A Simple Recommendation System Using Bayesian Theory and Expectation Maximization

Submitted by

Mahmudur Rahman

2011-1-60-009

and

Noman Ibn Foysal

2011-1-60-004

Submitted to

Department of Computer Science and Engineering

East West University

Supervised By

K. M. Imtiaz-Ud-Din
Senior Lecturer,
Dept. of Computer Science and Engineering

A Project Submitted in Partial Fulfillment of the requirements for the degree of B.Sc in Computer Science and Engineering

Department of Computer Science and Engineering

East West University, Dhaka

September 16, 2015

Abstract

From ancient time, service of any shop or business or the quality of any product got reputation by the users. When a user praised highly or badly about any product or service it created a butterfly effect of individual advertisement that reached far beyond the limit that the service/product provider could reach themselves . In classical period Books and newspapers have done this reviewing job and now this is again done by the individual user using various sites on the internet .By using this reputation new or undecided user of a particular product/service were able to choose properly We have tried to create an online service that helps to review a restaurant by an individual user which will not only able to create an online reputation for the users but also help the individual /a group of users to find a better choice of places for them.

Declaration

This is to certify that ,the project is an original work done by us and has not been submitted elsewhere for the requirement of any degree or diploma or for any other purposes except for publication .

Signature of the students

Mahmudur Rahman

2011-1-60-009

Noman Ibn Foysal

2011-1-60-004

Department of Computer Science and Engineering

East West University, Dhaka , Bangladesh

Letter of Acceptance

This project entitled A Simple Recommendation System Using Bayesian Theory and Expectation Maximization submitted by Mahmudur Rahman 2011-1-60-009 and Noman Ibn Foysal 2011-1-60-009 to theDepartment of Computer Science and Engineering, East West University, Dhaka , Bangladesh isaccepted as satisfactory for partial fulfilled of the requirements for the degree of B.Sc in Computer Science and Engineering on September 16 ,2015

Board of Examiners

K. M. Imtiaz-Ud-Din

Senior Lecturer,
Dept. of Computer Science and Engineering

East West University

Dr. Shamim H Ripon

Associate Professor & Chairperson
Department of Computer Science and Engineering
East West University

Acknowledgement

This project would not be possible without the help our supervisor ,K. M. Imtiaz-Ud-Din ,Senior Lecturer ,Dept of C.S.E. East West University . He made al issues clear to us . We would like to express sincere and deep regards to him .

Table of Contents

Chapter...page

Chapter1

Chapter 2

Chapter3

Chapter 4

Chapter 5

List of figures and tables

Chapter 1

Introduction

Recommendation 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, life insurance, persons (online dating), and Twitter followers .Recommender systems typically produce a list of recommendations in one of two ways - through collaborative or content-based filtering. Collaborative filtering approaches building a model from a user's past behavior (items previously purchased or selected and/or numerical ratings given to those items) as well as similar decisions made by other users. This model is then used to predict items (or ratings for items) that the user may have an interest in. Content-based filtering approaches utilize a series of discrete characteristics of an item in order to recommend additional items with similar properties. These approaches are often combined and the system that gets evolved from this collaboration is called Hybrid Recommender System[2].

In a simple way - A recommendation system is a system whose suggestion will most probably match the Users Expectation .

1.1 Motivation

In recent years thanks to corporations like Google, Facebook we came to know many artificially intelligent systems like Youtube's video recommendation or Facebook trends . Even though we were motivated by those systems we knew how tough can it be to make one. So instead we tried to make a 'smart' one .

1.2 Objective

Our goal was to create a simple recommendation system implemented on a simple platform accessible to general Users. So we have created an web based application to run our recommendation system.

1.3 Report Outline

Chapter 2 : Summary and Review of the related works to our proposed model.

Chapter 3 : Description of our proposed model in details . Theories behind the proposed model is described and the algorithm of the whole process is explained.

Chapter 4: Description of the application.

Chapter 5 : What have we done and what can be is concluded here .

Chapter 2

State of the Art

For a workable recommendation system a number of implemented and proposed recommended systems are there in scientific research papers. To create our own model we have carefully studied some of the paper .In this chapter we will discuss the general ideas behind a recommendation system and how some researchers used those ideas to propose their own model.

2.1 General Ideas

Generally three types of recommendation system:

2.1.1 Content based recommendation system:

Content-based filtering[2] 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. This approach has its roots in information retrieval and information filtering research. To abstract the features of the items in the system, an item presentation algorithm is applied. A widely used algorithm is the tf–idf representation (also called vector space representation). To create a user profile, the system mostly focuses on two types of information: 1. A model of the user's preference. 2. A history of the user's interaction with the recommender system.

Example: music/video recommender system

2.1.2 Collaborative recommendation system

Collaborative filtering[2] 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 such as movies without requiring an "understanding" of the item itself. Many algorithms have been used in measuring user similarity or item similarity in recommender systems. For example, the k-nearest neighbor (k-NN) approach and the Pearson Correlation as first implemented by Collaborative Filtering is based on the assumption that people who agreed in the past will agree in the future, and that they will like similar kinds of items as they liked in the past. When building a model from a user's profile, a distinction is often made between explicit and implicit forms of data collection.

Example: Facebook /MySpace friend suggestion

2.1.3 Hybrid recommendation system

The term hybrid recommender[2] system is used here to describe any recommender system that combines multiple recommendation techniques together to produce its output. There is no reason why several different techniques of the same type could not be hybridized, for example, two different content-based recommenders could work together and a content based and a collaborative filtering could be mixed for better results

Example: Netflix

2.2 Location-Based Recommendation System Using Bayesian User's Preference Model in Mobile Devices

In this paper[3] Moon-Hee-Park and his colleagues tried to describe a design for mobile based which uses Bayesian network to generate suggestions to users . The authors wanted to created a system for mobile phones and hand gadgets because unlike PC's , hand phones are smaller and slower processors which would make it difficult to process a huge dataset for a recommendation application . Modern phones are built with powerful processors so we actually don't have to be too much concerned about the processing speed , but the idea to use a large dataset in a clever way to make application faster and user friendly makes it a very good design principle the application makers .

The authors have used a restaurant recommendation system as an example to describe their design principles . They use context aware recommendation system which means the design uses environment feature such as time or location as a feature . This features will be fetched form the environment by the phone's sensors and settings (GPS, temperature, time etc.) .

Normally , recommendation system use collaborative filtering to predict a product's likeness to a user ;Can be memory-based or model-based . Memory based collaborative system finds correlation between users and Model based collaborative system finds correlation between items . As model based filtering is scalable with huge datasets , this kind of filtering was preferred .

The authors divided their whole system into three modules where Bayesian network principle is used as the learning module , which is in fact the core module of the recommendation system . The context log collection is used to set up a user profile which is then processed by the learning module to create a service profile which maps the end result .

Mainly context information is collected from mobile sensors and user inputs which is used later train a Bayesian Network to create Conditional Probability Table which represents user's likelihood as a probability distribution . CPT uses Expectation maximization algorithm to learn . In a Bayesian network sometime available data to train a probabilistic model is incomplete . Expectation minimization algorithm enables to estimate the missing/incomplete data.

2.3 Bayesian Network-Based trust Model

Large distributed systems are often required in the fields of distributed computing, e commerce, various web service and file sharing methods like peer to peer. This distributed systems use autonomous agents who act on their own according to users will and choice. The autonomous agents sometimes act as service providers and they sometimes act as service consumers. While acting on their free will the performed actions of these autonomous agents can be satisfactory or unsatisfactory. Their also exists such agents that acts solely to benefit a certain monitoring party and doing so can cause harm to other agents acting in that environment. These agents are known as malicious agents. Agents most of the time work without any central supervisory party to monitor their actions. Most commonly used method to detect and stop malicious agents working in a certain environment are access authorization and user authentication. But these methods are only effective when malicious agents are known. Unknown agents or agents introduced in an environment newly can be very harmful, and this is a problem. In a centralized or decentralized environment working agents can build values according to a certain parameter which is commonly known as trust. This paper[4] describes a trust and reputation mechanism that allows agents to discover partners who meet their individual requirements, through individual experience and sharing experiences with other agents with similar preferences in a shared working environment. Bayesian networks provide a flexible method to present differentiated trust and combine different aspects of trust. A Bayesian network-based trust model is presented for a file sharing peer-to-peer application, later in this paper.

2.4 A context aware music recommendation system using fuzzy Bayesian networks with utility theory

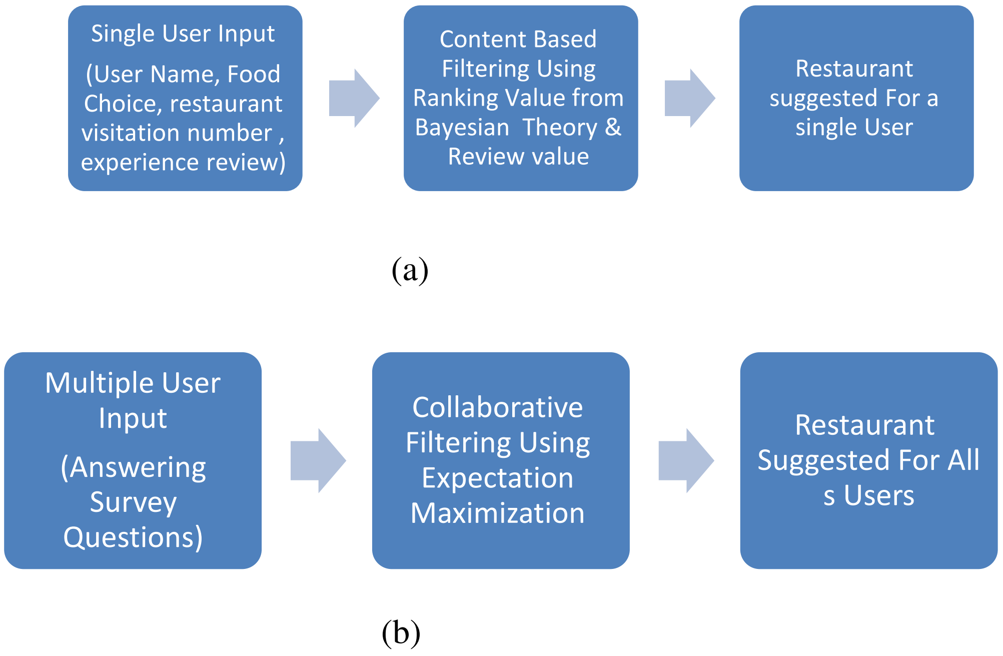
As the World Wide Web becomes the source and distribution channels of diverse digital music a large amount of music is accessible to people. Music recommendation gets required for each person since it becomes a difficult and time consuming job to search and change the music whenever the user wants it. This is the main reason of using music recommendation system. There are several music recommendation systems for both commercial and academic areas, which deal with the user preference as fixed. But the music preferred by user may change over time. The change in users choice gets unnoticed most of the time in this common music recommendation system. This paper[5] proposes a context aware music recommendation system (CA-MRS) that uses fuzzy system, Bayesian networks and the utility theory in order to recommend music according to user choice.CA-MRS uses fuzzy system to deal with vast and diverse source information. Bayesian networks are used to find and generate models for appropriate context. And finally the utility theory works to consider user preference by context. There is no general definition of what a fuzzy system is. At present most of what are called fuzzy systems to deal with large amount of information are based on production system. The working of production systems to infer appropriate data from large diverse amount of information can simply described by using a series pattern of if or else conclusions. On the other hand Bayesian networks works to generate a model between user and the result context found from fuzzy system. Utility theory then defines the whole process from the view of users need and choice and finds a suitable solution for the user that provides optimized or maximized utility for user. The proposition in this paper can be used in handheld mobile devices.

Chapter 3

Proposed Model

By carefully reading and analyzing a lot of recommendation models we came to a conclusion that , our proposed model should be simple enough to implement as an web application and also effective enough to match use's expectation . In the rest of the chapter our model is presented as briefly as possible.

3.1 Model Structure


Single User Input
(User Name, Food Choice, restaurant visitation number , experience review)
Content Based Filtering Using Ranking Value from Bayesian Theory & Review value
Restaurant suggested For a single User
Multiple User Input
(Answering Survey Questions)
Collaborative Filtering Using Expectation Maximization
Restaurant Suggested For All s Users

(a)

(b)

Figure 3.1: The basic structure of the application (a)content based filtering ,(b) collaborative filtering

A User creates their own profile by themselves . By accessing the data base which has stored the User's history we can use it to calculate a list of restaurants that the User should go. Here we are using the User's own past experience to create a list of recommendation so it is Content Based filtering and it is done by using the principals of Bayesian theory.

If we can get a recommendation list for one user we can get it for many. Now by taking the list from all the User in a group ,we can create a list for the whole group and even for a new user in the group . As we are accessing more than one users history and interaction between the Users is necessary, it is done by a Collaborative filtering done by Expectation Maximization Algorithm .

3.2 Content Based filtering using Bayesian Theory

Bayesian networks can be applied to model causal relationships between observations and their possible effects based on conditional probabilities. Using these conditional probabilities objects can be valued highly or lowly in the race of suggestion.

3.2.1 Definition of Bayesian theory

In probability theory and statistics, Bayes' theorem[1] (alternatively Bayes' law or Bayes' rule) describes the probability of an event, based on conditions that might be related to the event.

Bayes's theorem is stated mathematically as the following equation:

$$P(A|B) = \frac{P(A)\,P(B|A)}{P(B)},$$

Where A and B are events

P (A) and P (B) are the probabilities of A and B without regard to each other.

P (A l B), a conditional probability, is the probability of A given that B is true.

P (B l A), is the probability of B given that A is true.

3.2.2 Importance of Bayesian Theory

As the World Wide Web becomes the source and distribution medium of diverse digital information a large amount of data is accessible to people. But most of the time people face difficulties to choose the suitable and most associated result with their preferences. Search engines search for results that people wish to find but when results of this searches are also relatively vast in amount recommendation systems come in handy. Restaurant recommendation works to recommend users suitable restaurants that suit their choice in a particular time depending on user's previous experience. And to do this Bayesian theory can be used.

We can modify the simple Bayesian theory[3][4] to meet this circumstances like below'

P (R l C)= [P (R)* P (C l R)]/ P (C)

Where, P(C) = probability of choosing a food type independently,

P(R) = probability of choosing a restaurant independently,

P (Cl R) = probability of choosing a food type given that a particular restaurant choice is true,

P (Rl C) = probability of choosing a restaurant given that a particular food type is true as a choice

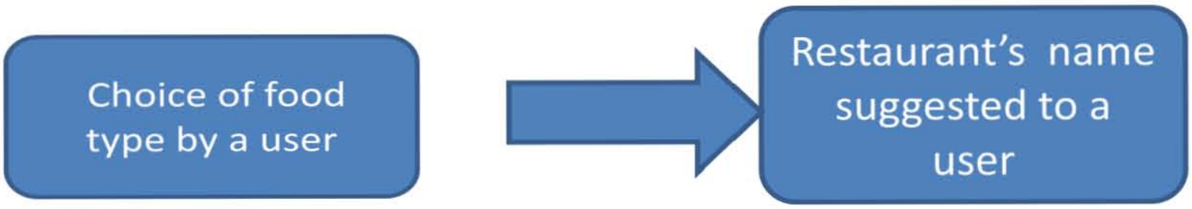

Choice of food type by a user
Restaurant's name suggested to a user

Fig 3.1.1:The Bayesian Network

3.2.3 Probability calculation in Bayesian theory

Suppose for a specific user "abcd", restaurant choice="nannarbiriyani" and foodtype="Indian";

We have,

Probability of choosing Indian food by user-abcd, p(C)=5/9,

Probability of choosing nannarbiriyani by user-abcd, p(R)=3/9,

Probability of choosing Indian food in nannarbiriyani by user-abcd, p(C|R) =3/3,

From this values we can calculate, P (R| C) = probability of choosing a nannarbiriyani given that a food type=Indian is true as a choice

Therefore, P (R | C)= [P (R)* P (C | R)]/ P (C)

=>P (R | C) =0.6

3.2.4 Calculating rank value

To calculate the rank value of nannarbiriyani for indian the probability value of nannarbiriyani for indian and review value of nannarbiriyani for indian is multiplied

Therefore,

rank value = P(R=nannarbiriyani|C= Indian)*review(nannarbiriyani for Indian)

=0.6*4

=2.4

3.3 Collaborative filtering using Expectation Maximization

3.3.1 Definition of Collaborative filtering

Collaborative filtering[2] is the process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, etc.

Mostly collaborative filtering uses algorithm s like k- nearest -neighbor algorithm K build a model .It uses explicit and implicit data to create a better user profile .

Examples of explicit data collection include the following:

- Asking a user to rate an item on a sliding scale.

- Asking a user to search.

-Asking a user to rank a collection of items from favorite to least favorite.

-Presenting two items to a user and asking him/her to choose the better one of them.

-Asking a user to create a list of items that he/she likes.

to take explicit data we have used a rating system to create a user top list and a simple survey to understand user's choice better.

3.3.2 Survey

The survey[2] includes six questions .This six questions would be asked for every restaurant in the users top list and the user is expected to be answered in yes or no. The questions are -

1. Did you enjoy the food ?

2.Did you like the service ?

3.Was the food in your price range ?

4.Do you want to visit the restaurant again ?

5.Is the restaurant near your workplace or home ?

6.Would you recommend this restaurant to your friends ?

this survey will help to Create a better user profile and also help a friend to find a nice place to eat . We will use the survey result in a Expectation maximization Algorithm to find a suitable place .

3.3.3 Expectation Maximization

Expectation-maximization[6] (EM) is a method to find the maximum likelihood estimator of a parameter theta of a probability distribution. In our case we have a very small range of questions to project a distribution and we will not exactly use EM as it is normally used .

In our survey , we can get **x** no. of yes for **n** no. of questions (in our case six questions) . So to represent this we will use binomial distribution to calculate Effectiveness .

3.3.4 Calculating effectiveness

Effectiveness(of any restaurant)= nCx*$(\text{theta})^{x}(1\text{-theta})^{n-x}$....................................3.1

Important note: In eq. 3.1 we have used Binomial distribution to calculate effectiveness . Binomial distribution is a discrete distribution , it calculates X no. of successes in N no. of trials . The answers of our survey is given in yes or no ,which re discrete value and we are trying to get X no. yes's in N questions , in our case 6 questions .

Here theta[7] is the probability of a restaurant that would be suggested by user's friends .If the user is new they can use random theta value between 0 to 1 .

To understand this better let's take a sample survey example of a user whose top list contains 5 restaurants.

Table 3.1

	Q1	Q2	Q3	Q4	Q5	Q6
Rest A	Y	Y	N	Y	Y	N
Rest B	Y	Y	Y	N	N	N
Rest C	N	Y	Y	Y	Y	N
Rest D	Y	Y	Y	Y	Y	Y
Rest E	N	N	N	N	N	N

Let's assume we do not have a theta value from update . So for each restaurants in the top list a random theta value is taken.

theta(A)=0.45

theta(B)=0.48

theta(C)=0.5

theta(D)=0.52

theta(E)=0.55

By using table 3.1 and eq. 3.1 we get effectiveness for each restaurants

E(A)=0.186

E(B)=0.311

E(C)=0.234

E(D)=0.019

E(E)=0.0101

Then we will have to normalize this,

P(any restaurant)= Effectiveness(that restaurant)/sum of the effectiveness of all top restaurant...3.2

by using 3.2 we get

P(A)=E(A)/(E(A)+E(B)+E(C)+E(D)+E(E))

P(A)=0.244

P(B)=0.409

P(C)=0.307

P(D)=0.249

P(E)=0.013

Our next job is to calculate the like likelihood[7]

3.3.5 Calculating Likelihood

Likelihood (any restaurant R) = P(R)*no of yes and P(R)*no. of no..3.3

using 3.3 we have got,

L(A)=.244*4Y and .244*2N=0.976Y and 0 .488 N

similarly,

L(B)=1.227 Y and 1.227 N

L(C)=1.228 Y and 0.614 N

L(D)=1.494 Y and 0.098 N

L(E)=0.013 Y and 0.065 N

3.3.6 Update theta value by considering the likelihood of all the friends of the User

Until now all this calculation have been done using a single users perspective . Now we have to calculate how much Restaurant "X" likely to be recommended by the other users if restaurant "X" is present in one/some/ all of the other users' list . If User 1 is new

and contains no lists of itself, then we will use the values of other users only.

Table 3.2

	Likelihood of Restaurant A	
	Yes	No
User-1 (himself)	0.976	0.488
User 2	0.544	0.544

....

....

....

User 9	0.678	1.356
User 10	1.37	.274
SUM	7.95	3.28

Now for user 1 theta (A)=SUM Y/(SUM Y+SUM N)=7.95/(7.95+3.28)=0.7079 similarly we have calculate ,theta(B), theta(C),.........(all restaurants that are present in one/some/ all of the friend's top-list)

Now if we take the value of these thetas and repeat the whole process again and again we will get the better value of theta . We can repeat this process until the rate of change becomes significantly low (less than 1%) or we can restrict the iteration no. 3-5 (by 5th iteration we would be able to get a acceptable no. of theta).

3.4 Algorithm of the Process

CreateListOfRestaurantForAnUser( )

Take input for result subjects by the User

    Restaurant name

    Number of restaurant use

    Review

    Food type for restaurant

Calculate summation ‘n’ of restaurant visit by user

Calculate summation ‘m’ of restaurant visit by user where food type is specific

Calculate summation ‘p’ of restaurant visit by user where restaurant name is specific

Calculate summation ‘q’ of restaurant visit by user where restaurant name and food type is specific

Calculate probability ‘S’ where S=[q/p]*[p/n]/[m/n]

```
Calculate rank value 'R', where R=review*S
Store value 'R' of specific restaurant into database
Update corresponding 'R' value of restaurants in database
Repeat process for each user

Return list from high 'R' value to low 'R' value
```

CreateYesandNoValue( )

```
for each restaurants that are in a user's  list
    answer the six questions
    if (updated theta value does not exist)
        assign a random theta value between 0 and 1
    else
    continue with the updated theta value

for each restaurants that are in a user's  list
    calculate effectiveness    // for example Effectiveness(restaurant A)= $nCx*(theta)^{x}(1-theta)^{1-x}$

for each restaurants that are in user's  list
```

```
    normalize        // for example P(restaurant A)= Effectiveness(restaurant
                              A)/sum of the effectiveness of all top list restaurant


for each restaurants that are in a user's  list

    calculate Likelihood        //for example Likelihood(restaurant A)=

                              Effectiveness(restaurant A)*No. of yes's and

                              Effectiveness(restaurant A)*No. of no's

                              this will create values and no values
```

UpdateThetaValue( )

```
    for all users

        for each restaurant in a user's list

            take yes and no values


for each restaurants that are in all users' list

    take the sum of the yes values= sum Y and sum of no values=sum N

    update theta        //for example theta(restaurant A)= sum Y/(sum Y+ sum N)
```

//we can continue updating theta value by repeating these two functions until it converges or these two function run for a fixed no. of iteration (3-5 times) for a good value of theta .

Chapter 4

The Application "Restaurant Recommendation System

Restaurant recommendation system follows the method of content based[2] and collaborative recommendation. Here user provides specific choice and result gets sorted out depending on the probability which are associated with user's choice and previous use data. The user can also manually take input about restaurant's name, use information, review and food type the specific restaurant served. This information gets stored into database and the application uses this information to calculate probability of restaurants which indicates the most matching associated with user's choice. In other words this application calculates probability of future restaurant visit depending on past restaurant visit data where high probability represents user will find it mostly matching with his or her choice and taste.

The general work definition of these modules is,

Display menu: Gives the user option to choose food type, go to input section to insert and store data about restaurant and also to go to the update section to update corresponding probability information[8][9][10]

Input section: takes input for database[8][9][10][11]

Map: Gives user access to Google map for finding specific restaurant location.[12]

Recommendation review: Collects input about recommendation review from user calculates effectiveness value, likelihood and probability regarding those likelihood values.[10]

Database: collects and stores data from display, input, recommendation review and calculation module and show data through output module.[10][11]

Output: Shows inserted, calculated and updated data. [8][9][10]

Example of application work:

Probability calculation in Bayesian theory

Suppose for a specific user “abcd”, restaurant choice=”nannarbiriyani” and foodtype=”Indian”;

We have,

Probability of choosing Indian food by user-abcd, $p(C)=5/9$,

Probability of choosing nannarbiriyani by user-abcd, $p(R)=3/9$,

Probability of choosing Indian food in nannarbiriyani by user-abcd, $p(C|R) =3/3$,

From this values we can calculate, $P (R| C)$ = probability of choosing a nannarbiriyani given that a food type=Indian is true as a choice

Therefore, $P (R \mid C)= [P (R)* P (C \mid R)]/ P (C)$

$\Rightarrow P (R \mid C) =0.6$

Calculating rank value

To calculate the rank value of nannarbiriyani for indian the probability value of nannarbiriyani for indian and review value of nannarbiriyani for indian is multiplied

Therefore,

rank value = P(R=nannarbiriyani| C= Indian)*review(nannarbiriyani for Indian)

=0.6*4

=2.4

<u>The expectation maximization section</u>

Here,

The effectiveness calculation formula,

Effectiveness(of any restaurant)= nCx*(theta)^{x}(1-theta)$^{n-x}$

For user abcd where restaurant choice is nannarbiriyani

Total number of review Q/A=6

Total number of review question yes answer=3

Total number of review question no answer=3

Initial probability value= 0.5

Restaurant effectiveness evalue :

0.3125

Restaurant effectiveness evalue sum:

0.859375

Restaurant probability from effectiveness, evalue[specific evalue/ sum of all evalue]:

0.36363636363636

Restaurant yes likelihood[probability*yes review]:

1.0909090909091

Restaurant no likelihood[probability*no review]:

1.0909090909091

Restaurant yes likelihood sum[summation of all yes likelihood]:

1.0909091234207153

Restaurant no likelihood sum[summation of all no likelihood]:

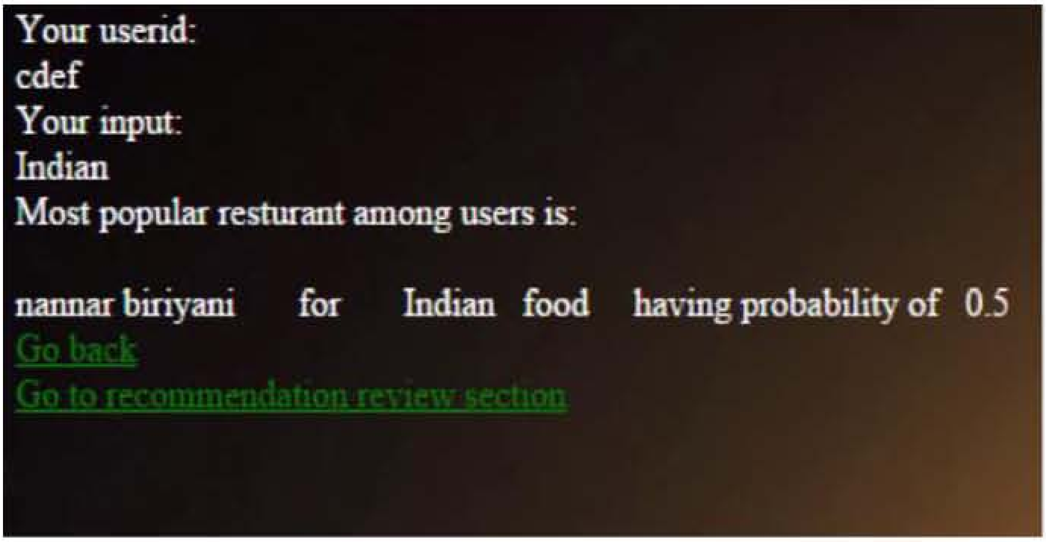
1.0909091234207153

Restaurant probability after a single iteration[sum of yes liklihood/sum of yes and no likelihood]

0.5

Application preview

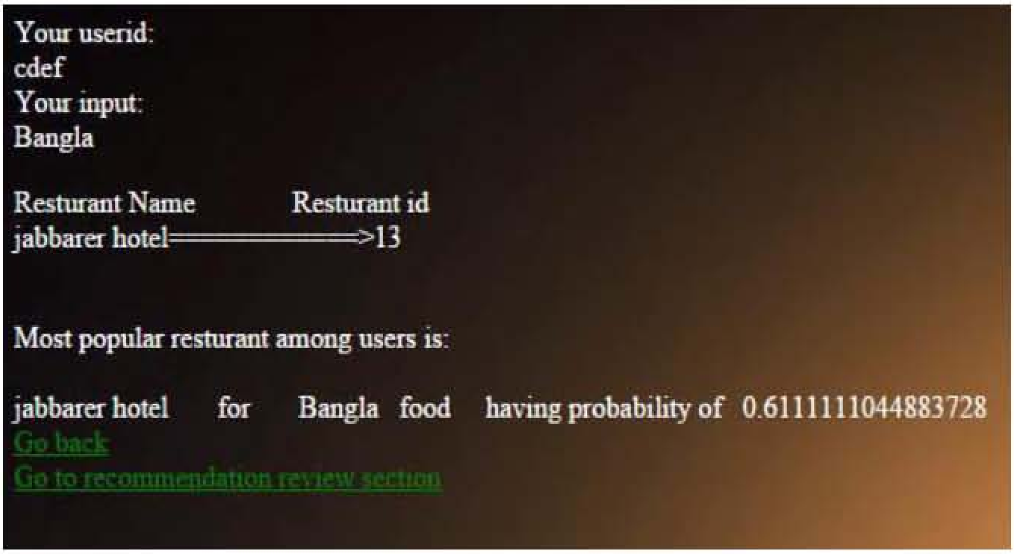
- For new user, only collaborative recommendation



Your userid:
cdef
Your input:
Indian
Most popular resturant among users is:

nannar biriyani for Indian food having probability of 0.5

- After a while user gets old and have experience



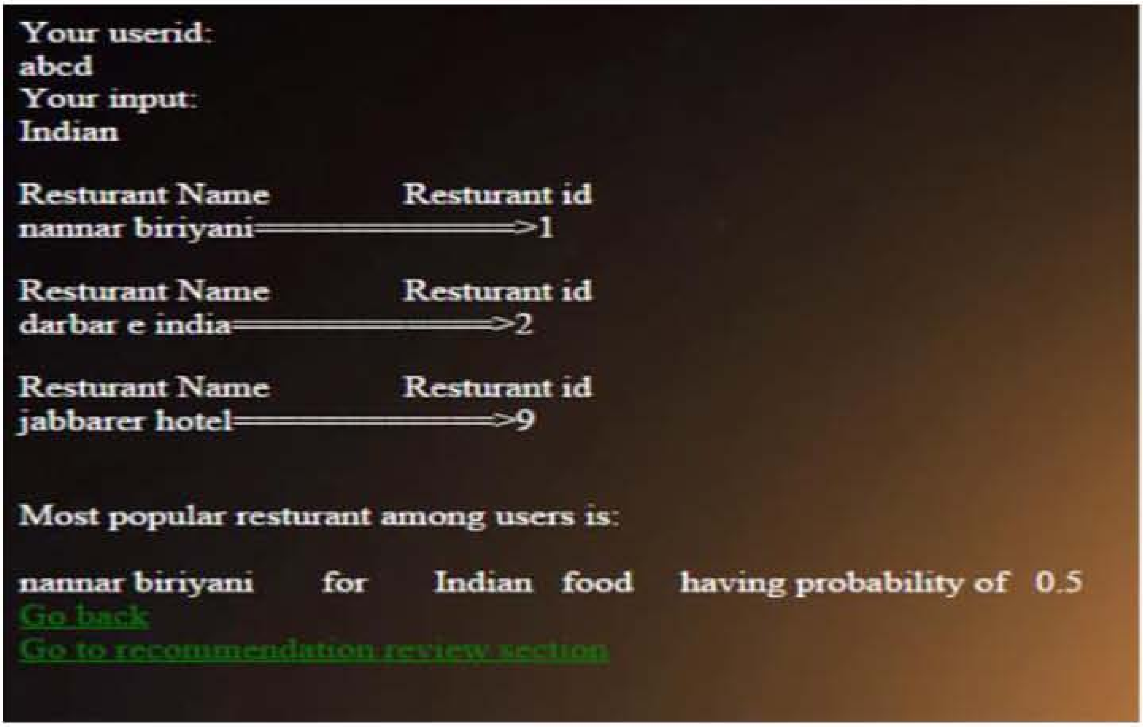
Your userid:
cdef
Your input:
Bangla

Resturant Name Resturant id
jabbarer hotel==========>13

Most popular resturant among users is:

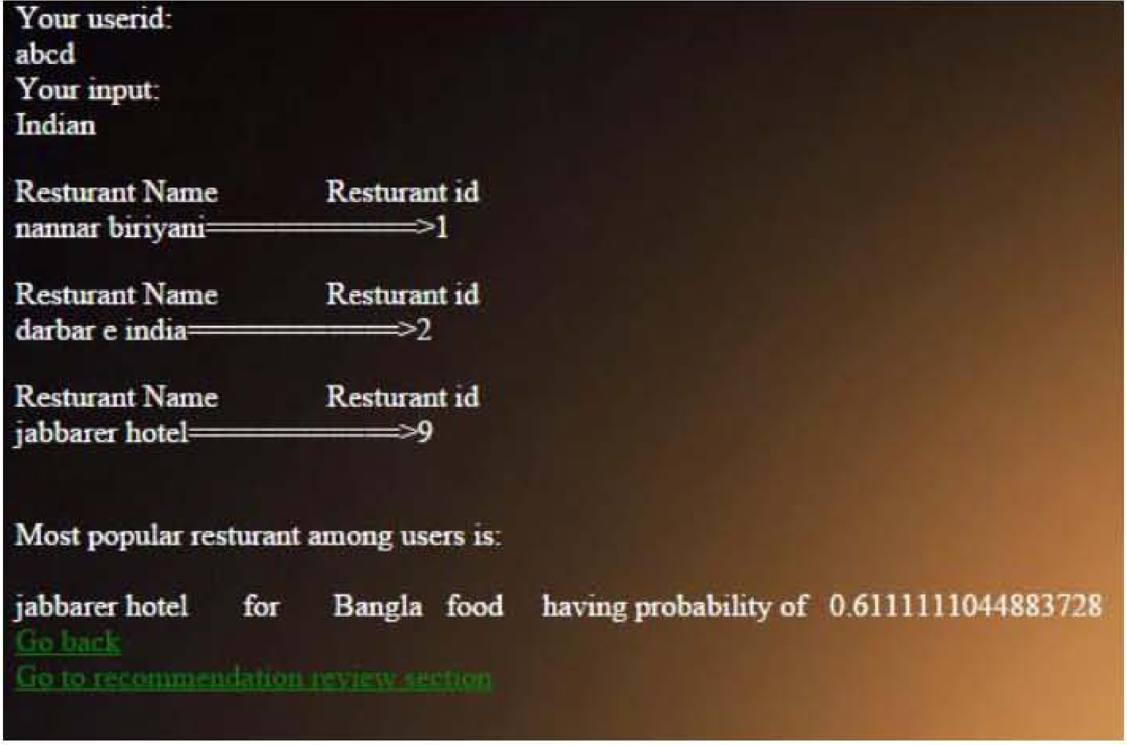
jabbarer hotel for Bangla food having probability of 0.6111111044883728

- For existing user, content based recommendation and collaborative recommendation



```
Your userid:
abcd
Your input:
Indian

Resturant Name          Resturant id
nannar biriyani================>1

Resturant Name          Resturant id
darbar e india================>2

Resturant Name          Resturant id
jabbarer hotel================>9


Most popular resturant among users is:

nannar biriyani     for     Indian   food    having probability of   0.5
Go back
Go to recommendation review section
```

- Popularity changes due to user recommendation review



```
Your userid:
abcd
Your input:
Indian

Resturant Name          Resturant id
nannar biriyani==============>1

Resturant Name          Resturant id
darbar e india==============>2

Resturant Name          Resturant id
jabbarer hotel==============>9


Most popular resturant among users is:

jabbarer hotel     for     Bangla   food    having probability of   0.6111111044883728
Go back
Go to recommendation review section
```

Chapter 5

Conclusion and Future Works

The recommendation system we have constructed is a simplistic way of filtering the users context/user's profile history and fetching the popular outcome among a group of users . The application we have created to implement this model is in its primitive stage . So , it's very difficult to understand how good the proposed actually work. But the interesting thing is the model is highly scalable and a lot can be done using the basic ideas of the proposed model .

5.1 The Strength of the system

The most positive thing about this system is it runs its filtering one after another . There are a lot of models and applied systems out there which filters various contexts and other derived data in a parallel way which eventually make them use a lot of memory .

5.2 The weaknesses of the system

The system does not make any clusters of Users which is a 'must be' filter for many existing models. As creating the system our idea was that an User will join a Group that reflects their own interests . A lot of filtering techniques were not use intentionally to see how the core algorithm works without them .

5.3 The Drawbacks of Application

The content based recommendation part only generates sorted list if and only for specific user and food type previous record exists. The application check misspelling ,for misspelling multiple entry gets inserted. Every update has to be done manually,

5.4 Future Works

The existing system can be changed in many ways to make it more useful and smart . For example a large group of users can be subdivided into smaller ones by using their age/location/income as a clustering factor. So , we will be able to understand how an User in a similar age group or income level makes their choices . Basically we can upgrade the model by -

1. Clustering Users by age , income , location

Collaborative filtering for each cluster will give a better trend for the Users in a cluster

2. We can run content based filtering for Restaurant to determine popular food choice given by a cluster of users.

This will give them a better view of customers’ needs.

Appendix

Source code:

The restaurant recommendation system is a web based application. It contains multiple HTML (hypertext markup language) and PHP(hypertext preprocessor).

They are,

1. display.html
2. input.html
3. map.html
4. recreview.html
5. updateform.html
6. show.php
7. process.php
8. recreview.php
9. update.php
10. urecreview.php

The HTML codes in HTML format sheets are given below,

display.html

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
color: white;
background-image: url("1.jpg");
}

</style>
</head>
<body>
<h1> Display section:</h1>
<br>
```

```
<form method="post" action="show.php">
User ID:<br>
<input type="text" name="userid">
<br>


Food type:<br>
<input type="radio" name="foodtype" value="Bangla">Bangla<br>
<input type="radio" name="foodtype" value="Indian">Indian<br>
<input type="radio" name="foodtype" value="Chinese">Chinese<br>
<input type="radio" name="foodtype" value="Thai">Thai<br>
<input type="radio" name="foodtype" value="English">English<br>
<input type="radio" name="foodtype" value="Fast food">Fast food<br>
<br><br>
<input type="submit" value="Submit">
<a href="http://localhost/project/input.html" style="color: yellow">Go to input section</a>
<a href="http://localhost/project/map.html" style="color: yellow">go to map view</a>
<a href="http://localhost/project/recreview.html" style="color: yellow">Go to recommendation review section</a>
<br><br>
<a href="https://twitter.com/intent/tweet?button_hashtag=foodBee" class="twitter-hashtag-button">Tweet #foodBee</a>
<script>!function(d,s,id){var js,fjs=d.getElementsByTagName(s)[0],p=/^http:/.test(d.location)?'http':'https';if(!d.getElementById(id)){js=d.createElement(s);js.id=id;js.src=p+'://platform.twitter.com/widgets.js';fjs.parentNode.insertBefore(js,fjs);}}(document, 'script', 'twitter-wjs');</script>
</form>


</body>
</html>
```

input.html

```
<!DOCTYPE html>
```

```
<html>
<head>
<style>
body {
color: white;
background-image: url("2.jpg");
}

</style>
</head>
<body>
<h1>Input section:</h1>

<form method="post" action="process.php">
User ID:<br>
<input type="text" name="userid">
<br>
Resturantname:<br>
<input type="text" name="resturantname">
<br>

Resturantreview(between 1 to 5):<br>
<input type="number" name="review" min="1" max="5">
<br>
Food type:<br>
<input type="radio" name="foodtype" value="Bangla">Bangla<br>
<input type="radio" name="foodtype" value="Indian">Indian<br>
<input type="radio" name="foodtype" value="Chinese">Chinese<br>
<input type="radio" name="foodtype" value="Thai">Thai<br>
<input type="radio" name="foodtype" value="English">English<br>
<input type="radio" name="foodtype" value="Fast food">Fast food<br>
```

```
<br>
Resturant visitation number:<br>
<input type="text" name="resturantuse">
<br>
<br><br>
<input type="submit" value="Submit">
</form>
<a href="http://localhost/project/display.html" style="color: yellow">Go to display section</a>
<a href="http://localhost/project/updateform.html" style="color: yellow">Go to update section</a>
<a href="http://localhost/project/recreview.html" style="color: yellow">Go to recommendation review section</a>
</body>
</html>
```

map.html

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="initial-scale=1.0, user-scalable=no">
<meta charset="utf-8">
<style>
html, body {
height: 100%;
margin: 0;
padding: 0;
    }
    #map {
height: 100%;
    }
.controls {
```

```
margin-top: 10px;
border: 1px solid transparent;
border-radius: 2px 0 0 2px;
box-sizing: border-box;
 -moz-box-sizing: border-box;
height: 32px;
outline: none;
box-shadow: 0 2px 6px rgba(0, 0, 0, 0.3);
}

#pac-input {
background-color: #fff;
font-family: Roboto;
font-size: 15px;
font-weight: 300;
margin-left: 12px;
padding: 0 11px 0 13px;
    •   text-overflow: ellipsis;
width: 300px;
}

#pac-input:focus {
border-color: #4d90fe;
}

.pac-container {
font-family: Roboto;
}

#type-selector {
color: #fff;
```

```
background-color: #4d90fe;
padding: 5px 11px 0px 11px;
}

#type-selector label {
font-family: Roboto;
font-size: 13px;
font-weight: 300;
}

</style>
<title>Places Searchbox</title>
<style>
    #target {
width: 345px;
    }
</style>
</head>
<body><a href="http://localhost/project/display.php">go back</a>
<input id="pac-input" class="controls" type="text" placeholder="Search Box">
<div id="map"></div>
<script>
// This example adds a search box to a map, using the Google Place Autocomplete
// feature. People can enter geographical searches. The search box will return a
// pick list containing a mix of places and predicted search terms.

functioninitAutocomplete() {
var map = new google.maps.Map(document.getElementById('map'), {
center: {lat: -33.8688, lng: 151.2195},
zoom: 13,
mapTypeId: google.maps.MapTypeId.ROADMAP
```

```
  });

  // Create the search box and link it to the UI element.
var input = document.getElementById('pac-input');
varsearchBox = new google.maps.places.SearchBox(input);
map.controls[google.maps.ControlPosition.TOP_LEFT].push(input);

  // Bias the SearchBox results towards current map's viewport.
map.addListener('bounds_changed', function() {
searchBox.setBounds(map.getBounds());
  });

var markers = [];
  // [START region_getplaces]
  // Listen for the event fired when the user selects a prediction and retrieve
  // more details for that place.
searchBox.addListener('places_changed', function() {
var places = searchBox.getPlaces();

if (places.length == 0) {
return;
    }

    // Clear out the old markers.
markers.forEach(function(marker) {
marker.setMap(null);
    });
markers = [];

    // For each place, get the icon, name and location.
var bounds = new google.maps.LatLngBounds();
```

```
places.forEach(function(place) {
var icon = {
url: place.icon,
size: new google.maps.Size(71, 71),
origin: new google.maps.Point(0, 0),
anchor: new google.maps.Point(17, 34),
scaledSize: new google.maps.Size(25, 25)
    };

    // Create a marker for each place.
markers.push(new google.maps.Marker({
map: map,
icon: icon,
title: place.name,
position: place.geometry.location
    }));

if (place.geometry.viewport) {
     // Only geocodes have viewport.
bounds.union(place.geometry.viewport);
    } else {
bounds.extend(place.geometry.location);
    }
   });
map.fitBounds(bounds);
  });
  // [END region_getplaces]
}


</script>
```

```
<script src="https://maps.googleapis.com/maps/api/js?libraries=places&callback=initAutocomplete"
async defer></script>
</body>
</html>
```

recreview.html

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
color: white;
background-image: url("5.jpg");
}

</style>
</head>
<body>
<h1>Recommendation review section:</h1>
<br>

<form method="post" action="recreview.php">
User ID:<br>
<input type="text" name="userid">
<br>
Resturantname:<br>
<input type="text" name="resturantname">
<br>

Recommendation review(for YES=1 and No=0):<br>
1.Was this particular restaurant recommendation helpful?<br>
```

```
2.Would you recommend it to your friend?<br>
3.How was the food in the restaurant?<br>
4.How was the service in the restaurant?<br>
5.Was the food in your price range?<br>
6.Will you go in that restaurant again?<br>
<br>for each answer evaluate answer value and insert the summation here:<br>
<input type="number" name="reviewy" min="0" max="6">
<br>

<input type="submit" value="Submit">
</form>
<a href="http://localhost/project/display.html" style="color: yellow">Go to display section</a>
<a href="http://localhost/project/updateform.html" style="color: yellow">Go to update section</a>

<form method="post" action="urecreview.php">
Resturantname:<br>
<input type="text" name="resturantname">
<br>
<input type="submit" value="Update">
</form>



</body>
</html>
```

updateform.html

```
<!DOCTYPE html>
<html>
<head>
<style>
```

```
body {
color: white;
background-image: url("5.jpg");
}

</style>
</head>
<body>
<h1>Update section:</h1>
<br>

<form method="post" action="update.php">
User ID:<br>
<input type="text" name="userid">
<br>
Restaurant name:<br>
<input type="text" name="resturantname">
<br>



Food type:<br>
<input type="radio" name="foodtype" value="Bangla">Bangla<br>
<input type="radio" name="foodtype" value="Indian">Indian<br>
<input type="radio" name="foodtype" value="Chinese">Chinese<br>
<input type="radio" name="foodtype" value="Thai">Thai<br>
<input type="radio" name="foodtype" value="English">English<br>
<input type="radio" name="foodtype" value="Fast food">Fast food<br>
<br><br>
<input type="submit" value="Submit">
<a href="http://localhost/project/input.php" style="color: yellow">Go to input section</a>
```

```
</form>


</body>
</html>
```

show.php

```
<?
mysql_connect("127.0.0.1","root","");
mysql_select_db("food_db");



$value0 = $_POST['userid'];
$value1 = $_POST['foodtype'];




echo "Your userid:<br>";
echo $value0;
echo"<br>";
echo "Your input:<br>";
echo $value1;



$result=mysql_query("SELECT * FROM food_table WHERE foodtype='$value1'AND
userid='$value0' ORDER BY probability DESC");
while($results=mysql_fetch_array($result)){
        echo'<p>Resturant Name
&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbspResturant
id<br>'.$results['resturantname'].'============>';
        echo $results['id'].'</p>';


}
```

```
$result2 = mysql_query("SELECT MAX(theta) AS value_sum FROM food_table");

$row = mysql_fetch_assoc($result2);

$value = $row['value_sum'];

$result3 = mysql_query("SELECT resturantname AS name FROM food_table where theta='$value'");

$row1 = mysql_fetch_assoc($result3);

$valuename = $row1['name'];

$result4 = mysql_query("SELECT foodtype AS type  FROM food_table where theta='$value'");

$row2 = mysql_fetch_assoc($result4);

$valuetype = $row2['type'];


echo "<br>Most popular resturant among users is:<br>";

echo "<br>$valuename&nbsp&nbsp&nbsp&nbsp&nbsp for &nbsp&nbsp&nbsp&nbsp&nbsp
$valuetype&nbsp&nbspfood&nbsp&nbsp&nbsp having probability of&nbsp&nbsp $value<br>";

echo '<a href="http://localhost/project/display.html" style="color:green">Go back</a><br>';

echo '<a href="http://localhost/project/recreview.html" style="color: green">Go to recommendation
review section</a>';


?>

<!DOCTYPE html>

<html>

<head>

<style>

body {

color: white;

background-image: url("1.jpg");

}


</style>

</head>

<body>

</body>
```

```
</html>
```

process.php

```
<?
mysql_connect("127.0.0.1","root","");
mysql_select_db("food_db");

$value0 = $_POST['userid'];
$value1 = $_POST['resturantname'];
$value2 = $_POST['resturantuse'];
$value3 = $_POST['review'];
$value4 = $_POST['foodtype'];
echo "Your input:<br>";
echo $value1;
echo "<br>";
echo $value2;
echo "<br>";
echo $value3;
echo "<br>";
echo $value4;


$sql = "INSERT INTO food_table (userid,resturantname, resturantuse, review, foodtype) VALUES
('$value0','$value1', '$value2', '$value3', '$value4')";


if (!mysql_query($sql)) {
	die('Error: ' . mysql_error());
}


$result1 = mysql_query("SELECT * FROM food_table");

$num_rows = mysql_num_rows($result1);

echo "<br>$num_rows Rows<br>";//row count



$result2 = mysql_query("SELECT SUM(resturantuse) AS value_sum FROM food_table WHERE userid='$value0'");

$row = mysql_fetch_assoc($result2);

$column_sum = $row['value_sum'];

echo "<br>resturant use sum<br>";

echo "<br>$column_sum<br>";//resturant use sum


$result3 = mysql_query("SELECT SUM(resturantuse) AS value_sum1 FROM food_table WHERE foodtype='$value4'AND userid='$value0'");

$row1 = mysql_fetch_assoc($result3);

$column_sum1 = $row1['value_sum1'];

echo "<br>resturant use sum for specific food type<br>";

echo "<br>$column_sum1<br>";//resturant use sum where food type fixed


$result4 = mysql_query("SELECT SUM(resturantuse) AS value_sum2 FROM food_table WHERE resturantname='$value1'AND userid='$value0'");

$row2 = mysql_fetch_assoc($result4);

$column_sum2 = $row2['value_sum2'];

echo "<br>resturant use sum for specific resturant<br>";

echo "<br>$column_sum2<br>";//resturant use sum where resturant name fixed


$result5 = mysql_query("SELECT SUM(resturantuse) AS value_sum3 FROM food_table WHERE resturantname='$value1'AND foodtype='$value4' AND userid='$value0'");

$row3 = mysql_fetch_assoc($result5);

$column_sum3 = $row3['value_sum3'];

echo "<br>resturant use sum for specific resturant and foodtype<br>";
echo "<br>$column_sum3<br>";//resturant use sum where food type,resturant name fixed



//probability calculation


$bc=$column_sum1/$column_sum;

$r=$column_sum2/$column_sum;

$bcr=$column_sum3/$column_sum2;

$rbc=($bcr*$r)/$bc;

$pval=$rbc*$value3;


$pvalr= round( $pval, 4, PHP_ROUND_HALF_ODD);


echo "<br>$pval<br>";

echo "<br>$pvalr<br>";


$sql1 = "UPDATE food_table SET probability='$pvalr'WHEREresturantname='$value1'AND
userid='$value0'";


if (!mysql_query($sql1)) {

        die('Error: ' . mysql_error());

}




echo '<a href="http://localhost/project/input.html" style="color: grey">Go back</a><br>';

echo '<a href="http://localhost/project/recreview.html" style="color: grey">Go to recommendation
review section</a>';


?>

<!DOCTYPE html>

<html>

<head>
```

```
<style>
body {
color: white;
background-image: url("1.jpg");
}

</style>
</head>
<body>
</body>
</html>
```

recreview.php

```
<?
mysql_connect("127.0.0.1","root","");
mysql_select_db("food_db");

$value0 = $_POST['userid'];
$value1 = $_POST['resturantname'];
$value2 = $_POST['reviewy'];



echo "Your input:<br>";
echo $value0;
echo "<br>";
echo $value1;
echo "<br>";
echo $value2;
echo "<br>";


$value3=6-$value2;

echo 'yes answer:<br>';

echo $value2;

echo '<br>';

echo 'no answer:<br>';

echo $value3;

echo '<br>';


$sql = "UPDATE food_table SET reviewy='$value2',reviewn='$value3'WHERE
resturantname='$value1'AND userid='$value0'";



if (!mysql_query($sql)) {

            die('Error: ' . mysql_error());

}


$n=6;

$x=$value2;


$theta=0.5;

$difftheta=1-$theta;

$pow=pow($theta,$x);

$diffnx=$n-$x;

$pow1=pow($difftheta,$diffnx);



function factorial($n) {

if ($n <= 1) {

return 1;
```

```
   } else {
return factorial($n - 1) * $n;
   }
}


function combinations($n, $x) {
   //note this defualts to 0 if $n < $x
if ($n < $x) {
return 0;
   } else {
return factorial($n)/(factorial($x)*factorial(($n - $x)));
   }
}



$evalue= (combinations(6,$value2))*($pow)*($pow1);






$result2 = mysql_query("SELECT SUM(evalue) AS value_sum FROM food_table WHERE
userid='$value0'");
$row = mysql_fetch_assoc($result2);
$evalue_sum = $row['value_sum'];
echo "<br> Restaurant evalue sum<br>";
echo "<br>$evalue_sum<br>";//resturantevalue sum


$result3 = mysql_query("SELECT SUM(evalue) AS value FROM food_table WHERE
resturantname='$value1'AND userid='$value0'");
$valueuse = mysql_fetch_assoc($result3);
$r3 = $valueuse['value'];
echo "<br> Restaurant evalue<br>";
echo "<br>$r3<br>";//resturantevalue


//probability calculation


$pevalue=($r3)/$evalue_sum;


$likey=$pevalue*$value2;


$liken=$pevalue*$value3;


echo "<br> Restaurant probability from evalue<br>";
echo "<br>$pevalue<br>";
echo "<br> Restaurant yes liklihood<br>";
echo "<br>$likey<br>";
echo "<br> Restaurant no liklihood<br>";
echo "<br>$liken<br>";



$sql2 = "UPDATE food_table SET likey='$likey' WHERE resturantname='$value1'AND
userid='$value0'";


if (!mysql_query($sql2)) {
        die('Error: ' . mysql_error());
}


$sql3 = "UPDATE food_table SET liken='$liken' WHERE resturantname='$value1'AND
userid='$value0'";


if (!mysql_query($sql3)) {
        die('Error: ' . mysql_error());
}
```

```
$sql1 = "UPDATE food_table SET evalue='$evalue' WHERE resturantname='$value1'AND
userid='$value0'";


if (!mysql_query($sql1)) {

        die('Error: ' . mysql_error());

}


$result4 = mysql_query("SELECT SUM(likey) AS value_sum1 FROM food_table WHERE
resturantname='$value1'");

$row2 = mysql_fetch_assoc($result4);

$likey_sum = $row2['value_sum1'];

echo "<br> Restaurant yes liklihood sum<br>";

echo "<br>$likey_sum<br>";//resturantlikey sum


$result5 = mysql_query("SELECT SUM(liken) AS value_sum2 FROM food_table WHERE
resturantname='$value1'");

$row3 = mysql_fetch_assoc($result5);

$liken_sum = $row3['value_sum2'];

echo "<br> Restaurant no liklihood sum<br>";

echo "<br>$liken_sum<br>";//resturant liken sum


$rtheta=($likey_sum)/($likey_sum + $liken_sum);


$sql4 = "UPDATE food_table SET theta='$rtheta' WHERE resturantname='$value1'";


if (!mysql_query($sql4)) {

        die('Error: ' . mysql_error());

}


echo '<a href="http://localhost/project/recreview.html" style="color: yellow">Go back</a>';

?>

<!DOCTYPE html>
```

```
<html>
<head>
<style>
body {
color: white;
background-image: url("1.jpg");
}

</style>
</head>
<body>
</body>
</html>
```

update.php

```
<?
mysql_connect("127.0.0.1","root","");
mysql_select_db("food_db");

$value0 = $_POST['userid'];
$value1 = $_POST['foodtype'];
$value2 = $_POST['resturantname'];

echo "Your userid:<br>";
echo $value0;

echo "<br>Your foodtype input:<br>";
echo $value1;
echo "<br>Your resturant name input:<br>";

echo $value2;



$result=mysql_query("SELECT id,review FROM food_table WHERE userid='$value0' AND
foodtype='$value1' AND resturantname='$value2' ORDER BY probability ASC");

while($results=mysql_fetch_array($result)){

            echo'<p>Resturant id<br>'.$results['id'].'</p>';

$valueid=$results['id'];

$value3=$results['review'];

echo "<br>$value3<br>";



$result1 = mysql_query("SELECT * FROM food_table");

$num_rows1 = mysql_num_rows($result1);

$num_rows=$num_rows1-0;

echo "<br>$num_rows Rows<br>";//row count


$result2 = mysql_query("SELECT SUM(resturantuse) AS value_sum FROM food_table WHERE
userid='$value0'");

$row = mysql_fetch_assoc($result2);

$column_sum = $row['value_sum'];

echo "<br>Restaurant use sum<br>";

echo "<br>$column_sum<br>";//resturant use sum


$result3 = mysql_query("SELECT SUM(resturantuse) AS value_sum1 FROM food_table WHERE
foodtype='$value1'AND userid='$value0'");

$row1 = mysql_fetch_assoc($result3);

$column_sum1 = $row1['value_sum1'];

echo "<br>Restaurant use sum for specific food type<br>";

echo "<br>$column_sum1<br>";//resturant use sum where food type fixed
$result4 = mysql_query("SELECT SUM(resturantuse) AS value_sum2 FROM food_table WHERE resturantname='$value2'AND userid='$value0'");

$row2 = mysql_fetch_assoc($result4);

$column_sum2 = $row2['value_sum2'];

echo "<br>Restaurant use sum for specific restaurant<br>";

echo "<br>$column_sum2<br>";//resturant use sum where resturant name fixed


$result5 = mysql_query("SELECT SUM(resturantuse) AS value_sum3 FROM food_table WHERE resturantname='$value2'AND foodtype='$value1' AND userid='$value0'");

$row3 = mysql_fetch_assoc($result5);

$column_sum3 = $row3['value_sum3'];

echo "<br>Restaurant use sum for specific food type & restaurant<br>";

echo "<br>$column_sum3<br>";//resturant use sum where food type,resturant name fixed



//probability calculation


$bc=$column_sum1/$column_sum;

$r=$column_sum2/$column_sum;

$bcr=$column_sum3/$column_sum2;

$rbc=($bcr*$r)/$bc;

$pval=$rbc*$value3;



$pvalr= round( $pval, 4, PHP_ROUND_HALF_ODD);


echo "<br>$pval<br>";

echo "<br>$pvalr<br>";


mysql_query("UPDATE food_table SET probability='$pvalr'WHERE id='$valueid'");

}
```

```
echo '<a href="http://localhost/project/updateform.html" style="color: yellow">Go back</a>';


?>


<!DOCTYPE html>
<html>
<head>
<style>
body {
color: white;
background-image: url("1.jpg");
}


</style>
</head>
<body>
</body>
</html>
```

---

ucreview.php

```
<?
mysql_connect("127.0.0.1","root","");
mysql_select_db("food_db");


$valuename = $_POST['resturantname'];





$result3 = mysql_query("SELECT SUM(evalue) AS value FROM food_table WHERE
resturantname='$valuename'");
```

```
$valueuse = mysql_fetch_assoc($result3);
$r3 = $valueuse['value'];
echo "<br> Restaurant evalue<br>";
echo "<br>$r3<br>";//resturantevalue

$result2 = mysql_query("SELECT SUM(evalue) AS value_sum FROM food_table");
$row = mysql_fetch_assoc($result2);
$evalue_sum = $row['value_sum'];
echo "<br> Restaurant evalue sum<br>";
echo "<br>$evalue_sum<br>";//resturantevalue sum

$result4 = mysql_query("SELECT SUM(reviewy) AS value1 FROM food_table WHERE
resturantname='$valuename'");
$valueuse1 = mysql_fetch_assoc($result4);
$r4 = $valueuse1['value1'];
echo "<br> Restaurant reviewy<br>";
echo "<br>$r4<br>";//resturantreviewy

$result5 = mysql_query("SELECT SUM(reviewn) AS value2 FROM food_table WHERE
resturantname='$valuename'");
$valueuse2 = mysql_fetch_assoc($result5);
$r5 = $valueuse2['value2'];
echo "<br> Restaurant reviewn<br>";
echo "<br>$r5<br>";//resturantreviewn

//probability calculation

$pevalue=($r3)/$evalue_sum;

$likey=$pevalue*$r4;

$liken=$pevalue*$r5;
```

```
echo "<br> Restaurant probability from evalue<br>";

echo "<br>$pevalue<br>";

echo "<br> Restaurant yes liklihood<br>";

echo "<br>$likey<br>";

echo "<br> Restaurant no liklihood<br>";

echo "<br>$liken<br>";


$sql2 = "UPDATE food_table SET likey='$likey' WHERE resturantname='$valuename'";


if (!mysql_query($sql2)) {

        die('Error: ' . mysql_error());

}


$sql3 = "UPDATE food_table SET liken='$liken' WHERE resturantname='$valuename'";


if (!mysql_query($sql3)) {

        die('Error: ' . mysql_error());

}


$result6 = mysql_query("SELECT SUM(likey) AS value3 FROM food_table WHERE
resturantname='$valuename'");

$valueuse3 = mysql_fetch_assoc($result6);

$r6 = $valueuse3['value3'];

echo "<br> Restaurant likey sum <br>";

echo "<br>$r6<br>";//resturantlikey sum


$result7 = mysql_query("SELECT SUM(liken) AS value4 FROM food_table WHERE
resturantname='$valuename'");

$valueuse4 = mysql_fetch_assoc($result7);

$r7 = $valueuse4['value4'];

echo "<br> Restaurant liken sum <br>";
echo "<br>$r7<br>";//resturantlikey sum

$rtheta=($r6)/($r6 + $r7);

$sql4 = "UPDATE food_table SET theta='$rtheta' WHERE resturantname='$valuename'";

if (!mysql_query($sql4)) {
        die('Error: ' . mysql_error());
}

echo '<a href="http://localhost/project/recreview.html" style="color: yellow">Go back</a>';



?>
<!DOCTYPE html>
<html>
<head>
<style>
body {
color: white;
background-image: url("1.jpg");
}

</style>
</head>
<body>
</body>
</html>
```

Bibliography

[1] Bayes theory, Internet: https://en.wikipedia.org/wiki/Bayes%27_theorem [September 16, 2015]

[2] Recommendation system & recommendation system types, Internet:https://en.wikipedia.org/wiki/Recommender_system [September 16,2015]

[3] Moon-Hee Park, Jin-Hyuk Hong, Sung-Bae Cho.: Location-Based Recommendation System Using Bayesian User's Preference Model in Mobile Devices

[4] Yao Wang , Vassileva, J.:Bayesian Network based trust model[October 13-17,2003]

[5] Han-Saem Park, Ji-Oh Yoo, Sung-Bae Cho.: A Context-Aware Music Recommendation System Using Fuzzy Bayesian Networks with Utility Theory[2006]

[6] Expectation maximization theory,Internet: https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm [September 16,2015]

[7] Chuong B Do & Serafim Batzoglou.: What is the expectation maximization algorithm[2008]

[8] HTML tutorial: http://www.w3schools.com/html/default.asp

[9] CSS tutorial: http://www.w3schools.com/css/default.asp

[10] PHP tutorial: http://www.w3schools.com/php/default.asp

[11] Mysql database tutorial: http://www.w3schools.com/php/php_mysql_intro.asp

[12]Google map API.: https://developers.google.com/maps/documentation/javascript/examples/places-searchbox

[13]Tweeter buttons.: https://about.twitter.com/resources/buttons