International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 05 | May -2017 www.irjet.net p-ISSN: 2395-0072

Developing Software For Online Users,Predicting User Behaviour For Selection Of Consumer Goods According His/her lifestyle

Sumedh R Joshi [1], Abhishree P Deshmukh, Namrata J Andhare

[1]BE Student Jsmp's PVPIT PUNE & Fno 205 Floria near suncity Vadgaon(bk) Pune-411051
[2]BE Student Jspm's PVPIT PUNE&Srno.14 Gintanjali colony Warje Pune-411051
[3]BE Student Jspm's PVPIT PUNE&Fno 302 Opposite side of hanuman temple,Balewadi Gaon,Pune-411045.

Abstract - *Our approach to design system that provide different consumable goods to user as per his/her lifestyle. In daily life human required many products like cosmetics, home appliances etc. user cannot understand which product is suitable so human get confuse which product is good and suitable for his lifestyle. Human may purchase product which does not match his/her life style and waste money on such item. Sometime human buy product because of different advertisement that are shown on TV, poster etc. So our system provide list of product that are suitable to his/her lifestyle through reviews. Because while buying product firstly user make survey on review given by people. Also user may find all types of product in one site with their expenses so they do not require to visit different site for that user need to register to the site. This System provide more befits to user. User can purchase product as per his requirement. This System can provide 100 daily need product only.*

Key Words: *E-commerce, Artificial Intelligence, web mining*

1. INTRODUCTION

People standard of living has been increased and changed due to technology. For making life easier and effective online shopping is introduced. Today online shopping is very big trend in market, were people can buy product without wasting energy. Online shopping made human life easy but people can buy product by the effect of advertisement. They are confused while buying product they can't understand, the product is really suitable and affordable to him. In market many product are available and existing system provide all types of product i.e expensive and less expensive so obviously customer may confused because as per cost highly expensive product is best so our system can display only the product which can be affordable to user by using prediction . The system can use historical data like review given by people for analysis so that predict best product for each user. This System based on E-commerce website.

1.1 Literature Survey

Early many approaches were developed over traditional Shopping system such like Amazon, Flipkart, snapdeal, etc., to manage online shopping of different Product. These systems are attractive, user friendly and easy to operate but still do not meet the expectation that they not predict exact product for user. Following explain above approaches in detail.

Flipkart: Flipkart is very famous E-commerce website. This website provides all types of product. This system allows customer to buy product online as well as using mobile app. when customer wants to buy product online they visit website or mobile app and search product then add to cart and buy product. This is better approach over buying product from shops but this system provide all the product so that people may confuse while buying product. This system provides all the product to all customer because of this customer may get confuse which product to buy.so this system doesn't make predictions.

Amazon: Amazon is E-commerce website for online shopping. Amazon also provide mobile app for online shopping. while buying product customer need to visit website and buy the product. Amazon provides prediction but still they not provide exact product which can be suitable for human. they provide prediction such as when user visit for website, they enter product which they want to buy while entering system provide options related to search but can't provide exact product which can be match user lifestyle.so still this system can't provide exact match of product.

To overcome the above problems our prediction system is proposed to manage the overall online shopping. The goal is to save customer effort and provide exact match product to customers by providing facilities like prediction system, suitable product without searching different product,

expenses of product and fast delivery service which will result in customer satisfaction. This system saves time, reduce human effort, and gives customer satisfaction, thus beneficial customer to get product. This system provides efficiency and accuracy with cost effectiveness.

2. Problem Statement

People can buy product by the effect of advertisement, they can't understand which product is suitable for their lifestyle, so provide product which can be match for their lifestyle.

2.1 Existing System:-

Most of the product suggestions mechanism relies on pre-existing user search history to pick product . For example, Flipkart relies on a ecommerce which analysis the product which have been search recently i.e search history.

Disadvantages of Existing System:

- Existing E-commerce website recommend products to users based on history of search i.e which product search mostly by user, which may not be the most appropriate to reflect a user's preferences on product selection in real life

2.2 Statement of Scope:-

- A review-based product recommendation system for online website, which recommends product to users based on their life styles instead of recently search product.
- System discovers life styles of users & measures the similarity of life styles between users, and recommends product to users if their life styles have high similarity.
- We model a user's review documents i.e review given by product from user, from which his/her life styles are extracted by using the Collaborative filtering Algorithm.
- Similarity metric to measure the similarity of life styles between users, and calculate users
- We integrate a linear feedback mechanism that exploits the users feedback to improve recommendation accuracy.

2.3 Proposed System:

- A collaborative filtering -based product recommendation system for online shopping site, which recommends Product to users based on their life styles instead of search history.
- By taking advantage of review given by, system discovers life styles of users from review, online shopping measures the similarity of life styles between users, and recommends product to users if their life styles have high similarity.
- We model a user's review as life documents, from which his/her life styles are extracted by using the Collaborative Filtering Algorithm.
- Similarity metric to measure the similarity of life styles between users, and calculate users'.
- Impact in terms of life styles with a product-matching graph.

3. DETAILED DESIGN AND IMPLEMENTATION

This System based on E-commerce which consists of E-commerce module in this customer can enter their basic information like name, address, email id, locality area, financial detail etc.in particular format, so by using the review system can also predict product which can suitable for lifestyle. It is very important for user to clearly provide required details so that system can predict best product for user. Along with these details, system can use historical data for analysis i.e which product is mostly used daily, depending survey system can predict the value and stored in database. System provide limited product which are mostly used in daily Life. System can provide module, by using this module user can update profile so system clearly understand requirement so provide best suitable product.

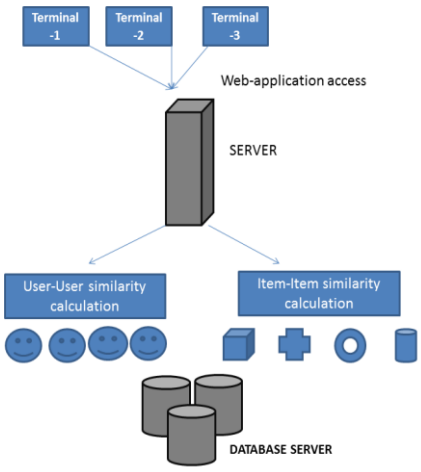

Fig: System Architecture

Product module:

In this module customer can enter only product category which they want to buy.

Edit module:

In this module, User can edit profile because needs can be changed or financial conditions are change so makes change in basic information, so system can predict product as good as possible.

Search Module:

The searching of the product can be done using predicted value in database. And use product rank associated with them.

Analysis Module:

In this module, product which can be displayed will be decided with help of predicted value associated with them. Only the product which can be suaitable can be displayed on user side.

Prediction Module:

In this module, product can be recommended by calculating similarity between users and reviews and recommend top product to users.

4. Algorithmic Analysis:

4.1 User-User Similarity Collaborative Filtering Algorithm

```
package com.algo;

import java.text.DecimalFormat;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.stream.Collectors;
import java.util.stream.DoubleStream;
import java.util.stream.IntStream;

import com.model.Review;
import com.model.User;
import com.services.ReviewServices;
import com.services.UserServices;

public class UserUserSimilarity {

        public static void main(String[] args) {
                int userID = 1;
List<Integer>recommendedProductID=predictUserUserSimi
larity(userID);
                if(recommendedProductID.size()!=0){

        recommendedProductID.forEach(productID>{Syste
m.out.println("Recommendation=====>:uid="+userID+"pro
ductid="+productID);
});
}else{
System.out.println("No Similarity Found");
    }
}
Public static List<Integer> predictUserUserSimilarity(int
userID) {
List<List<Review>>unRatedProductRatings
=ReviewServices.getUnRatedProductRatings(userID);
//unRatedMovieRatings.forEach(System.out::println);


List<List<Double>> similarityLists = new
ArrayList<>();List<Integer> unratedProductID = new
ArrayList<>();
for (int i = 0; i < unRatedProductRatings.size(); i++)
{
List<Review>unRatedProductRating
=unRatedProductRatings.get(i);
List<Double>similarityList=new
ArrayList<>();unRatedProductRating.forEach(x->{
System.out.println(x);
if(!unratedProductID.contains(x.getProduct_id()))unratedPr
oductID.add(x.getProduct_id());similarityList.add(x.getRatin
g());
});
similarityLists.add(similarityList);
                }
// *********get user similarity keys
List<List<String>>keys=newArrayList<>();
for(int i = 0; i < unRatedProductRatings.size(); i++) {
List<Review>unRatedProductRating=
unRatedProductRatings.get(i);
List<String> key=new ArrayList<>();
unRatedProductRating.forEach(x->{
//System.out.println(x.getUid());
key.add(x.getUid()+"#"+userID);
});
keys.add(key);   }
//similarityLists.forEach(System.out::println);
//keys.forEach(System.out::println);
List<Integer>userusersim=predictUserUserSimilarity(unrat
edProductID similarityLists, keys, userID);
System.out.println("user-user similarity");
//userusersim.forEach(System.out::println);
return userusersim.parallelStream().
distinct().collect(Collectors.toList());
            }
```

```
privatestaticList<Integer>
predictUserUserSimilarity(List<Integer>
unratedMovieID,List<List<Double>> similarityLists,
List<List<String>> keys, int userID) {
List<Double>userSimilarityRate=new ArrayList<>();
List<Integer>userRecommendedMoviesID=new
ArrayList<>();
List<Map<String, Double>> similarity =
getSimilarity(userID);
List<List<Double>>similarityListsVal=getSimilarityList(simil
arity, keys);
System.out.println("********finally***********");
similarityLists.forEach(System.out::println);
similarityListsVal.forEach(System.out::println);

for(int i=0; i<similarityListsVal.size() &&
i<similarityLists.size(); i++){
doubletotal=similarityListsVal.get(i).stream().mapToDouble
(map->map.doubleValue()).sum();
double total2=0;
for(intq=0;q<similarityListsVal.get(i).size()&&q<similarityLi
sts.get(i).size(); q++){
total2+=similarityListsVal.get(i).get(q)*similarityLists.get(i).
get(q); System.out.println(round(total));
// System.out.println(round(total2));

System.out.println("Similarity="+round(total2)+"/"
+round(total)+"="+round(round(total2)/round(total)));
DoublesimilarityUser=round(round(total2)/round(total));
userSimilarityRate.add(similarityUser);
}

System.out.println("$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
");
//userSimilarityRate.forEach(System.out::println);
//unratedMovieID.forEach(System.out::println);
for(intp=0;p<userSimilarityRate.size();p++){
if(userSimilarityRate.get(p)>3.5){
userRecommendedMoviesID.add(unratedMovieID.get(p));
}
}

return userRecommendedMoviesID;

}
private static List<List<Double>>
getSimilarityList(List<Map<String, Double>> similarity,
List<List<String>> keys) {
List<List<Double>>similarityListsVal=new ArrayList<>();
for(int i=0; i<keys.size(); i++){
List<Double> sval=new ArrayList<>();
 for(int j=0; j<keys.get(i).size(); j++)
int x=i;
int y=j;
//System.out.println(similarityLists.get(i).get(j));
//System.out.println(keys.get(i).get(j)+"="+keys.get(x).get
(y));
List<Double>vall=similarity.stream().filter(map>map.cont
ainsKey(keys.get(x).get(y))).map(map-
>map.get(keys.get(x).get(y)))
.collect(Collectors.toList());
sval.add(vall.get(0));
}
similarityListsVal.add(sval);
//System.out.println("===");
}
return similarityListsVal;
}

private static List<Map<String, Double>> getSimilarity(int
userID) {
List<User> allUsers
UserServices.getAllUsers();
List<String> userName = new
ArrayList<>();
List<Map<String, Double>> similarities = new
ArrayList<>();
for (int i = 0; i < allUsers.size(); i++) {
Map<String, Double> similarity = new
HashMap<>();
for (int j = i+1; j < allUsers.size(); j++) {
if
(userName.contains(allUsers.get(i).getFulname())||allUser
s.get(i).getFulname().equals(allUsers.get(j).getFulname())
)
continue;
else {
//if (i < allUsers.size() - 1) {
System.out.println(allUsers.get(i).getFulname() + "==" +
allUsers.get(j).getFulname());
double cos_theta =
calculateCosTheta(allUsers.get(i).getId(),
allUsers.get(j).getId());
similarity.put(allUsers.get(j).getId() + "#" +
allUsers.get(i).getId(), cos_theta);
System.out.println("Similarity==>" + cos_theta + "\n");
System.out.println(allUsers.get(i).getFulname() + "==" +
allUsers.get(j).getFulname());
double cos_theta1 =
calculateCosTheta(allUsers.get(i).getId(),
allUsers.get(j).getId());
similarity.put(allUsers.get(i).getId() + "#" +
allUsers.get(j).getId(), cos_theta1);
System.out.println("Similarity==>" + cos_theta1 + "\n");
// }
}
}
similarities.add(similarity);
userName.add(allUsers.get(i).getFulname());
System.out.println("==========================
==========");
}
```

```
userName.clear();
        //System.out.println("########similarity
vector##########");
//similarities.forEach(System.out::println);
return similarities;
}

        private static double round(double number) {
/*DecimalFormat numberFormat = new
DecimalFormat("#.000");
return
(Double.parseDouble(numberFormat.format(number)));*
/
if (Double.isNaN(number))
return 0;
  else {
DecimalFormat numberFormat = new
DecimalFormat("#.000");
return
(Double.parseDouble(numberFormat.format(number)));
                }
        }
private static double calculateCosTheta(int u1, int u2) {
List<List<Double>> userRatings = loadRatings(u1, u2);
userRatings.forEach(System.out::println);
double A_barB_bar =
calculateA_barB_bar(userRatings);
System.out.println("A_barB_bar=" + A_barB_bar);
double modAmodB = calculateModAmodB(userRatings);
System.out.println("modAmodB=" + modAmodB);
return round(A_barB_bar / modAmodB);
        }

private static double
calculateModAmodB(List<List<Double>> userRatings) {
                double total = 0;
                // double square = 0;
List<Double> l1 userRatings.get(0);
List<Double> l2 = userRatings.get(1);
/*
* l1.stream() .map(x->x*x)
.collect(Collectors.toList())
.forEach(System.out::println);
double l1Square = l1.stream().map(x -> x *
x).mapToDouble(y -> y.doubleValue())
.sum();
double l2Square = l2.stream().map(x -> x *
x).mapToDouble(y -> y.doubleValue())
.sum();
return (Math.sqrt(l1Square) * Math.sqrt(l2Square));}
        private static double
calculateA_barB_bar(List<List<Double>> userRatings) {
                double total = 0;
List<Double> l1 = userRatings.get(0);
List<Double> l2 = userRatings.get(1);
for (int i = 0; i < l1.size() && i < l2.size(); i++) {
total += (l1.get(i) * l2.get(i));
                }

                return total;
        }
private static List<List<Double>> loadRatings(int i, int j) {
List<Review> user1Reviews =
ReviewServices.getReviewRatingByUserID(i);
List<Review> user2Reviews =
ReviewServices.getReviewRatingByUserID(j);
        // user1Reviews.forEach(System.out::println);
                // System.out.println("=====");
                //
user2Reviews.forEach(System.out::println);List<Double>
user1Ratings =
user1Reviews.stream().map(x ->
x.getRating()).collect(Collectors.toList());
List<Double> user2Ratings =
user2Reviews.stream().map(x ->
x.getRating()).collect(Collectors.toList());

                //
user1Ratings.forEach(System.out::println);
                // System.out.println("==");
                //
user2Ratings.forEach(System.out::println);
for (int d = 0; d < user1Ratings.size(); d++) {
if (user1Ratings.get(d) == 0.0)
                                        user2Ratings.set(d, 0.0);

                }

for (int d = 0; d < user2Ratings.size(); d++) {
if (user2Ratings.get(d) == 0.0)
  user1Ratings.set(d, 0.0);

}
//user1Ratings.forEach(System.out::println);
                // System.out.println("==");
                //
user2Ratings.forEach(System.out::println);
List<List<Double>> r = new ArrayList<>();
r.add(user1Ratings);
r.add(user2Ratings);
return r;
        }

}
```

5. RESULT AND EVALUATION:

Figure shows a graph of features of the system vs. rating scale for the 3 different methods in online shopping system namely, the flipkart, snapdeal and Amazon online shopping system . The user interface of prediction based online shopping system is more attractive than the snapdeal and flipkart. Comparing the processing speed of these three

systems we find that speed of prediction based online shopping system is good. Hence prediction based online shopping system is the cheapest automation solution for the online shopping.

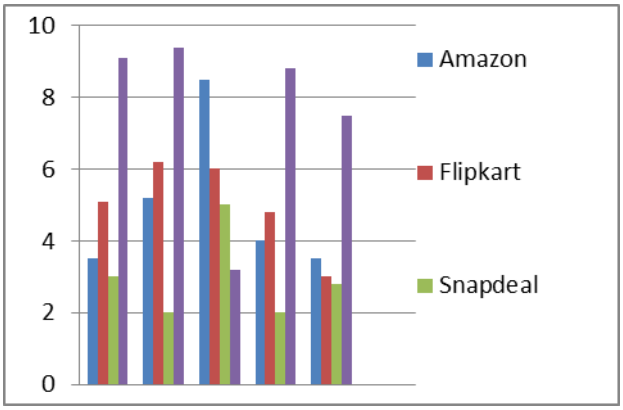

Chart -1: Result and Revolution.

6. CONCLUSION

In this paper, an automated prediction system for the online hopping is proposed to overcome the traditional method of online shopping. This system changes the process of online shopping thus reduce efforts and confusion while buying product. It eliminates human confusion while shopping due to prediction system. It saves time. It also simplifies the overall online shopping process with real time feedback from customers making the system more dynamic. It saves time of customers by providing facilities like search different product, reduce effort of human to select suitable product, online shopping,expences of product, provide product which can be affordable to user which will result in customer satisfaction. This System can give offers on product to increase the use of online shopping. This system provides customers a user-friendly, convenient and attractive user interfaces with images of every product item by which they can easily add them into cart for buy. Customers can give feedback to the system, so that system can analyze the prediction. Thus, the proposed system is advancement in the field of online shopping by automatically managing the system and provide suitable product using E-commerce technology. This automated system saves time, reduce human efforts, reduce manpower and gives customer satisfaction, thus beneficial for customer. This system provides efficiency and accuracy with cost effectiveness for user.

ACKNOWLEDGMENT

Every research work has support, guidance and help of many people and it becomes the important responsibility of the author to express deep sense of gratitude for the same. So we take this opportunity to express our deep sense of gratitude for our guide Prof. Pankaj Waghralkar, for giving us this good opportunity to select and present this topic and also for providing all the facilities and knowledge required for the successful completion of our research work. With all respect and gratitude, we would like to thank all the people, who have helped us directly or indirectly to make our work complete.

REFERENCES

[1] e-Commerce Metamodel Based on MDA G. M. Tarazona, B. C. Pelayo, O. Sanjuán and L. A.Rodríguez. IEEE LATIN AMERICA TRANSACTIONS, VOL. 12, NO. 4, JUNE 2014

[2] A Framework for Designing E-Commerce Platform of New Agricultural Products for Services,Lai Wei 12, Hong Chen 1 , Jie Zhang 1,3 1 School of Management and Economics, University of Electronic Science and Technology of China, Chengdu, 610054, China 2 College of Art and Science, Sichuan Agricultural University, Dujiangyan, 611830, China 3 Sichuan Provincial Advisory Commission, Chengdu, 610054,

[3] A Design of Security Assessment System for E-commerce Website,Xia Wang, Ke Zhang, Qingtian Wu,Research Institute of Electronic Science and Technology,University of Electronic Science and Technology of China,Chengdu, China.

[4] Security Issues Related to Web Services In E-com Mrs.Pallavi Chandratre ME computer Vidyalankar Prof. UmeshKulkarniDepartementOf ComputerEngg

[5] Vidyalankar Institute of Technology, Mumbai, 2014 International Conference on Advances in Communication and Computing Technologies