ISSN: 0975-766X
CODEN: IJPTFI

Available Online through ***Research Article***
www.ijptonline.com

RECOMMENDATION ENGINE FOR WEB PRODUCTS

Arjun Biyani[1], Ramya G[2]
School of Information and Technology and Engineering (SITE), VIT University, Vellore -632014, TamilNadu, India.
Email: arjunbiyani@gmail.com

Received on 25-10-2016 *Accepted on 02-11-2016*

Abstract:

Recommendation engines are the system, which works as a information filter and predicts the requirement of the user on preferable search. It is very popular and implementedthese days on various web products including ecommerce, social networking site, and search engines. They collaborate information and build a information model from variousactivitiesperformed by the user, predicts related information and suggest them accordingly. Also it assumes the other interest that a user may have. It works on a algorithm which stores the users interests and show them what they like to see. The main problem with recommendation engines are sometimes the predictions are out of the pool, as the engine is based on algorithm, due to user activity it confuses and result out inappropriate results. The paper explains the strength and techniques of the recommendation engines and discussing issues and there solutions.

Key words: Simulations, events estimate, fuzzy logic.

1. Introduction

Recommendation systems are a subclass of information filtering system that seek to predict the 'rating' or 'preference' that user would give to an item (such as music, books, or movies) or social element (e.g. people or groups) [1]. Information is an entity, which relates many data/items on web. Many scientists consider the search for related work as an extremely time-consuming part of their responsibilities. The enormity of time taken is partly caused by the increasing number of publications, which grows exponentially at a yearly rate of 3.7 % [2]. Filtrations of data and simulating results related to it are still not accurately processed by many organizations, Sometimes the unwanted results are rendered by the recommender system. Recommender system plays an important role in many organizations especially in product-based company like emerging ecommerce companies. Resulting out the recommended results gives ease to user for performing activities on the web very fast and easily. Certainly it is considered by many merchants that products should be recommended through personalized or non-personalized way. In personalized the activity of the user is tracked and data is used for recommendation, in non-personalized instead of

tracking user activity it renders the result according to the trending activities. Main aim of implementing this concept is to convert browsers to buyers.

Recommendation Types

Scalable machiene learning

Recommender systems are implemented by many company's like amazon, flipkart, myntra and other ecommerce website . Focusing on the recommender type they render relative results to your search and interest for example if one is looking for a book related to programming under books category then while surfing within the website it will render recommended results relate to user preferences of search data. This data is used in many ways for analytics and evaluations. mostly all ecommerce companies uses these data to evaluate the product and rating them accordingly also it helps to manage the product listing in a way so that user can take more interest in buying those products .same as that of search engines like Google and yahoo who follows same concept of analyzing and rating the search results which are more trending and popular. Many advertising company's uses the same concept for displaying adds, they retrieve all the cookies information of the user and results or display adds according to the fetched keywords.

4. Applications of recommender system

Recommender system is useful in so many applications, but here we discuss few important and highly used applications.

1. Product Recommender: This is very important recommender system and it is related to product. This is used by on-line retailers. Foreg: myntra is a on-line vendors and it represents its product with some suggestions of product and these suggestions are based on the customer. Product Recommender is always helpful for the user.
2. Movie Recommender: This is totally bases to the customer. In that Recommender user provide rates. Foreg: rating given by the viewers to a movie.
3. News Articles Recommender: This is based on readers. Thetrending news, portal and blogs renders and articles according to the interest of the reader and these articles are based on the past searches and interest of the user. Various key terms and similar words are considered for recommending the similar type of result.

4. Content-Based Recommender:- In that method all the users and items are consider as atomic units. So the assumption and prediction as a atomic unit not as a single unit. Content-Based Recommendations sometime also known as content based filtering.

There are two basic architectures for Recommender System and these are

1. Content-Based is used for item and focus on similarities and properties of items.
2. Collaborative focus on the collaboration and understanding between user and items. It maintains a relationship between them.
3. In that user play a very important role and give the rating to the items.

Collaborative Filtering (CF) Is also known as CF. in CF we collect a feedback from the users and that feedback is collected in the form user rating of items in a specific domain and find the behavioral rating patterns of user which helps in recommending the item. There are two types of CF methods:

1- Neighborhood-based collaborative filtering.

2- Model-basedcollaborative filtering.

Neighborhood-based Collaborative Filtering:-

Sometimes it knows as memory based approach. In that method we apply predictions for the user on a subset of users. Prediction is made by weighted combination of users ratings. In that filtering we apply algorithm. In that algorithm first according to similarity of users we assign them weight then we choose some users according to their highest similarity and at last we apply some predictions on the combination of selected weighted combination of their ratings.

Cf method is not useful when we use a large items and large scale of users because the complexity of their search is high. At that time we use Item-based Collaborative Filtering.

Item-based Collaborative Filtering: In that we find the similarity between users's rated items. Thismethod is fast and produce good result. Item play a very important role in that.

Significance Weighting:This is based on overlapping items. In that we choose some small number of neighbor's based on overlapping item belongs to bad predictors. A method to handle this type of problem is that first find the Significance Weighting factor and then multipliesweight with Significance Weighting factor.

Default Voting: This method is based on correlation and that correlation is based on small correlated items with the help of this we find a value which help us to rating of item and we apply the rating on that items which have not rated till yet. That method is useful to improve the collaborative Filtering.

Inverse User Frequency: When we find the similarity between the rated items and users is not useful at that case Inverse User Frequency is used. In that we use Cf and we change the original rating of item and user by multiply it by the factor.

Case Amplification: In that method we make original

Weights.

Model-based Collaborative Filtering: In that filtering method we use user rating and in that user rating we apply few parameters on statistical model for user rating.

In the neighborhood method we make some recommendation according to the similarities between item and user and in the latent factor model we find the similarity between item and user and also we find the hidden lower dimension structure of data.Foreg: When a user gives rating to a movie and this rating is depending to the taste of the user.

Matrix factorization technique is highly useful and successful Latent Factor models and in that model user and item both are represented as unknown feature (column) vector

Hybrid Approaches: This method is the combination of both methods one is Collaborative Filtering and another is Content-Based.In the first hybrid approach

we make separate recommendation list firstly and then we combine that result and make a final result

5. Multilevel indexing engine

Multilevel indexing engine is a seek index engine which works on indexing interrelatedindexes of objects. Each object represents the entity with sub details indexes (Properties of the objects). In Fig 2.0. Object is represented with indexes each object have a unique index and sub properties with indexes in data dictionary.

Fig 2 Sample Object.

Index no.	Object Name	Relative Indexes
01	Mobile	02,04,06
02	Electronics	01,04,05

Words Starting with Letter A			
Index no	Word Name	Relative Index's	Sub Categories

			Index
01	Arts Compete	09,02	09,10
02	Architects	01,10	11,20
….			
Words starting with B			
….	….	….	…
Words Starting with Z			

Fig 2.1. Data Dictionary Table

Data dictionary in this model is a common keywords dictionary consisting of each unique word of category and item stored in a database.[4] Each keyword is represented by a index and have column of interrelated indexes for simulating proper results and recommending them accordingly. Whenever a word is fired as a query it relates the sorted keyword in dictionary and then refers the auto indexes details for that keyword.

Data dictionary (Fig 2.1) plays an important role in indexing relative search and also for recommending the non-personalized and personalized results to make a clean and efficient framework .it plays and important role for indexing and hashing. Data dictionary here is used for a particular domain where the multilevel indexing is been applied. It all starts with sorting the insertion attributes and categories as a keyword in the dictionary with having all words with the unique index no that refers and represents the word with as a unique entity[3]. As soon as the insertion of any type is done in the database the data dictionary stores the name of the entity and it's subcategory in the data dictionary in a sorted manner with a relative index to each other. Eachtuples in the database table contains the hash index of the keywords which is already defined the data dictionary. Before insertion, data dictionary is checked for the existence of the keyword. If it is available then the existing hash index of the keyword is used.

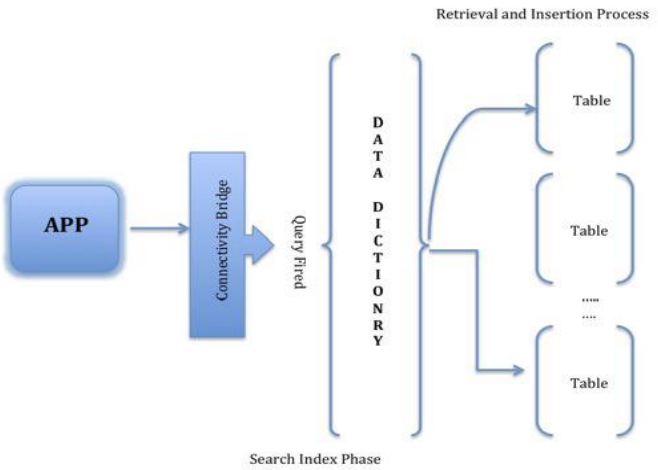

Fig. 2.2 Indexing Process.

6. Recommended results using index search

Multilevel Indexing engine helps in fetching the recommended results in a very fast manner just like a data dictionary a user table cache is maintained server side for bot personalized and non-personalized where personalized table is common for all users as for trending recommended results. We have tested on a web application where bot personalized and non-personalized results are rendered the object (Fig 3.0) in a collections with a details of a event name and its description. Each object is represented by its unique id and the category here is referenced to other collection named category with a field name computers [5].

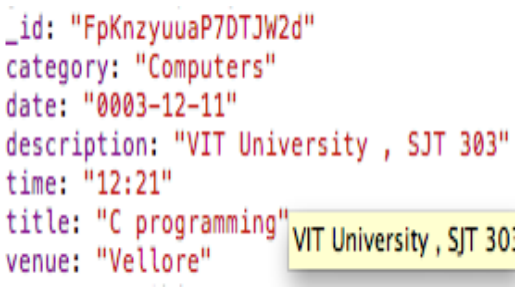

Fig 3.0 Mongo Db Collection object.

7. Implementaion module

This type of recommendation module can be applied in real-time systems (Fig 4.0). Real-time system is implemented in various domains now days, which require user interaction in shortspan of time and updating data globally either through auto synchronization or through pushing updates through sockets. It can also be used to implement and synchronize various API; one of the popular API is Google Maps.

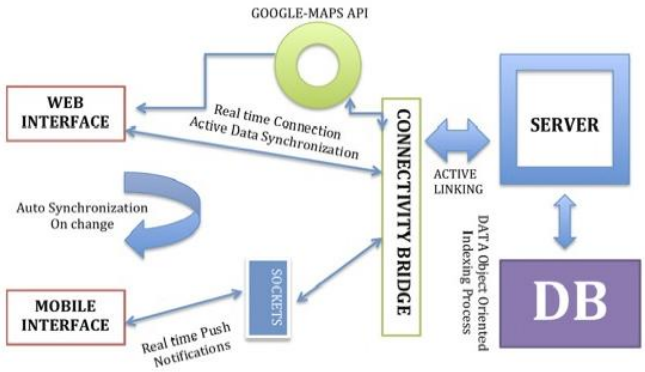

Fig 4.0. Real-time Process Model.

Application can also integrate social graph, which connects, and correlate the people mutually interested in the item or category related to users. This renders more specific and feasible results of recommendation [8].

References

1. Wikipediahttp://en.wikipedia.org/wiki/Recommender_system.

2. May, R. M. 1998. Scientific Wealth, Science, vol. 245, pp. 593-296.

3. ISSN : 2276-3299 Vol.2,Issue 3 :Page No.241-144

4. D C. Aggarwal, F. Al-Garawi, and P. S. Yu. **.**Intelligent crawling onthe World Wide Web with arbitrary predicates**..** In WWW10, HongKong, May 2001.

5. Junghoo Cho, Hector Garcia-Molina, and Lawrence Page.Efficientcrawling through url ordering. In 7th World Wide Web Conference(WWW7), Apr 1998

6. Sanjit A. Seshia, Member, IEEE, and JiaZou, Student Member, IEEE

7. Life Fellow, IEEE, Edward A. Lee, Fellow, IEEE, Slobodan Matic, Member, IEEE

8. J. Zou, J. Auerbach, D. Bacon, and E. A. Lee. Ptides on flexible taskgraph: Real-time embedded system building from theory to practice. InConference on Languages, Compilers, and Tools for Embedded Systems(LCTES), Dublin, Ireland, 2009. ACM.