password. Login successful he will do some operations like View My Profile, Search friend and accept friend request, Search for Post, My Accounts, View Recommendations, View All users interest on the Individual Post, View top K hit rate.

5. EXPERIMENTAL RESULTS

We have conducted experiment on several products all collected from Google server and they are added into the offline application and now we try to check rating prediction and recommendation model for set of social reviews. Here we try to execute the application on local server like tomcat and then try to check the performance of our proposed application in terms of identifying the Cyberbulled messages and non-Cyberbulled messages so accurately. Here for designing the application we use **AVA** as the programming

language and **MYSQL** as the backend database for storing and retrieving the information to and from the database. Here we used **HTML,JSP and CSS** as the front end technologies to design the **GUI** and then collect the reviews from various social users. From the below graph we can clearly identify the hit rate results for all the posts which are uploaded by the admin into the database. The X-axis represents the type of products and Y-Axis represents the number of persons who rated those products in positive manner.



6. CONCLUSION

In this paper we for the first time have construct a DEEP Collaborative filtering we try to collect the opinion on products based on rating prediction method by making the use of social users' sentiment to infer ratings. First, we extract product features from user reviews. Then, we find out the sentiment words, which are used to describe the product features. Here the admin will add a set of pre-defined words which comes under three categories and based on the matching of words from the database, the review was classified in one of the three categories. In this proposed work there is a method to get the comparative report for K-nearest Neighborhood between set of same category products.

7. REFERENCES

[1] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, ``Collaborative filtering recommender systems," *Found. Trends Hum.-Comput. Interact.*, vol. 4, no. 2, pp. 81_173, 2011.

[2] R. Kumar, B. K. Verma, and S. S. Rastogi, ``Social popularity based SVDCC recommender system," *Int. J. Comput. Appl.*, vol. 87, no. 14,pp. 33_37, 2014.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ``Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Pro-cess. Syst.*, 2012, pp. 1097_1105.

[4] Y. Jhamb, T. Ebesu, and Y. Fang, ``Attentive contextual denoising autoen-coder for recommendation," in *Proc. ACM SIGIR Int. Conf. Theory Inf.Retr.* New York, NY, USA: ACM, 2018, pp. 27_34.

[5] X. Cai, J. Han, and L. Yang, ``Generative adversarial network based heterogeneous bibliographic network representation for personalized citation recommendation," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 1_8.

[6] Y. Peng, S.Wang, and B.-L. Lu, ``Marginalized denoising autoencoder viagraph regularization for domain adaptation," in *Proc. Int. Conf. Neural Inf.Process.* Berlin, Germany: Springer, 2013, pp. 156_163.

[7] P. Baldi and P. J. Sadowski, ``Understanding dropout," in *Proc. Adv. NeuralInf. Process. Syst.*, 2013, pp. 1_9.

[8] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, ``Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput.Vis. Pattern Recognit.*, Jun. 2009, pp. 248_255.

[9] H. J. Xue, X. Dai, J. Zhang, S. Huang, and J. Chen, ``Deep matrix factorization models for recommender systems," in *Proc. IJCAI*, 2017,pp. 1_7.

[10] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdi-nov, ``Dropout:Asimpleway to prevent neural networks from over_tting,"*J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929_1958, 2014.

[11] Y. Koren, R. Bell, and C. Volinsky, ``Matrix factorization techniques forrecommender systems," *Computer*, vol. 8, pp. 30_37, Aug. 2009.

[12] S. Li, J. Kawale, and Y. Fu, ``Deep collaborative filtering via marginalizeddenoising auto-encoder," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, New York, NY, USA: ACM, 2015, pp. 811_820.