International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 07 | July 2020 www.irjet.net p-ISSN: 2395-0072

SOCIAL NETWORK based SEQUENTIAL USER INTEREST MOEDLING

Miss. Vedanti R. Chavanke[1], Prof. Devidas S. Thosar[2], Prof. Kishor N. Shedge[3]

[1]Student, Computer Engineering Department, Sir Visvesvaraya Institute of Technology, Chincholi, Nashik, Maharashtra, India
[2]Assistant Professor, Computer Engineering Department, Sir Visvesvaraya Institute of Technology, Chincholi, Nashik, Maharashtra, India
[3]Assistant Professor, Computer Engineering Department, Sir Visvesvaraya Institute of Technology, Chincholi, Nashik, Maharashtra, India

---***---

Abstract - *Social network creates online user group and share their experiences, interest and views with each other. To provide better service to users and grow a business, there is need to analyze user interest, need, preferences, and habits. The social circle and influence of people in contact also matters to the users purchase. Sequential actions of friends and temporal auto correlation influences user point of interest. The proposed work includes recommendation generation based on deep learning. A Social-Aware Long Short-Term Memory (SA-LSTM) algorithm is proposed. The SA-LSTM includes stacked LSTMs for sequential modeling and Stacked Denoiseing AutoEncoders SDAEs for social influence modeling. Based on the POI of users, friends of users having same interest are searched in system. System selects top k active users based from friend list by matching the POI of user, recommendation is generated. The system working is tested on epinion dataset and time and accuracy of system is calculated.*

***Key Words*:** Social network, Recommendation system, User interest modeling, deep learning, Recurrent Neural Network,

1. INTRODUCTION

In recent years, use of internet and smart phone devices is increased rapidly. With the help of internet the use of online services such as online booking, online shopping, etc. increases. Along with these services, social network communication is also increased with the help of google plus, facebook, linked in etc. Social network creates online user group and share their experiences, interest and views with each other. To provide better service to users and grow a business, there is need to analyze user interest, need, preferences, and habits. The social circle and influence of people in contact also matters to the users purchase. Analysis of user purchase helps to generate profitable business model.

By analyzing these needs, a recommender system suggests products to the user by analyzing and predicting the user point of interest. Along with social circle, location of user is also maters to define point of interest. Mobile based online shopping and access to the social network helps to get user preferences with geo location information. Irrespective of linear time, location preferences can vary. For example social connection and purchasing strategy of user at working place, casual places and at residence varies. These POI extraction method is not time sensitive. The POIs are not temporally correlated. This is static POI analysis technique where users interest are studied without time or sequence of occurrence constraint.

Shopping portals like amazon, ebay, flipcart provides a facility to user to share the purchase information on social media sites like facebook, google+, etc. People find this as a trustworthy source of getting information about product, and like to follow such product. This sharing is a product recommendation to the nearest community. People find this recommendation most trustworthy than the recommendation generated by system. Friends' friends of friends follow the product and it shows common interest among certain community.

The proposed system aims to design a system to predict user interest based on social influence and temporal autocorrelation aspect. To predict user interest, deep learning technique is used. The system considers the sequence of user activities such as sequence of items that user has purchased online, User point of interest by collecting the products visited history. System also considers the same sequence of user's friends. By collecting this information system predicts the item that may be purchased or visited by the user.

The prediction system helps to generate recommendation to user. Such recommendations are likely lead to hit by the user. Such recommendation system has many applications in variety of domain where recommendation generation help in increase sale or user visits. For example, for restaurant recommendation such system is also useful. Several new restaurants are available but user visits the restaurants which are generally recommended by his/her friends. The proposed system works on the same line and tries to find user interest and provide a trustworthy recommendation based on his/her friend circle latest visits and users having same interest as like the user. Latest visits are identified with the temporal information.

2. RELATED WORK

In recent years many recommendation generation system are proposed. The system users behavior/interest, social influence temporal information, product rating, reviews, etc.

Matrix factorization is a system that generates predictions over rating of a product. Future rating of product is predicted. Based on the future rating value, the product is recommended to the user. The technique uses collaborative filtering. The system uses user-item rating matrix. The matrix is then factories in two lower rank matrices. The one matrix represent the latent factors of users and other matrix represent latent factors of item. Such system do not consider user behavior and not follow the temporal aspect.[2]

To overcome this drawback R. Ronen, E. Yom-Tov, G. Lavee proposes a new system that uses users browsing logs and search query logs. Along with the product rating individual user interest is also considered for recommendation generation.[3]

Similar to matrix factorization method a new AutoRec method is proposed by S. Sedhain, A.K. Menon, S. Sanner and L. Xie, It is also a collaborative filtering method. The system proposes a non-linear auto-encoder model. It tries to reduces the computational overhead and improves the efficiency of system.[4]

Eagle et al. built a system to identify structures in routine. The system builds an analytical result by observing daily behavior of users. It finds eigen vectors and finds principal components in the user's behavioral data set. The dataset contains daily behavior of user. It also finds behavioral similarity between individuals and groups.[5]

S. Isaacman, R. Becker, R. Caceres, proposes a technique to identify important locations in user's life like home ,working place, etc. the technique is based on clustering and regression. It analyses cellular network data. This is a static scenario. It do not focuses on users interest or temporal autocorrelation. It do not ses sequential modeling of data.[6]

A new recommender system is proposed by P. Matuszyk, J. Vinagre.The system uses incremental matrix factorization. This technique keep data up to date by forgetting the old outdated data. The current preferences of users are preserved by eliminating old data. Five new data forgetting techniques are proposed in this system. [7]

Recurrent Recommender Networks(RNN) is present to predict user's future behavioral trajectories. It uses autoregressive model with low-rank factorization. The system aims to guess missing ratings. The system considers the temporal dynamics . It is non-linear recommender system based on Long Short-Term Memory (LSTM). The system dynamically models the user item relationship.[8]

Spatial Temporal spatial temporal prediction method .Recurrent Neural Networks (ST-RNN) is used to predicts next location of user. It uses local temporal and spatial contexts of user. ST-RNN system captures time interval and geographical distance information.[9]

Qin et al. proposes a new algorithm to extract information about social influence and social circle of users. This technique mines users' real friends and partition those friends in different groups according the closeness of their relationship. This is a static information analysis called as cold start problem. The system do not use sequential modeling or any dynamic data. [10]

C. H. Liu, J. Xu, J. Tang proposes a technique to find user and item category relationship and tries to predict next category of product that user may visit or purchased. The system uses Long Short-Term Memory (LSTM) with Recurrent Recommender Networks(RNN). It is auto encoder-based deep model to model social influence. Social influence based only product recommendation is proposed in the system. [1]

3. PROBLEM FORMULATION

Lot of research work is done on product recommendation and friend recommendation. The product recommendation is based on rating and social influence at the current timestamp. The friends are recommended based on the trust factors. Rather than generating the recommendation for individual product a category recommendation generates better results and serving more options. For recommendation generation only social influence on user is considered. The point of interst of user need to be initially identified and recommendation should be made from social circle having same POI. There is need to develop a system that generates category recommendation based on social influence of nearby top k users having same POI as user.

4. SYSTEM OVERVIEW

4.1 ARCHITECTURE OF SYSTEM

Following fig 1 represents the architecture of system. The dataset is input to the system. The dataset contains product category information, timestamp specific user interest information and user friend list information. System analyze the dataset as per the user defined time slot contents and find top k active friends having same interest as user. Based on the generated information SA-LSTM algorithm is executed and recommendations are generated. The generated recommendations are compared with the next slot information and accuracy of the system is evaluated.

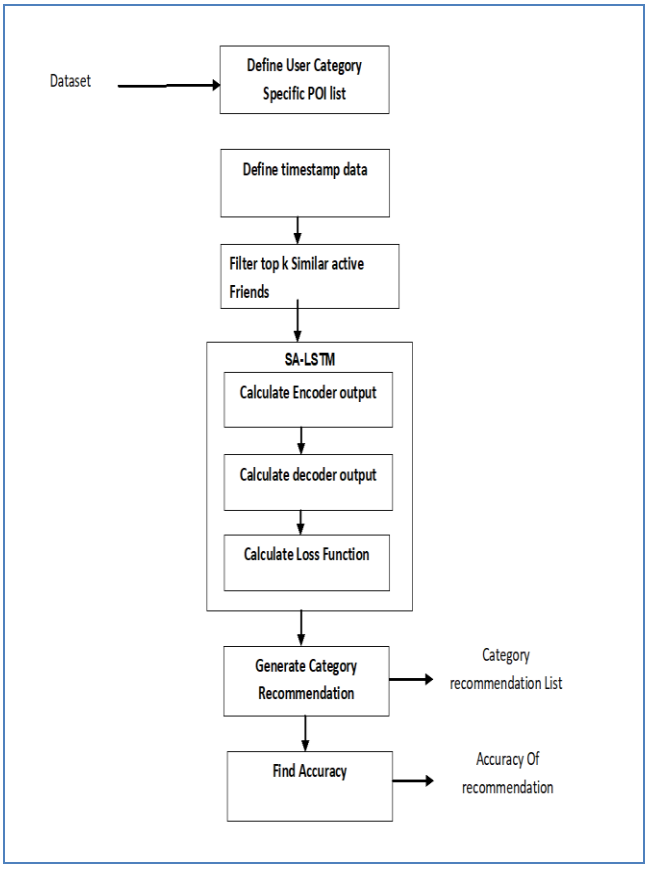

Fig -1: Architecture of System

4.2 SYSTEM WORKING

User point of interest is highly influenced by his/her social circle. The social circle includes user's friends. User would like to visit pages or purchase product based on the social influence. User interests are categorized in C categories. Rather than analyzing single product, category POI analysis gives the generalized global view.

LSTM-based sequential model is proposed in this system to suggest POI categories. Long short-term memory (LSTM) is a recurrent neural network (RNN). LSTM model is made up of 4 units: cell memory, an input gate, an output gate and a forget gate. Social-Aware LSTM includes 2 parts:

1. Stacked Denoiseing AutoEncoders (SDAEs)
2. stacked LSTMs for sequential learning

Following fig. 2 represents the SA-LSTM structure:

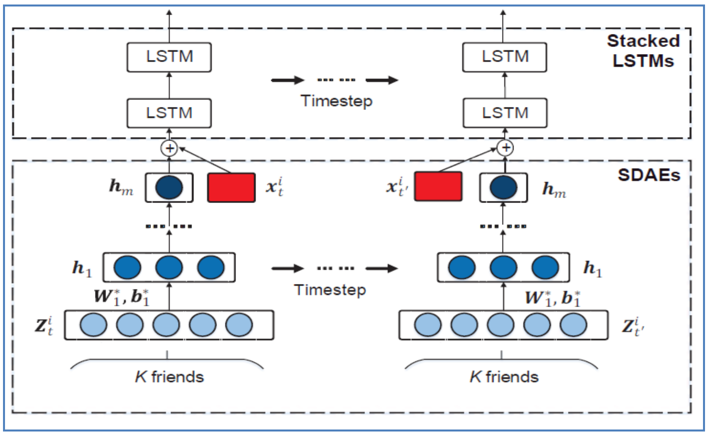

Fig -2: SA-LSTM Model

Top k users friends activities of selected user at timestamp t are input to the SDAE system. The system learns compact social influence. (marked in fig 2 with blue color) The target user representation (marked in fig 2 with red color) is concatenated with input to the LSTM at timestamp t. After getting predictive category list, Friend of friendliest is filtered to generate friend recommendation. The POI values of all K friends of selected user i are concatenated at timestamp t. It is denoted as Z^i_t.

Social Aware Long short-term memory SALSTM algorithm is used to generate recommendation based on timestamp based sequential data modeling. From the list of active k users POI categories are extracted. Based on Category list of POI of k active friends, SALSTM generates recommendation categories for user.

5. ALGORITHM: SA-LSTM

Input: Category List (c1, c2,…cn)
Target User t
K Friend List fk
K friends POI
Output: Category vector Z
Processing:

1. Split the data using timestamp information in multiple slots
2. For each slot
3. Find User POI
4. Get category specific friends POI
5. Filter friends with same POI
6. Find Top K active friends
7. Qlt: Define layer encoder output for timestamp t
   $q_l^t = \delta(W_l d(Z_t^i) + b_l)$, l=1
   $q_l^t = \delta(W_l d(Z_{l-1}^t) + b_l)$, l Ɛ{2,…m}
   Where δ is activation function
   d is dropout function
8. Plt: Define layer decoder output for timestamp t

$p_l^t = \delta(W_l.d(q_m^t) + b_l)$, l=m

$p_l^t = \delta(W_l.(p_{l+1}^t) + b_l)$, l Ɛ{1,...,m-1}

Where W_l is weight of layer l

B bias of layer l

9. Lae: Calculate Loss function

   $L_{ae}(Z, Z') = \sum_{k=1}^{|Z|}(z_k - z`_k)^2)$

   Where Z is target output values with C categories
10. Suggestion: Filter candidates using Z category list
11. Calculate accuracy of recommendation

6. MATH MODEL

I = {I1, I2, I3, I4}, Set of Input
I1 = target user t
I2 = Category List (c1, c2,…cn)
I3 = Users POI
I4 = Friend List

O={O1,O2}, Set of output
O1 = user POI Category recommendation
O2 = Accuracy of recommendation

F={F1,F2,F3,F4,F5,F6,F7,F8,F9,F10,F11,F12,F13,F14, F15}, Set of function
F1 = Get User POI
F2 = Get friend list
F3 = Get friends POI
F4 = filter friends with same POI
F5 = Get top k active friend
F6 = Generate category specific Z vector for timestamp t
F7 = encoder function
F8 = dropout function
F9 = Define layer encoder output for timestamp t
F10 = calculate weight of layer
F11 = calculate bias of layer
F12 = Plt: Define layer decoder output for timestamp t
F13 = Calculate loss function
F14 = Suggestion: Filter candidates using Z category list
F15 = Calculate accuracy of recommendation

7. IMPLEMENTATION

A desktop based application is generated. This application is developed using Java -jdk1. Netbeans 8.1 IDE is used as a development tool. The system is implemented and tested on core i3 system with 4 gb ram.

7.1. Performance Measure:

1. Time: The system measure time for execution for product recommendation and friends recommendation.
2. Accuracy: This measure checks the relevance in recommendation. It checks number of valid categories recommended to user. It is calculated as:

$$\text{Accuracy} = \frac{\text{No of generated relevant recommendation}}{\text{Total relevant recommendations}}$$

7.2. DATASET:

Epinion dataset is used for testing.[11] The dataset contains 3 files:

1. rating_with_timestamp.txt

The file includes rating given by user to a product with timestamp information. The file contains 6 columns as: userid, productid, categoryid, rating, helpfulness and time point

2. trust.txt

It includes the trust relations between users. Trust represents the friendship. It has 2 columns. Both columns contain user id information.

3. Category List:

   The 27 distinct categories are enlisted with its category id.

8. RESULT

Initially dataset pre-processing is done. The data is preprocessed and divided in number of categories. The data in rating_with_timestamp.txt is splitted in 27 distinct categories for splitting the data for processing.

The algorithm is executed and top m recommendations are generated to the user and accuracy of recommendation is evaluated.

The system is tested for 3 and 5 slots. Each slot is created of size 2 months[1]. The recommendations are generated for each slot and average accuracy of recommendation is generated.

Following table shows the time required for processing and accuracy of existing system[1] and proposed system. The existing system generates recommendation using top k active friends and proposed system filters the friend list as per the user POI and then find the top k active friends with same POI as user. The Proposed system requires higher time for processing as compared to the existing system but the accuracy of proposed system increases as compared to the existing system.

No. of recommendation per user	Execution Time-Existing System (in Sec.)	Execution Time Proposed (in Sec.)	Accuracy Of Existing system	Accuracy of Proposed System
3	145.404	584.356	0.2	0.255556
4	218.605	783.842	0.229167	0.2375
5	399.673	1101.995	0.21	0.29

Table 1: Time and accuracy analysis of Recommendation System for 3 slots

Following figure shows the time and accuracy analysis of existing and proposed system . The accuracy of proposed system increases as compared to the existing system. The processing time is also high for proposed system as it again filters the friend list as per the user POI.

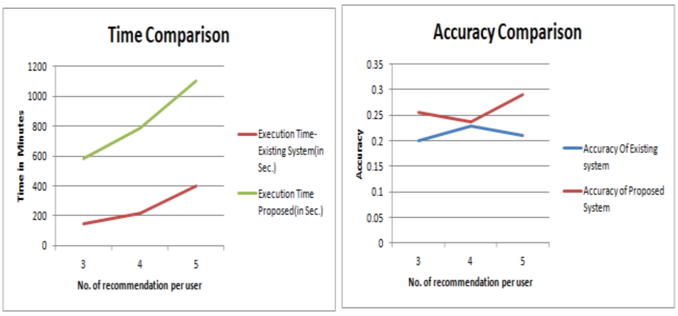

Fig 3: Recommendation Generation results for 3 slots

Following table 2 shows the time and accuracy analysis for 5 slots. As number of slots increases the time required for processing also increases. Proposed system requires higher time for processing as compared to the existing one but generates higher accuracy as compared to the existing system. As number of slot increases the accuracy is also increases (as compared to the table 1). The recommendation accuracy increases for number of recommendation generation count.

No. of recommendation per user	Execution Time-Existing System (in Sec.)	Execution Time Proposed (in Sec.)	Accuracy Of Existing system	Accuracy of Proposed System
3	332.168	1233.496	0.253333	0.336667
4	459.914	1645.894	0.27	0.29
5	1183.861	3431.092	0.284	0.312

Table 2: Time and accuracy analysis of Recommendation System for 5 slots

Following graph shows the time and accuracy analysis for 5 slots. The time required for processing is high for posed system and gradually increases as number of recommendation count increases. The accuracy of proposed system increases than existing system.

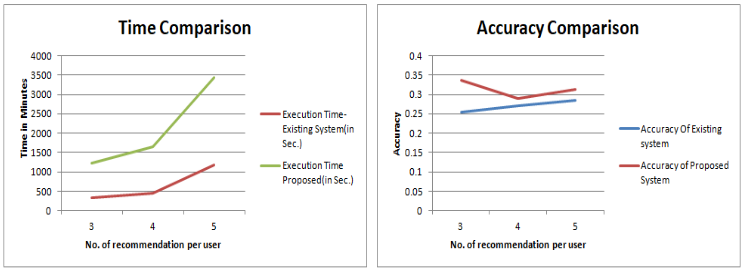

Fig 4: Recommendation Generation results for 5 slots

9. CONCLUSION

A neural network based deep learning approach is proposed for recommendation generation. Based on social influence and user point of interest recommendations are generated. A temporal autocorrelation occurs between user and sequential actions user's circle. Social-Aware Long Short-Term Memory (SA-LSTM) algorithm is proposed based on neural network. This is a hybrid deep learning model. It includes features stacked LSTMs for sequential modeling and an auto-encoder-based deep model for social influence modeling. This algorithm predicts future user point of interest based on the common POI among friends. In future system can be implemented on distributed environment for efficiency improvement.

REFERENCES

[1] Chi Harold Liu, Jie Xu, Jian Tang and Jon Crowcroft, "Social-aware Sequential Modeling of User Interests: A Deep Learning Approach," IEEE Transactions on Knowledge and Data Engineering, Oct. 2018, PP(99):1-1

[2] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems", Computer'09.

[3] R. Ronen, E. Yom-Tov, G. Lavee, "Recommendations meet web browsing: enhancing collaborative filtering

using internet browsing logs", Proceedings of IEEE ICDE'16, pp. 1230–1238.

[4] S. Sedhain, A.K. Menon, S. Sanner and L. Xie, "Autorec: autoencoders meet collaborative filtering", Proceedings of WWW'15, pp. 111-112.

[5] N. Eagle and A. Pentland, "Eigenbehaviors: Identifying structure in routine, Behavioral Ecology and Sociobiology", Vol. 63, No. 7, 2009, pp. 1057-1066.

[6] S. Isaacman, R. Becker, R. Caceres, S. Kobourov, M. Martonosi, J. Rowland, and A. Varshavsky, "Identifying important places in peoples lives from cellular network data", Proceedings of Int'l Conf. on Pervasive Computing, 2011, pp. 133-151.

[7] P. Matuszyk, J. Vinagre, M. Spiliopoulou and J. Gama, "Forgetting methods for incremental matrix factorization in recommender systems", Proceedings of the 30th Annual ACM Symposium on Applied Computing (SAC'15), pp. 47-953.

[8] C.Y. Wu, A. Ahmed, A. Beutel, A.J. Smola and H. Jing, "Recurrent recommender networks", Proceedings of WSDM'17, pp. 495-503.

[9] Q. Liu, S. Wu, L. Wang, and T. Tan, "Predicting the next location: a recurrent model with spatial and temporal contexts", Proceedings of AAAI'16, pp. 194-200.

[10] H. Qin, T. Liu, and Y. Ma, "Mining user's real social circle in microblog", Proceedings of ASONAM'12, pp. 348-352.

[11] https://www.cse.msu.edu/ tangjili/trust.html