Volume 6, Issue 4, April 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Geolocation Based Recommender System

Deepika Bhagwani, Hemant Pandey, Neha Gupta, Yogesh Sharma
Computer Science Department, Maharaja Agrasen Institute of Technology, Rohini,
New Delhi, India

Abstract: ***The project entitled "Geo Location based Recommendation System" is a recommendaton software which is based on the concept of machine learning. The system recommends users by reading the current location of the user, interacting with the database, and provides results based on the calculations. The algorithm takes into account three different parameters to obtain the result. The algorithm used in the project uses Haversine formula to calculate the distance between user and other people in the vicinity. It is capable of recognizing various people and activities related to user's interests. The job profile and organization is also taken into account to obtain the final result. The software recommends people with similar interests with their location to the user.***

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

I. INTRODUCTION

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. [1] 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.

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. [2]

"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 24positioning 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. [3] 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. [4] 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). [5]

Content-based Filtering

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; beside, 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. [6] Content-based recommendation engine is the basic pillar of our system.

Collaborative Filtering

Collaborative filtering methods are based on collecting and analyzing a large amount of information on users' behaviors, 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 analyzable content and therefore it is capable of accurately recommending complex items. [7] We are devising algorithm to implement Collaborative recommendation engine (user's behavior), as an extension of our project, to make the approach hybrid.

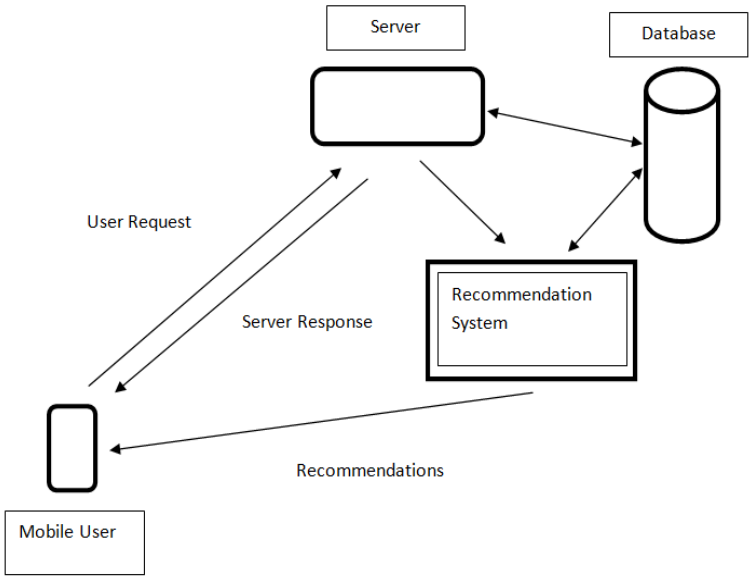
Need and Scope of Recommendation Engines

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. [8] 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. [9]

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 analyze 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. [10]

II. DESIGN ARCHITECTURE



III. 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 [11]:

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 = 6,371km)

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 behavior. 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.

IV. 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 [12] 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.

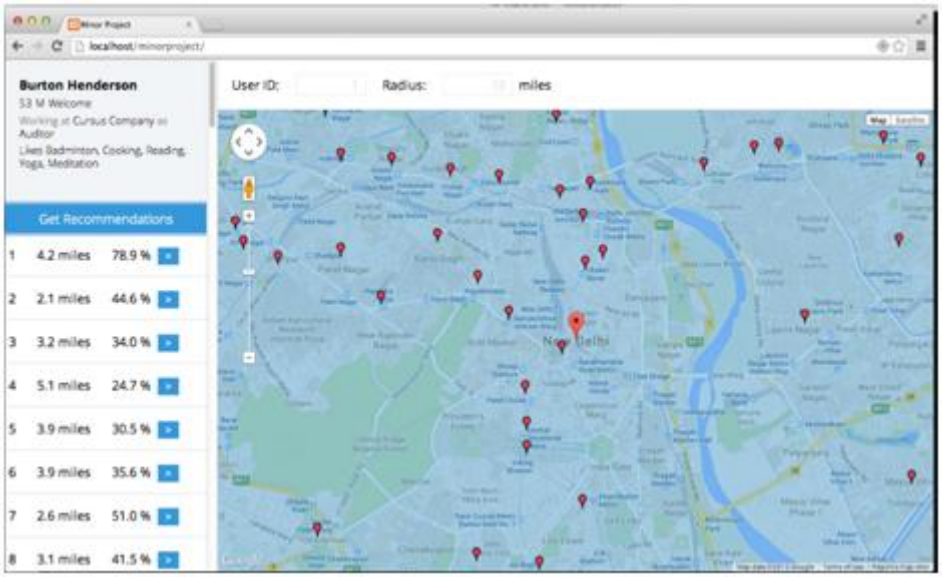
All these scores are then added and an **affinity factor** is obtained which returns the recommended users on the basis on the affinity factor generated.

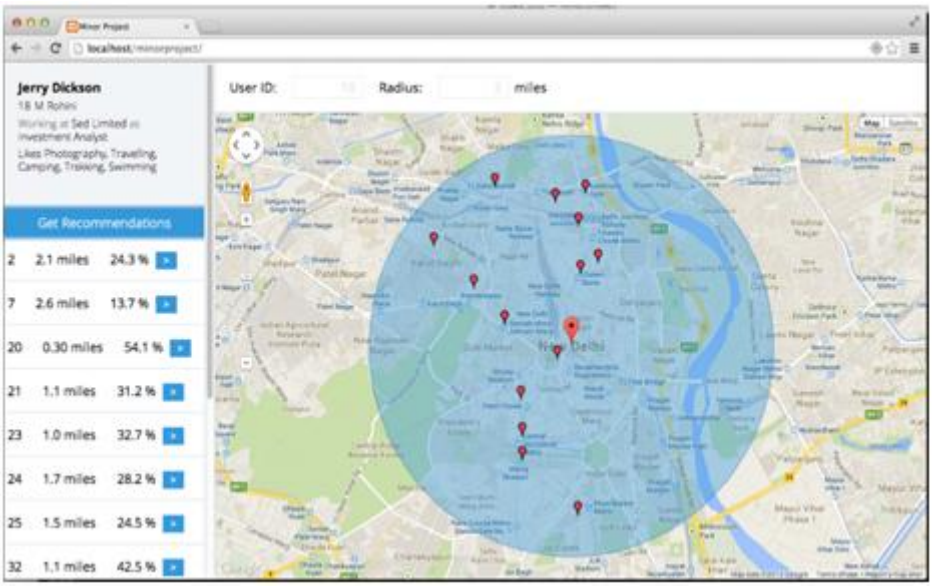
The Pseudo Code for the Algorithm

After request of recommendation by user

1. Calculate the distance (**D**) from other users in database by using Haversine formula.
2. If (D>R): next user
   Else:
   $$\alpha = 50\left(1 - \frac{D}{R}\right)$$
3. **I1, I2, I3, I4, I5** are user interests
   11, 9, 7, 5, 3 are their weightage scores
   $\beta = 0$
   for I1
   $$\beta = \beta + 11\left(1 - \frac{Difference\ \ of Position}{5}\right)$$
   for I2
   $$\beta = \beta + 9\left(1 - \frac{Difference\ \ of Position}{5}\right)$$
   for I3
   $$\beta = \beta + 7\left(1 - \frac{Difference\ \ of Position}{5}\right)$$
   for I4
   $$\beta = \beta + 5\left(1 - \frac{Difference\ \ of Position}{5}\right)$$
   for I5
   $$\beta = \beta + 3\left(1 - \frac{Difference\ \ of Position}{5}\right)$$

4. Profile
   Attribute 1 = A1.org name
   Attribute 2 = A2.job profile
   $\gamma = 0$
   If A1 matches $\gamma = \gamma + 10$
   If A2 matches $\gamma = \gamma + 5$
5. Affinity factor = $\alpha + \beta + \gamma$





V. CONCLUSION

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 with ease. All the three modules of the system were construed individually, and then integrated to work properly all together. The future plan of the project has been carefully laid out and will be carried out in the future.

REFERENCES

[1] B. Khalesi, "Social Network Analysis on Location-Based Recommender Systems," University of Amsterdam, August 2012.
[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. a. L. R. a. B. Shapira, "Introduction to Recommender Systems Handbook," Springer, pp. pp. 1-35, 2011.
[6] M. d. G. a. G. S. Pasquale Lops, "Content-based Recommender Systems: State of the 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 Systems: A Survey of the State-of-the-Art and Possible Extensions"," IEEE Transactions on Knowledge and Data Engineering, no. 17 (6): 734–749. doi:10.1109/TKDE.2005.99., June 2005.
[9] M. L. B. d. l. R. J. L. Montaner, " "Developing trust in recommender agents"," Proceedings of the first international joint conference on Autonomous agents and multiagent systems, p. part 1. pp. 304–305.
[10] R. Burke, "Hybrid Web Recommender Systems," The Adaptive Web, Peter Brusilovsky, Alfred Kobsa, Wolfgang Nejdl (Ed.), Lecture Notes in Computer Science, Springer-Verlag, Berlin, Germany, Lecture Notes in Computer Science, Vol. 4321, May 2007, 978-3-540-72078-2., pp. pp. 377-408.
[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 Telescope 68 (2), 159, 1984.

