Geolocation Based Recommender System

Nikita Pandey, Satvik Tandon and Princi Jain

June 6, 2021

GEOLOCATION BASED RECOMMENDER SYSTEM

Nikita Pandey
Department of computer science Engineering
Galgotias University
Email nikitapandey515@gmail.com

Satvik Tandon
Department of Computer Science Engineering
Galgotias University
Email id-rajutandon1964@gmail.com

Princi Jain
Department of Computer Science Engineering
Galgotias University
Email mukeshebeena00@gmail.com

Abstract

The project entitled "*Geo Location based Recommendation System*" is a recommendation software. The purpose of this project is to build a recommender system using collaborative filtering technique. Collaborative filtering is one of the most successful algorithm in recommender system's field. This system recommends users other 'users' of this recommender application by reading the current location of the user which is based on the details the user enters. This software relies on the concept of Machine Learning. Machine Learning is the study of computer algorithms that improve automatically through experience. The system reads the current location of the user, interacts with the database and then provide results based on the calculation obtained. The algorithm will take three different parameters into account to obtain the required result. The three different parameters being location of the user, their profile and interest list which includes the topics which the user appeals to. The algorithm which is used in the project uses Haversine formula to calculate the distance between the user and other people in the vicinity. It is capable of recoganizing various number of people and many activities which are related to user's interests. To obtain the final result, the job profile of the user and also the name of the organization they work in is also taken in account to obtain the appropriate result. The software recommends people with similar interests with their location to the user.

Keywords: **geolocation, recommender system, machine learning, Haversine formula, recommendations**

1. INTRODUCTION

"Which movie should I watch next?", "Which restaurant will you recommend with good sea food?", these are the type of questions that we ask fellow people around us on a day to day basis. But unfortunately, not everyone has the same taste as you. Therefore, going by some other person's suggestion is not the smartest decision out there. But fortunately smart brains around the world has worked together to develop an intelligent computer based technique that scoops out particular stuff from this large pool of things internet has to offer us. This system recommends us based on our liking which it predicts on the basis of user's adoption. The project is aimed at developing a geo-location based recommendations system i.e. a system which can recommend people, places, activities, events to users of the system on the basis of certain parameters. The system will take into account various parameters, some being fed in by the user, some extracted from the system database. An algorithm is devised exclusively for the purpose; taking into account the most primitive analytical approach, the system will recommend the users accordingly. The system can be used in any Location Based Social Networking System.The application is aimed at creating a mobile social network that will recommend people, places and activities around the user which have high affinity with him/her. Thus, helping the user to be aware of what's going around him/her and connect with like-minded people. Recommendation System can mainly be classified into 6 types:

- Content-based Recommendation System (CBRS):

Content-based filtering methods are based on a description of the item and a profile of the user's preference. In a content-based recommender system, keywords are used to describe the items; besides a user profile is built to indicate the type of item this user likes. In other words, these algorithms try to recommend items that are similar to those that a user liked in the past (or is examining in the present). In particular, various candidate items are compared with items previously rated by the user and the best-matching items are recommended. Content-based recommendation engine is the basic pillar of our system.

- Collaborative-based Recommendation System(CFRS):

Collaborative filtering methods are based on collecting and analysing a large amount of information on users' behaviours, activities or preferences and predicting what users will like based on their similarity to other users. A key advantage of the collaborative filtering approach is that it does not rely on machine analysable content and therefore it is capable of accurately recommending complex items. We are devising algorithm to implement Collaborative recommendation engine (user's behaviour), as an extension of our project, to make the approach hybrid.

- Demographic Recommendation System (DRS):

Demographic Filtering (DF) technique uses the demographic data of a user to determine which items may be appropriate for recommendation. This system aims to categorize the users based on attributes and make recommendations based on demographic classes. ... The benefit of a demographic approach is that it does not require a history of user ratings like that in collaborative and content based recommender systems.

- Knowledge-based Recommendation System (KBRS):

Knowledge-based recommender systems (knowledge based recommenders) are a specific type of recommender system that are based on explicit knowledge about the item assortment, user preferences, and recommendation criteria (i.e., which item should be recommended in which context).

- Context-aware Recommendation System(CARS):

A content based recommender works with data that the user provides, either explicitly (rating) or implicitly (clicking on a link). Based on that data, a user profile is generated, which is then used to make suggestions to the user.

- Hybrid Recommendation System(HRS):

Recommender systems are software tools used to generate and provide suggestions for items and other entities to the users by exploiting various strategies. Hybrid recommender systems combine two or more recommendation strategies in different ways to benefit from their complementary advantages.

In this project which is focused on Geo-location based Recommender System only 2 types of Recommender Systems are used, those are-

Content-based Recommender System(CBRS) & Collaborative-based Recommender System(CFRS)

Machine Learning

The system will use machine learning technology at its core to give suggestions. Machine learning (MI) is a branch of Computer Science that deals with the study and creation of AI systems that are able to take a decision based on their experiences.

"The goal of machine learning is to build computer systems

that can adapt and learn from their experience."

– Tom Dietterich

Geolocation

Geolocation is the identification of the real-world geographic location of an object, such as any mobile device, or any networking device. Geolocation is closely related to the use of 24 positioning systems but can be distinguished from it by a greater emphasis on determining a meaningful location (e.g. a street address) rather than just a set of geographic coordinates.

Internet and computer geolocation can be performed by associating a geographic location with the Internet Protocol (IP) address, MAC address, Wi-Fi positioning system, or device GPS coordinates, or other, perhaps self-disclosed information. Geolocation usually works by automatically looking up an IP address on a WHOIS service and retrieving the registrant's physical address.

Recommendation System

Recommender systems or recommendation systems (sometimes replacing "system" with a synonym such as platform or engine) are a subclass of information filtering system that seek to predict the 'rating' or 'preference' that user would give to an item. Recommender systems have become extremely common in recent years, and are applied in a variety of applications.

The most popular ones are probably movies, music, news, books, research articles, search queries, social tags, and products in general. However, there are also recommender systems for experts, jokes, restaurants, financial services, live insurances, persons (online dating), and twitter followers. This field is growing exponentially in the era of internet-worshiping. And mobile recommendation systems, like ours, will serve the need of smartphone users. It will be possible to offer personalized, context sensitive recommendations using a model built from the characteristics of an item (content-based approaches) or the user's social environment (collaborative filtering approaches).

Need and Scope of Recommendation Engines

Every major internet company, from media outlets to social networks to software applications, has to meet an expectation of better understanding their customers as individuals, to provide them with information and suggestions that they themselves may not even have realized they want or need. These needs are met by recommendation engines. Some of the big names that have successfully implemented such systems to drive traffic to their website and thereby increase their revenues through online sales, advertising, etc. are Amazon, eBay, Pandora, Netflix etc.

Future of Recommendation Engine

With increase in information being created by people spending much of their time on internet we need systems that can better analyse the data and give useful suggestions to people. These intelligent systems are still in their early phases with time new technologies will be developed that will enable us to create better recommendation systems.

II. RELATED WORK

To understand more about a recommender system, we had a look at past works to get more insights on the topic. To start with, by Scienstein ,a hybrid recommender system, which used both content-based and collaborativebased techniques. This approach has the potential to alleviate the problem of finding relevant research papers. Instead of solely relying on text mining, Scienstein combines citation analysis, implicit ratings, explicit ratings, author analysis and source analysis to a recommender system with a user-friendly GUI. Since all current search engines and concepts for research paper recommender systems focused mainly on one approach (text analysis, citation analysis or ratings), each concept suffered few disadvantages. The Scienstein project aims to combine the already known concepts with new ones in order to create a holistic research paper recommender system.

With Scienstein the results were, users were provided one or several of the six inputs (text, references, authors, sources, ratings or documents), adjusted algorithms to their needs, and receive recommendations for research papers. In addition to classic references, Scienstein analyses references that were added by users and that we call ‘collaborative links’.

Next, a music recommendation system by Yading Song, Simon Dixon, and Marcus Pearce.

It surveyed a general music recommender framework from user profiling, item modelling, and item-user profile matching to a series of state

of-art approaches. The components/ parameters in Music Recommender System consisted-

1. User modelling

- user profile modelling

- user listening experience modelling

2. Item profiling

- editorial metadata

-cultural metadata

-acoustic metadata

3. Query Type

To recommend items via the choice of other similar users, collaborative filtering

technique was used. Collaborative filtering was further divided into three subcategories:

memory-based, model-based, and hybrid collaborative filtering. Though it is fast and accurate, the drawbacks were obvious. First of all, the user has to know about the editorial information for a particular music item. Secondly, it is also time consuming to maintain the increasing metadata. Moreover, the recommendation results were relatively poor, since it can only recommend music based on editorial metadata and none of the users’ information had been considered.

In music recommendation system which are built over the past years, the given results tend to be more personalized and subjective. Rather than using acoustic features in content-based model and ratings in collaborative filtering, context-based information retrieval model uses the public opinion to discover and recommend music, and this was the final result of the music recommender system.

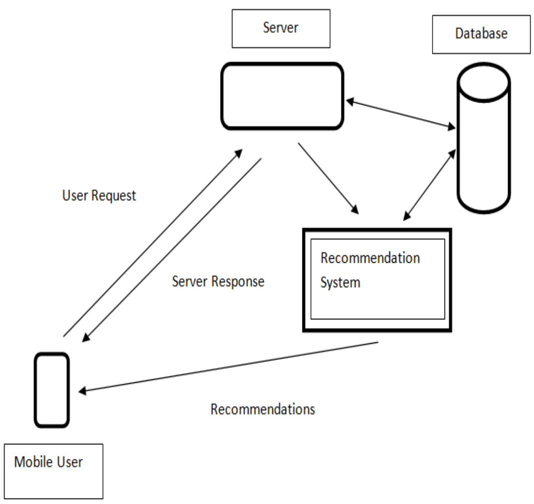
Next, a restaurant recommendation system by Xiaoyan Yang.

The parameters used in this system were a review corpus and a target user u. The output was a list of top-k restaurants recommended for u. The system consists of two main components: *profile generator* and *rating predictor*.

They map users and restaurants to a common latent space discovered from the review text. The intuition is that reviews, though in the format of unstructured free text, contain information about user preferences and opinions on different aspects of items.To make a recommendation, they first display several users randomly to choose from and as a target user u is selected, u's preference and review history would be created.Then in the next interface , the system recommends top-k restaurant list to u. These restaurants are selected from those that u has not rated/visited, based on their ratings produced by the rating predictor. To gain further information on recommendations, the system leaded to the restaurant profile interface

including recommended food according to its frequency of occurrence in review text and a list of representative reviews for efficient browsing and access to mainstream viewpoints on the restaurant.

III. DESIGN ARCHITECTURE



IV. IMPLEMENTATION

Recommendations will be based on certain parameters, either explicitly given by user or extracted from database by the system, and an affinity score will be calculated.

Affinity Score - This score will judge user's relative likeness to a certain object. To mean literally, user's affinity with all other users is represented by corresponding affinity scores.

Parameters to be used

Our research showed following parameters to be extremely useful:

1. Location

Location will be used to calculate distance between the user and objects (people, places, activities, etc.). Only objects that fall under a set radius (set by the user) will be showed to user. Also for certain objects (places, activities) preference will be given to closer objects i.e. for closer objects location will contribute more in the calculation of the affinity score.

We have divided location into two categories:

- Home Location: Where user spends most of his time
- Current Location: Where user is currently present

Both these categories will be used to

determine the affinity score.

Location will be taken from GPS through mobile. This will be in coordinates' form, that is longitudes and latitudes form.

We will use the widely known Haversine formula to calculate distance between two objects

Haversine formula:

$$a = \sin^2(\Delta\varphi/2) + \cos(\varphi_1).\cos(\varphi_2).\sin^2(\Delta\lambda/2)$$
$$c = 2.\text{atan2}(\sqrt{a}, \sqrt{(1-a)})$$
$$d = R.c$$

where φ is latitude, λ is longitude, R is earth's radius (mean radius =

note that angles need to be in radians to pass to trig functions

2. Profile

A profile will be created for each and every user (detail will be entered by user). This profile will contain attributes like name, age, gender, work status, home location (with timestamp), status, preferences.

Preferences will determine which data is publicly available. E.g. user doesn't want to show his home location.

Each attribute of profile will contribute towards affinity score calculation.

3. Interest List

User will explicitly mention his interests. These interests can be anything ranging from which game he likes to play to which programming language he likes. We will limit the maximum number of interests he can mention to 10. These interests will be used in recommendation. We will give internal score to each interest thereby determining individual weightage.

How the user interacts with the system constitutes user behaviour. All user activity will be logged by the system including who the user talks to, activities and events he likes etc. This will enable us to provide personalized recommendations.

V. ALGORITHM

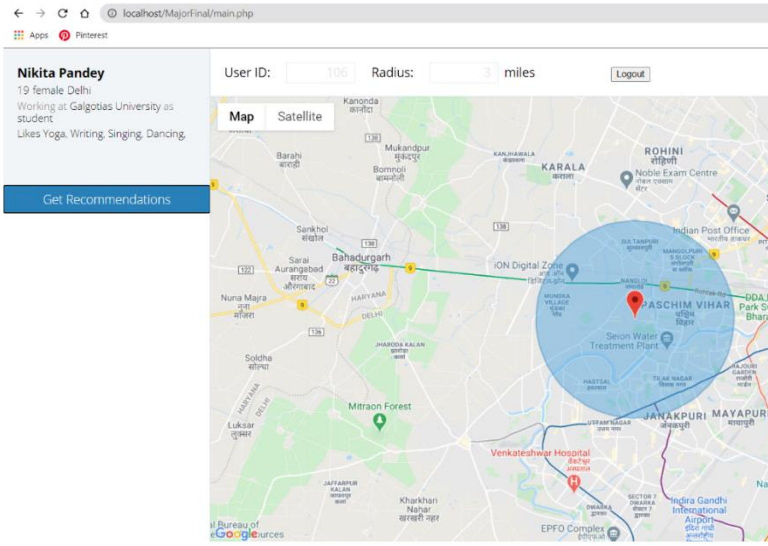
An algorithm has been devised considering the basic requirements of a recommendation system in general, then it was customized to sate the specific requirements of our system. The algorithm designed takes user's details like location, interests and job profile as input, evaluates an affinity factor based on a formula, and returns the affinity score of the recommendable people to the user.

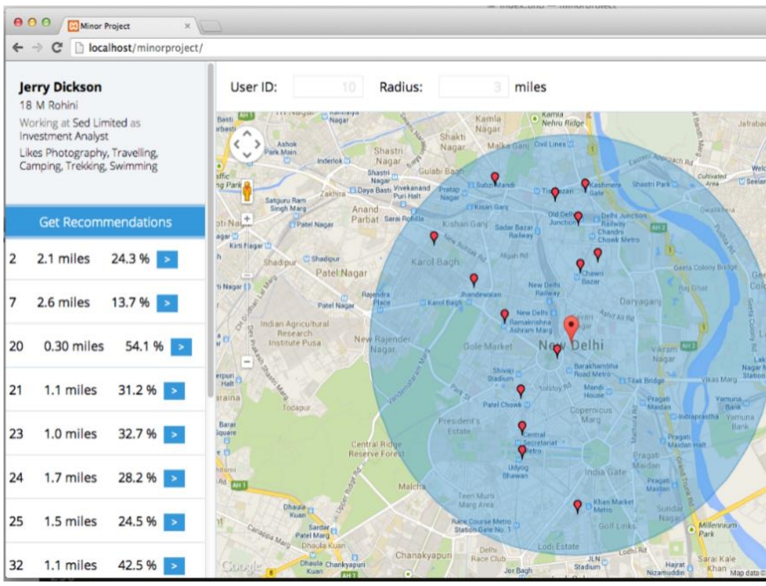
The algorithm calculates the **distance** of one user from other users using Haversine formula and calculates a score based on the distance calculated.

The algorithm also takes into account the various **interests** given by the user. The algorithm matches these interests with the interests of other users and gives a score based on the similarity of interests between two users.

Finally the **job profile** of two users is considered and a score is generated based on the similarity in the profile.

All these scores are then added and an **affinity factor** is obtained which returns the recommended users on the basis on the affinity factor generated.





VI. RESULT

The targets planned for the various phases of the project have been successfully achieved. The database which was created for the users is found to be accurate and working properly. The algorithm devised has been successfully tried and tested with the sample database for satisfactory results. Implementation of the software at front end has been handled.

ACKNOWLEDGMENT

We would like to express my gratitude towards our guide Ms. Runumi Devi for their able guidance and support in completing this project.

We would also like to express our gratitude for providing this opportunity to work on this project.

REFERENCES

1. B. Khalesi, “Social Network Analysis on Location-Based Recommender Systems”

2. P. Domingos,“A few useful things about machine learning ’’, Department of Computer Science and Engineering, University of Washington, 2015

3. D. Ionescu,“Geolocation 101:How it works, the Apps, and your privacy”, no.PCWorld,March 29,2010.

4. P. M. a. V. Sindhwani, “Recommender Systems”, Encyclopedia of Machine Learning,2010.

5. F. R. B. Shapira,“Introduction to Recommender Systems Handbook”,Springer,2011.

6. M.d.G.a.G.S. Pasquale Lops,“Content-based Recommender system:State of Art and trends”,2010.

7. “Recommender Systems in industrial contexts- PHD thesis(2012) including a comprehensive overview of many collaborative recommender systems”.

8. G. Adomavicius and A. Tuzhilin,“Toward the Next Generation of Recommender System : A survey of the state of the art and possible extensions”.

9. M. L. Montaner,“Developing trust in recommender agents”.

10. R. Burke,“Hybrid Web Recommender Systems”.

11. U. S. C. B. G. I. Systems,What is the best way to calculate the distance between 2 points?”,2010.

12. R.W. Sinnott,“Virtues of the Haversine”,Sky and Telephone 68,1984.

13. Singh, P. K., Pramanik, P. K. D., Dey, A.K. and Chaudhury, P. (2021) ‘Recommender Systems: an overview, research trends, and future directions’, Int. J. Business and Systems Research, Vol. 15, No.1, pp.14-52.

14. A Restaurant Recommender System by Analysing Ratings and Aspects in Review by Pingfu Chao, Aoying Zhou, Rong Zhang.

15. A Survey of Music Recommendation Systems and Future Perspectives by Yading Song, Marcus Pearce.

16. Scienstein: A Research Paper Recommender System by Bela Gipp, Christian Hentschel, Joeran Beel.

17. Francesco Ricci and Lior Rokach and Bracha Shapira, “Introduction to Recommender Systems Handbook”, Recommender System Handbook, Springer, 2011, pp. 1-35.

18. Elahi, Mehdi, Ricci, Francesco, Rubens, Neil (2016), “A Survey of Active Learning in Collaborative Filtering Recommender Systems”, Computer Science Review.

19. Joeran Beel; Stefan Langer; Marcel Genzmehr; Andreas Nurnberger (September 2013). “Persistance in Recommender Systems: Giving the same recommendations to the same users Multiple Times”(PDF). In Trond Aalberg, Milena Dobreva; Christos Papatheodorou; Giannis Tsakonas; Charles Farrugia(eds.). Proceedings of the 17th International Conference on Theory and Practice of Digital Libraries (TPDL 2013). Lecture Notes of Computer Science (LNCS) 8092. Springer pp. 390-394 Retrieved 1 November 2013.

20. Gomez-Uribe , Carlos A., Hunt, Neil (28 December 2015). “The Netflix Recommender System”. ACM Transactions on Management Information Systems 6(4):1-19.

21. Herlocker, J. L.; Konstan, J.A.; Terveen, L.G.; Rield, J.T.(January 2004). “Evaluating collaborative filtering recommender systems”. ACM Trans. Inf. Syst. 22(1): 5-53.

22. Said, Alan; Bellogin, Alejandro(2014-10-01). Comparative recommender system evaluation: benchmarking recommendation frameworks. Proceedings of the 8th ACM Conference on Recommender Systems.

23. Waila, P.; Singh, V.; Singh, M. (26 April 2016). “A Scientometric Analysis of Research in Recommender Systems”(PDF). Journal of Scientometric Research.

24. Hidasi, Balazs, Karatzoglou, Alexandros; Baltrunas, Linas; Tikk, Domonkos (2016-03-29). “Session-based Recommendations with Recurrent Neural Networks”.

25. Yang Ge; Hui Xiong; Alexandex Tuzhilin; Keli Xiao; Marco Gruteser; Michael J. Pazzani (2010). “An energy efficient mobile recommender system”(PDF). Proceedings of the 16th ACM SIGHOD Int’l Conf. On Knowledge Discovery and Data Mining. New York City, New York: ACM. pp 899-908. Retrieved 2011-11-17.