A Solution to CoCoS Problem in Recommender System based on SNA

Afshan Shujat
M.Tech Scholar
Department of Computer
Science and Engineering ,
Faculty of Engineering and
Technology, Jamia Hamdard
(Hamdard University), New
Delhi

Md. Tabrez Nafis
Assistant Professor
Department of Computer
Science and Engineering ,
Faculty of Engineering and
Technology, Jamia Hamdard
(Hamdard University),New
Delhi

Vishal Sharma
Assistant Professor
CSE Department, Bharti
Vidyapeeth College of
Engineering and Technology,
New Delhi

ABSTRACT

Recommender systems have become extremely popular in the last decade and are applied to various e-commerce websites. Recommender systems help users find useful, interesting items or content from a considerably large amount of information.

The major challenge in recommender system is Cold-start Problem [1], this appears when a user or item is new, the system may fail because not enough information is available for this user or item. There are so many solutions are proposed to solve cold-start problem including matrix factorization, graph based [2] [3] etc.

One more serious form of cold-start found in some real-time e-commerce applications that is named as CoCoS (Continuous Cold Start) problem[4], this is a recurring version of cold-start even for known users or items, since many users visit the website rarely, change their interests in time, or exhibit different personas. Hence both of the widely used approaches CBF (Content Based Filtering)[5] and CF(Collaborative Filtering)[6] will suffer from this problem.

The most basic assumption -"**similar users will like similar items**" [7] of Recommender System fails in some cases where user interest is changing over time and hence CoCoS problem arises. CoCoS problem is domain specific not all e-commerce website falls under CoCoS, but there are some websites that suffer from CoCos due to the changes in user's interest over time, the best example is travel and tourism domain based websites. Users booked a ticket or package on their current needs and interest by using different websites and we won't be able to make a useful set of suggested items because we don't have sufficient information on user's recent activities or requirements to generate a set of recommended items.

In this paper, we are proposing a solution to get the recent preferences and interest of user on the basis of Social Network Activities (SNA). Here we are targeting travel and tourism domain to state the CoCoS problem and to find out the solution of CoCoS by using SNA (Social Network Activities) . Here we are considering Facebook [8] Check-ins Count (FCC) for obtaining the information about user's recently visited places and to draw a conclusion for user's preferences for next visit.

General Terms

Recommender system Algorithm, Social network activities, continuous cold start problem

Keywords

Recommender system, cold-start problem, continuous cold-start, social network activities.

1. INTRODUCTION

In 1992 first recommender system[9] was developed by Goldberg, Nichols ,Oki and Terry[10]. Recommender system algorithm includes various approaches content-based recommendation[10], collaborative filtering recommendation[11][12], association rules based recommendation[13] and hybrid recommendation[14], collaborative filtering are the most popular method. All these approaches consider that similar users have similar interest and users' preferences are statics.

But this is not true to all domains,in some domains these approaches may fail to generate the useful recommendations for user due to the changes in interests and preferences over time, hence system falls under CoCoS problem. In this paper to address address this problem we are trying to find out the users' recent preferences by analyzing their activities on social network .

Facebook graph api[15] is being used for finding out FCC (Facebook Check-Ins Count), and on the basis of the value of FCC, user's recently visited places are being divided into two sets of places FVP(Frequently Visited Places) and IVP (Infrequently Visited Places). Now assuming that the places with High Facebook Check-Ins Count(HFCC) i.e. FVP are always warm and user is visiting theses places frequently. And the other places with Low Facebook Check-Ins Count (LFCC) i.e. IVP are not necessarily considered as warm because user is not visiting these places frequently.

On the basis of the meta data of these places and other obtained information i.e. LFCC and HFCC, a set of rules would be generated for finding user's preferences and can apply this information in Recommender System algorithm to suggest places to user. In this paper we will discuss the CoCoS problem in travel and tourism domain, a solution based on SNA, methodologies to collect information from Facebook and application of retrieved information in Recommender System algorithm and we will also work on small set of data to draw a conclusion.

2. PROBLEM STATEMENT

Here considering a recommender System RS that is based on travel and tourism domain . The task of RS is to predict preferences of users and provide a list of places that might be preferred by given user for booking purpose. This is usually

accomplished by either estimating the ratings (or likes or favorites counts) that a user U would assign to a place P i.e. CBF or selecting some places by CF and then recommend the best places to user of interest.

Suppose for RS a user U don't provide sufficient rating to places recently (say last 6 months), it means user is either new or inactive since long time on this system. Hence we don't have information about user's recent preferences if user is arriving on the website after long time and RS can't find right places of choices because travelling needs changes over time. In this case we can't use CBF and CF for recommended places to U due to lack of content information and preferences. It means RS is suffering from CoCoS[4] and we need to find out an appropriate approach to address CoCoS.

3. PROPOSED SOLUTION

Here we are proposing a solution to address CoCoS problem with RS by retrieving the information from SNA (Social Network Activities) i.e. FCC, as discussed earlier the system is falling under CoCoS due to the lack of sufficient information about the users' recent preferences for visiting a place. If somehow we will be able to get the information about the users' preference about places of choice so we can solve the problem of CoCoS. Here we are using Facebook as target social network and we are proposing a solution for travelling domain, we are using the information of users' places that we can get from check-ins. We can get FCC for places that were visited by user recently.

In this solution we are creating two sets of places – FVP (Frequently Visited Places) that has high value of FCC (HFCC) and IVP (Infrequently Visited Paces) that has low value of FCC(LFCC).

Now we are using FVP in CBF to solve the problem of lack of rating for places. Here we are considering that value of FCC is like rating for a places and if we emphasize on the context so visiting a place several times indicate user usually like to visit a place at a short interval of time. FVP generally includes users' hometown, work place, pilgrimage and business trip etc that are recurring visits of places after a certain interval of time or on some occasions. We will always keep the FVP in recommendation and we can say that these are "**always welcome recommendations**".

We are using IVP in CF, IVP includes the places that are considered to be as tourist places and users are visiting these places for enjoyment and spending holidays. These kind of places could have low value of FCC because users are interested in exploring new places each time when they plan a trip. Here we can generate some rules and with the combination of nearest neighbour approach and can find out some places of interest for making recommendation.

In Figure1, we have presented a solution using a diagram step by step.

Step 1. In the proposed solution first of all we need to get user access toke to hit the Facebook graph api and obtain the Check-Ins along with Cities. We will create an array of places P(n places) .

P = Array(P1 | FCC1, P2 | FCC2, P3 | FCC 3,…………….Pn | FCCn)

Step2. Then we need to calculate Average FCC.

Step3. In this step we will execute a loop and we will compare FCC of each place with Avg FCC.

 i) for i = 1 to n

 ii) if Avg FCC > Pi | FCCi then IVP = Array(Pi | FCCi)

 iii) else FVP = Array(Pi | FCCi)

 iv) end loop

Step4. All the places that fall under FVP are considered as "always welcome recommendations" and these will always in the recommended items.

Step5. And for IVP set we will create an attribute set for each place i to k.

 i) for i = 1 to k

 ii) Pi = {Attr1, Attr2, ……., Attrj}

 iii) Calculate the SI(Similarity Index)

 iv) if Si>1, generate the recommendations set as similar places Psimilar

 v) else Find out the nearest neighbours and generate the recommendation set for user as Pnn

 vi) end loop

Step6. In the end we will merge the three set of recommendations FVP+Psimilar+Pnn.
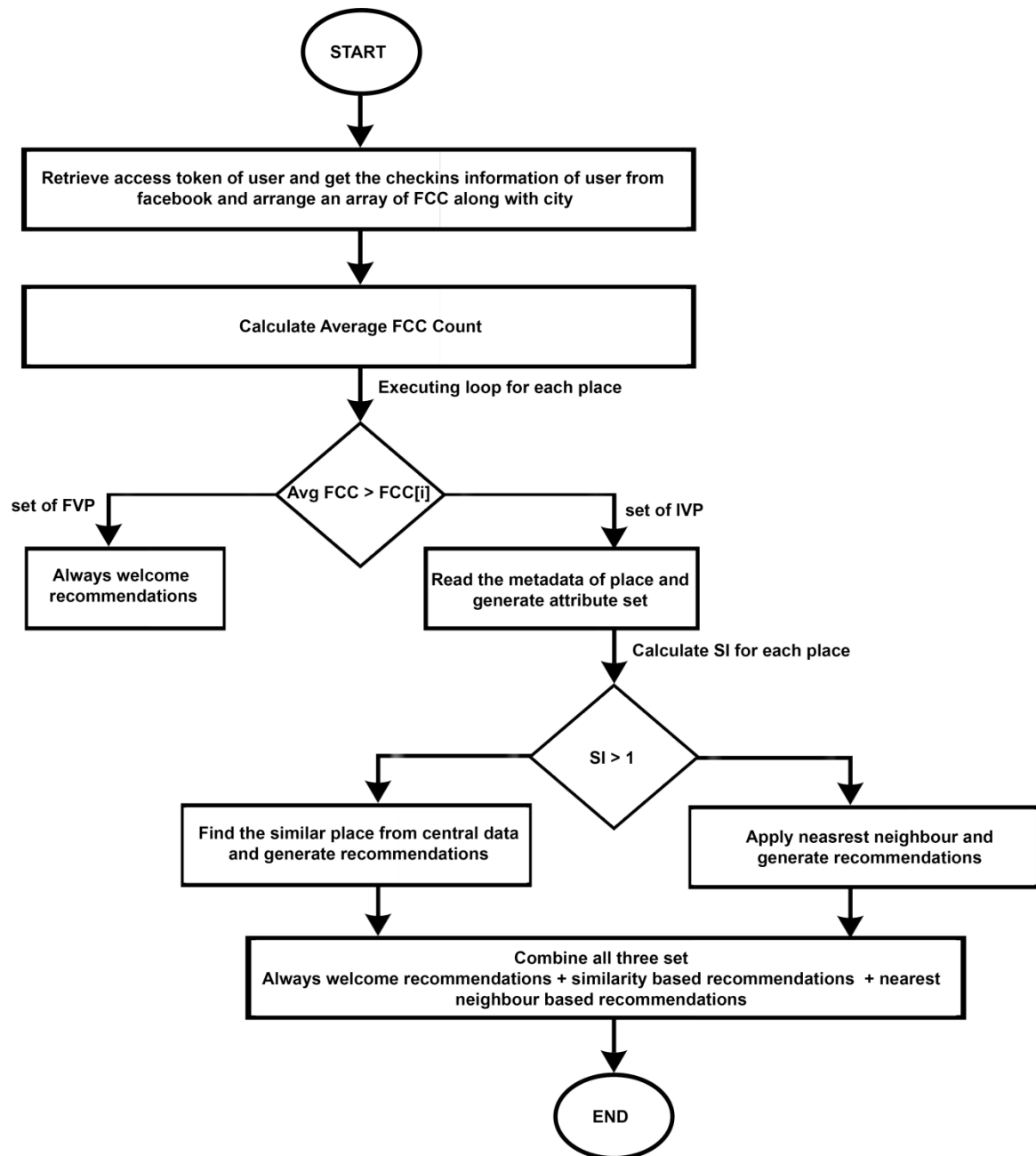
Fig1. Steps in proposed solution to generate recommendation

4. METHODOLOGY

Here we are using Facebook Graph API[15] to obtain the FCC.

4.1 Facebook Graph APIs to get user check-ins

The Graph API is the primary way to get data in and out of Facebook's platform. It's a low-level HTTP-based API that you can use to query data, post new stories, manage ads, upload photos and a variety of other tasks that an app might need to do[15].

https://graph.facebook.com/v2.3/me/tagged_places?access_token=Access_Token

4.2 Decision Making Factor for FVP and IVP

Here we need to calculate the average FCC (Avg FCC) to create to set of places FVP and IVP on the basis of the following formula.

> **Avg FCC = Total number of FCC/ Total number of places**
>
> **if FCC > Avg FCC, then we call it HFCC(High value of FCC)**
>
> **if FCC < Avg FCC, then we call it LFCC(Low value of FCC)**

4.3 Similarity Index

On the basis of a user's recently visited places with LFCC we can calculate similarity index(SI) for each place in IVP on the basis of the metadata of the places and the properties in metadata are termed as attributes . For Example, if User U has recently visited P1,P2,P3 and all three places have similar attributes X1, X2, X3 then we can say user will also like the places with similar attributes.

Two places are being called similar if there -

> **SI = Total number of similar attributes - Total number of dissimilar attributes**

Psimillar = {Places with similar attributes}

4.4 Nearest Neighbour Approach[16]

If user is interested in exploring the different places each time then we can apply the nearest neighbour approach. For example if user U1 has visited n(=3) places such as P1, P2, P3 and another user U2 has visited m(=3) places P1,P2,P4 then on the basis of nearest neighbour approach we can recommend place P4 to U1 and P3 to U2.

Pnn = {Places set obtained by applying nearest neighbour approach}

4.5 Always Welcome Recommendations

As we have already discussed about FVP, then we can treat these places as warm at all the time because these are the places that are being visited frequently by a user.

5. DISCUSSION

Here we are taking 4 users (our Facebook friends) and the places(along with set information i.e HFCC or LFCC) that they have recently visited, now we will apply the discussed methods to analyze the user behaviour for generating recommendation.

Table1.Set of places for users along with FCC

Users	Current Place	Native Place	city1	city2	city3	city4	city5
Shyam	**NCR/ HFCC**	**GorakhPur/HFCC**	Dehradun/LFCC	Mussorie/LFCC	Dharamashala/LFCC	?	?
Sunanda	**Orissa/ HFCC**	**Bhubneshawer/Orissa**	Kashmir/LFCC	Goa/LFCC	Darjeeling/LFCC	NCR/HFCC	?
Yogita	**NCR/HFCC**	**NCR/HFCC**	Guwahati/LFCC	Jammu/LFCC	Assam/LFCC	Goa/LFCC	
Ashmita	**NCR/HFCC**	**NCR/HFCC**	Singapore/LFCC	Thailand/LFCC	Agra/LFCC	Jaipur/LFCC	?

5.1 Analyzing Shyam's Preferences

We have the following set of places for Shyam from Table1.

Table2. Set of recently visited places for user Shyam

Shyam	**NCR/ HFCC**	**GorakhPur/ HFCC**	Dehradun/LFCC	Mussorie/LFCC	Dharamashala/LFCC	*?*	*?*

On the basis of the FCC(Fig1) we will divide the places in FVP and IVP sets.

FVP = {NCR, Gorakhpur}

IVP = {Dehradun, Mussorie, Dharamshala}

1) Places in FVP are considered as always welcome recommendations.

2) And for IVP set we can apply **similarity index based approach**. If we analyze the IVP set then we conclude that all the places are satisfying same set of attributes {hill station, tourist place, distance from NCR, expenses}. Hence we can conclude that user like to visit similar kind of places and we can add Shimla, Manali and other similar places with similar attributes in the recommender system output.

Table3. Shyam's places of interest for next visit shown by RS by applying proposed solution

Shyam	**NCR/ HFCC**	**GorakhPur/ HFCC**	Dehradun/L FCC	Mussorie/LF CC	Dharamasha la/LFCC	*Shimla/Man ali*	*Rishikesh/K ashmir*

5.2 Analyzing Sunanda's and Yogita Preferences

We are analysing these two users together because they both have similar kind of interest so we will apply here the nearest neighbour approach. We have obtained following information from Table1 of Yogita and Sunanda from Table 1. Now we will try to find their next place of interest to visit.

Table 4. Set of recently visited places for two users Sunanda and Yogita

Sunanda	**Orissa/ HFCC**	**Bhubnesha wer/Orissa**	Kashmir/LF CC	Goa/LFCC	Darjeeling/L FCC	NCR/HFCC	?
Yogita	**NCR/HFC C**	**NCR/HFC C**	Guwahati/L FCC	Jammu/LFC C	Thailand/LF CC	Goa/LFCC	?

On the basis of the FCC(Fig1) we will divide the places in two sets FVP and IVP.

Sunanda's FVP = { Orissa , NCR}

Sunanda's IVP = { Kashmir, Goa, Darjeeling }

Yogita's FVP = { NCR }

Yogita's IVP = { Guwahati, Jammu, Thailand, }

i) Places in FVP are considered as always welcome recommendations.

ii) And for IVP set we can apply the **nearest neighbour approach**, on the basis of this approach we can suggest Guwahati and Thailand to Sunanda and Darjeeling and Dubai to Yogita.

Table 5. Sunanda's and Yogita's places of interest for next visit shown by RS by applying proposed solution

Sunanda	**Orissa/ HFCC**	**Bhubnesha wer/Orissa**	Kashmir/LF CC	Goa/LFCC	Darjeeling/L FCC	NCR/HFCC	Guwahati/ Thailand
Yogita	**NCR/HFC C**	**NCR/HFC C**	Guwahati/L FCC	Jammu/LFC C	Thailand/LF CC	Goa/LFCC	Darjeeling/D ubai

5.3 Analyzing Ashmita's Preferences

We have the following set of places for Ashmita from Table1.

Table 6. Set of recently visited places for user Ashmita

Ashmita	**NCR/HFC C**	**NCR/HFC C**	Singapore/L FCC	Thailand/LF CC	Agra/LFCC	Jaipur/LFCC	?

On the basis of the FCC(Fig1) we will divide the places in FVP and IVP sets.

FVP = {NCR,}

IVP = { Singapore, Thailand, Agra , Jaipur }

i) Places in FVP are considered as always welcome recommendations.

ii) And for IVP set we can apply here **similarity index based approach**. If we analyze the IVP set then we conclude that the places Agra, Jaipur are satisfying same set of attributes {tourist place, distance from NCR, expenses} and Singapore, Thailand are satisfying the set of attributes {international tourist place, distance from NCR, expenses}. Hence we will show the places from both of the attribute set in recommendation.

Table 7. Ashmita's places of interest for next visit shown by RS by applying proposed solution

Ashmita	NCR/HFCC	NCR/HFCC	Singapore/LFCC	Thailand/LFCC	Agra/LFCC	Jaipur/LFCC	*Dubai/Jaisalmer*

6. CONCLUSION

In this paper we have proposed a solution to address CoCoS problem by exploiting the user social network information that was obtained by tracking the activities of the users.

Here we targeted the travel and tourism domain because this domain suffers from CoCoS as user preferences changes over time with need and changes in the choice of places. We have given the idea of exploiting social networking information of user in order to track user current preferences, likes and dislikes for making a conclusion about user's interest and preferences. Here we have proposed an idea to use FCC(using SNA) for resolving the CoCoS in travelling based recommender system. We have divided users' places in two set FVP and IVP. FVP set is always considered as warm because users are visiting these places frequently on certain intervals of time. For IVP set we are applying certain approaches – Similarity index based, nearest neighbor to find preferences of user.

7. FUTURE SCOPE

CoCoS problem is domain specific and we have proposed a solution for travel and tourism domain on the basis of SNA i.e. Check-Ins. We can improve this approach by tracking more activities of users on social network like the pages of places that user like recently, status and check-ins of other friends of user.

In other domain we can track other activities of social network to find out a solution like hashtag, user sharing , user likes, comments and the groups that users joined recently etc. And we can target different social network like Linkedin if we are working on recruitment based recommender system. There are so many social networks with different features we can target one or a combination of few to sort out the CoCoS problem in different domain similarly we can use twitter and quora as well if they help us in the domain we are working.

For the validation of the solution we can conduct a survey and then we can compare the output of proposed solution with the actual data of places for users' interest to draw a conclusion.

8. ACKNOWLEDGEMENT

9. REFERENCES

[1] T. Cover and P. Hart, "Nearest neighbor pattern classification, " in IEEE Transactions on Information Theory*,* vol. 13, no. 1, pp. 21-27, January 1967.

[2] A.I. Schein, A. Popescul, L.H. Ungar, and D.M. Pennock, "Methods and Metrics for Cold-Start Recommendations," Proc. 25th Ann. Int'l ACM SIGIR Conf., 2002.

[3] B. S. Kim, H. Kim, J. Lee and J. H. Lee, "Improving a recommender system by collective matrix factorization with tag information*,"* Soft Computing and Intelligent Systems (SCIS), 2014 Joint 7th International Conference on and Advanced Intelligent Systems (ISIS), 15th

International Symposium on*,* Kitakyushu, 2014, pp. 980-984.

[4] Bernardi, L., Kamps, J., Kiseleva, J. and J.I. Mueller, M. The Continuous Cold Start Problem in e-Commerce Recommender Systems(Vol-1448)[Online]. Available: http://ceur-ws.org/, 2015. http://ceur-ws.org/Vol-1448/paper6.pdf.

[5] Lops P, De Gemmis M, Semeraro G (2011) Content-based recommender systems: State of the art and trends. In: Recommender systems handbook, Springer, pp 73–105.

[6] Bollen D, Knijnenburg BP, Willemsen MC, Graus M (2010) Understanding choice overload in recommender systems. In: Proceedings of the fourth ACM conference on Recommender systems, ACM, pp 63–70, Barcelona, Spain, September.

[7] A. M. Rashid, I. Albert, D. Cosley, S. K. Lam, S. M. McNee, J. A. Konstan, and J. Riedl. Getting to know you: Learning new user preferences in recommender systems. In: Proceedings of the 7th International Conference on Intelligent User Interfaces, pages 127–134, 2002.

[8] facebook.com

[9] Witten I. H. and Frank I. Data Mining, Morgan Kaufman Publishers, San Francisco, 2000.

[10] Kohar, M. and Rana, C. Survey Paper on Recommendation System[Online]. Available: http://ijcsit.com, 2012. http://www.ijcsit.com/.

[11] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, John Riedl, GroupLens: an open architecture for collaborative filtering of netnews, Proceedings of the 1994 ACM conference on Computer supported cooperative work, p.175-186, October 22-26, 1994, Chapel Hill, North Carolina, United States.

[12] John S. Breese, David Heckerman and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence, pages 43-52, July 1998

[13] Deshpande, M., and Karypis, G. Item-based top n recommendation algorithms. ACM Trans. Inf. Syst. 22, 1 (2004), 143-177.

[14] Breese, J., Heckerman, D., and Kadie, C., Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence, page 4352, 1998.

[15] Overview - Graph API - Documentation - Facebook for Developers. FacebookDevelopers, 2016. https://developers.facebook.com/docs/graph-api/overview.

[16] T¨oscher A, Jahrer M, Legenstein R (2008) Improved neighborhood-based algorithms for large-scale recommender systems. In: Proceedings of the 2nd KDD Workshop on Large-Scale Recommender